

Yeşil Robotik Hücre Çizelgeleme

Program Kodu: 1001

Proje No: 215M845

Proje Yürütücüsü:
Doç. Dr. Hakan GÜLTEKİN

Araştırmacı :

Doç. Dr. Sinan GÜREL

Bursiyerler :

Vahid EGHBAL AKHLAGHİ

Rabia TAŞPINAR

Nurdan TATAR

Tuğçe YÜCEL

HAZİRAN 2018
ANKARA

ÖNSÖZ

Robotik üretim hücreleri belirli sayıda seri makine ve bu makinelerin yüklenmesi ve boşaltılmasından ve makineler arası parça transferlerinden sorumlu bir adet robottan oluşan sistemlerdir. Bu tür sistemler firmaların rekabetçiliklerine esneklik, hız, kalite ve maliyet açısından katkı sağlamakla beraber, insan sağlığı ve ergonomisi açısından zararlı görevleri de yerine getirmektedirler. Bu sebeplerle, firmalar robotik hücreleri giderek daha fazla oranda kullanmaktadırlar.

Bu tip sistemlerde kullanılan robotlar farklı teknolojik özelliklere sahip olabilmektedir. Üzerinde en çok çalışma yapılan robot tipleri aynı anda tutabildikleri parça sayısına göre tek tutuculu (single gripper) veya çift tutuculu (dual gripper) olarak sınıflandırılmaktadır. Sistemde üretilen parçalar da özdeş veya farklı tiplerde olabilmektedir. Yüksek yatırım maliyeti gerektiren bu sistemlerden maksimum faydanın sağlanabilmesi için çeşitli karmaşık operasyonel problemlerin çözülmesi gerekmektedir. Bunlar arasında robot hareket sıralamasının ve parça sıralamasının belirlenmesi sayılabilir. Diğer taraftan, endüstriyel robotlar önemli ölçüde enerji tüketmektedirler ve tükettikleri enerji hareket hızlarına bağlıdır. Fakat daha önce hiçbir çalışma robotların enerji tüketimini planlama probleminin içerisine dahil etmemiştir. Bu proje bu sebeple özgünlük taşımaktadır.

Bu proje kapsamında tek veya çift tutuculu robotların bulunduğu tek tip veya değişik tip parça üreten hücrelerdeki parça sıralaması ve robot hareket sıralaması problemleri robot hareket hızlarının belirlenmesi problemiyle eş zamanlı olarak ele alınarak hem üretim hızını maksimize eden ve hem de enerji tüketimini minimize eden çözümler belirlemek üzere kesin ve sezgisel çözüm algoritmaları geliştirilmiştir.

Çalışmanın bir diğer katkısı, tek ve çift tutuculu robotların enerji tüketimi ve üretim hızı açılarından birbirleriyle karşılaştırılarak daha maliyetli olan çift tutuculu robotların hangi durumlarda diğerinden daha iyi sonuçlar verdiğinin belirlenmiş olmasıdır. Elde edilen sonuçların parametrelere duyarlılıkları tespit edilerek yönetsel çıkarımlar sağlanmıştır.

Çalışmanın sonuçları, robotik hücre çözelgeleme alanında robot hız kontrolünü ilk defa ele alarak bu literatüre çok farklı bir katkı sağlamıştır. Dolayısıyla, hem bu alanda çalışan araştırmacılar tarafından ve hem de bu tip sistemleri kullanan veya kullanmayı düşüne firmalar veya bu firmalara bu tip sistemlerin satışını ve kurulumunu yapan firmalar tarafından kullanılabilir.

Bu projeye sağladığı kaynak ve destek ile projenin gerçekleşmesini sağlayan ve lisans ve lisansüstü öğrencilerinin proje kapsamında yaptıkları çalışmalarla bilgi ve deneyim kazanarak iyi bir şekilde yetişmesine katkıda bulunan TÜBİTAK'a proje ekibi olarak teşekkürlerimizi sunuyoruz.

İÇİNDEKİLER

ÖNSÖZ	i
ÖZET	vi
ABSTRACT	vii
1 GİRİŞ	1
2 Tek Tip Parça Üreten Tek Tutuculu Robotlu Hücrelerde Robot Hareket Sıralamasının ve Hızlarının Belirlenmesi	5
2.1 GİRİŞ	5
2.2 LİTERATÜR ÖZETİ	7
2.2.1 Robotik Hücre Çizelgelemesi	7
2.2.2 Robot Hızları ve Enerji Tüketimi	10
2.2.3 Çevrim Zamanı ve Enerji Tüketimini Ele Alan 2-Kriterli Optimizasyon	11
2.3 PROBLEM TANIMI VE MATEMATİKSEL MODELLER	12
2.3.1 2-Makinelik Robotik Hücreler İçin Çevrim Zamanı Hesabı	13
2.3.2 Robotun Enerji Tüketim Fonksiyonu	16
2.3.3 Matematiksel Modeller	17
2.4 HESAPLAMALI DENEYLER VE ANALİZLER	21
2.4.1 Kontrol Edilebilir Hız ve Sabit Hız Çözümlerinin Karşılaştırılması	21
2.4.2 Etkin Çözüm Kümeleri	24
2.5 Sonuç	27
3 Tek Tip Parça Üreten m-Makinelik Genel Robotik Hücreler	30
3.1 LİTERATÜR TARAMA	31
3.1.1 Robotik Hücre Çizelgelemesi	32
3.1.2 Robot Hızları ve Enerji Tüketimi	34
3.1.3 Çevrim Zamanı ve Enerji Tüketimini Ele Alan 2-Kriterli Optimizasyon	35
3.2 Geliştirilen Matematiksel Modeller	36
3.2.1 Karma Tam Sayılı Doğrusal Olmayan Model (MINLP)	37
3.2.2 Tamsayı İkinci Derece Konik Programlama (MISOCP) Modeli	42
3.2.3 MINLP - MISOCP Karşılaştırması	45
3.3 Sezgisel Algoritma (META)	46
3.3.1 META - MISOCP Karşılaştırması	60
3.3.2 META'nın Daha Büyük Boyutlu Problemlerde Performansı	70
3.4 Robot Hız Kontrolü Uygulamasının Sağladığı Tasarruf	82
3.5 Sonuç	84

4	Çift Tutuculu Malzeme Taşıyıcı Robotlu Hücrelerde Enerji Duyarlı Çizelgeleme	85
4.1	Giriş	85
4.2	Matematiksel Programlama Formülasyonları	87
4.3	MISOCP ve MINLP Modellerinin Hesaplamalı Karşılaştırması	97
4.4	Etkin Çözüm Kümesi Türetme Algoritması (ETA)	104
4.4.1	ETA: Aşama 1	106
4.4.2	ETA: Aşama 2	110
4.4.3	ETA: Aşama 3	112
4.4.4	ETF: Aşama 4	116
4.4.5	ETA: Aşama 5	117
4.5	Hesaplamalı Deney Sonuçları	117
4.6	Tek ve Çift Tutuculu m -Makinelik Robotik Hücrelerin Karşılaştırması	124
4.7	Sonuç ve Değerlendirme	128
5	Değişik Tip Parça Üreten Tek Tutuculu Robotik Hücreler	130
5.1	Giriş	130
5.2	Literatür Taraması	131
5.3	Matematiksel Modeller	134
5.4	Hibrit Metasezgisel Algoritma (HMA)	148
5.4.1	Parametre Kalibrasyonu	157
5.5	Hesaplamalı Deney Sonuçları	159
5.6	Sonuç ve Değerlendirme	167

ŞEKİL LİSTESİ

1	2 makinelik tek tutuculu robotik hücre	6
2	S_1 döngüsü	14
3	S_2 döngüsü	15
4	Enerji fonksiyonu alternatifleri	17
5	Çevrim zamanına karşın robotun enerji tüketimi	18
6	S_1 çevrimi için seçilen çevrim sürelerinde etkin çözüm kümesi	26
7	S_2 çevrimi için seçilen çevrim sürelerinde etkin çözüm kümesi	26
8	Farklı çevrim süresi değerlerinde S_1 ve S_2 döngülerinin enerji tüketim değerleri	26
9	2 makinelik çift tutuculu robotik hücre	86
10	Robot döngüsünde olabilecek dört alternatif aktivite arası parametrik mesafeler	89
11	2 makine, ÖT, DV veri kümesi için örnek etkin çözüm kümesi	100
12	2 makine, ÖT, YV veri kümesi için örnek Pareto önyüz	101
13	3 makine, GT, YV veri kümesi için örnek Pareto önyüz	102
14	3 makine, GT, YV veri kümesi için örnek Pareto önyüz	103
15	Etkin Çözüm Kümesi	105
16	5 makine için Örnek Gantt Şeması	111
17	2-Makinelik 2-Tip Parça için Çevrim Zamanı ve Enerji Ödünleşimi	147

18	3-Makinelik 2-Tip Parça için Çevrim Zamanı ve Enerji Ödünleşimi	147
19	Pareto etkin çözümler	151
20	Genetik Algoritma akış şeması	152
21	Tabu Arama algoritması akış şeması	153
22	HMA ortalama sapma ve ortalama çözüm süreleri	159
23	HMA ile türetilen 10, 20 ve 50 noktalı Pareto önyükler	167

TABLO LİSTESİ

1	Test Problemlerinde Kullanılan Mesafe Değerleri (metre)	21
2	Test Problemlerinde Kullanılan Parametre Değerleri	22
3	Eklemeli-Eşit Mesafeler için Sonuçlar	23
4	Sabit Mesafeler için Sonuçlar	24
5	Genel Mesafeler için Sonuçlar	25
6	Eklemeli Genel Mesafeler için Sonuçlar	27
7	Özet Sonuçlar	28
8	S_1 çevrimi için farklı \overline{Ct} Seviyeleri için En İyi Hız ve Enerji Tüketimi	28
9	S_2 çevrimi için farklı \overline{Ct} Seviyeleri için En İyi Hız ve Enerji Tüketimi	29
10	MINLP - MISOCP Karşılaştırması	46
11	META - MISOCP Çevrim Zamanı Karşılaştırması	64
12	META - MISOCP Enerji Tüketimi Amacı Açısından Karşılaştırması	65
13	META'nın Enerji Tüketimi Amacı Açısından 5-Makinelik Sistemlerde Performans Analizi	73
14	META'nın Enerji Tüketimi Amacı Açısından 10-Makinelik Sistemlerde Performans Analizi	80
15	Enerji Tüketimi En Küçükleme Amacının Sistemin Enerji Tasarrufuna Etkisi	83
16	Parametre Değerleri	97
17	MISOCP ve MINLP Model Çözüm Süreleri	98
18	2 makine, ÖT, DV veri kümesi için çözüm süreleri	100
19	2 makine, ÖT, DV veri kümesi için çözüm süreleri	101
20	3 makine, GT, YV veri kümesi için çözüm süreleri	102
21	3 makine, GT, YV veri kümesi için çözüm süreleri	104
22	2 makine, SA ve MIQCP karşılaştırması	118
23	2 makine, ETA ve MISOCP karşılaştırması(Devam)	119
24	Veri seti 11, ETA ve MISOCP karşılaştırması	120
25	3 makine, ETA ve MISOCP karşılaştırması	121
26	3 makine, ETA ve MISOCP karşılaştırması(Devam)	122
27	5 makine, ETA ve MISOCP karşılaştırması	123
28	Tek ve Çift Tutuculu Robotun Bulunduğu Farklı Sistemlerin Enerji Tüketimi Açısından Karşılaştırması	126
29	$n=2$ için MINLP2 ve MISOCP'e ait Test Sonuçları	148
30	$n=3$ için MINLP2 ve MISOCP'e ait Test Sonuçları	148

31	HMA parametre kalibrasyonu için kullanılan deęerler	158
32	HMA parametre kombinasyonları sapma deęerleri ve çözümler	160
33	Hesaplamalı çalışmada kullanılan parametre deęerleri ve özellikleri	161
34	3x3 veri kümesi için MISOCP HMA karşılaştırma sonuçları	163
35	3x3 veri kümesi için parametre bazında elde edilen sonuçlar	164
36	3x3 veri kümesi için pareto nokta bazında yüzde sapma ve çözüm süreleri	164
37	3x4 veri kümesi için MISOCP HMA karşılaştırma sonuçları	165
38	4x3 veri kümesi için MISOCP HMA karşılaştırma sonuçları	166

ÖZET

Bu çalışmada endüstride kullanımı hızlı bir şekilde yaygınlaşan, buna paralel olarak da üzerine yapılan akademik çalışma sayısı aynı hızda artan robotik hücreler konu alınmıştır. Belirli sayıda makine ve bu makinelere hizmet veren bir malzeme elleçleme robotundan oluşan seri üretim sistemlerine robotik hücre adı verilmektedir. Sistemdeki robot üretilen parçalar üzerinde herhangi bir işlem yapmamakta, makine yükleme/boşaltma ve makineler arası parça transferini gerçekleştirmektedir.

Literatürdeki çalışmaların neredeyse tamamı üretim hızının maksimize edilmesini tek amaç olarak ele almaktadır. Bu alandaki çalışmaların tamamında robotun yaptığı bütün hareketleri mümkün olan en yüksek hızda gerçekleştirdiği varsayılmaktadır. Dolayısıyla robot hareketleriyle ilgili süreler sabit birer problem parametresidir. Halbuki robotların enerji tüketimleri hareket hızlarına bağlıdır ve yüksek hızda yapılan hareketler yüksek enerji tüketimine sebep olmaktadır. Diğer taraftan, hareketlerin en yüksek hızda yapılması bazı anlarda robotun bir sonraki hareket başlayana kadar boşta beklemesine sebep olmaktadır.

Bu projede parça sıralaması, robot hareket sıralaması ve robot hareket hızlarının belirlenmesi problemleri beraberce ele alınmıştır. Proje kapsamında, i) Tek tip parça üreten tek tutuculu sistemler; ii) Tek tip parça üreten çift tutuculu sistemler ve iii) Farklı tip parça üreten tek tutuculu sistemler ayrı ayrı ele alınmıştır. Ele alınan sistemler için üretim hızı maksimizasyonu ve enerji tüketimi minimizasyonu beraberce ele alınmıştır. Dolayısıyla, her problem için 2-kriterli bir optimizasyon modeli kurulmuş ve başatlanmayan (etkin, nondominated) çözümler kümesinin belirlenmesi hedeflenmiştir.

Tek tutuculu tek tip parça üreten 2 makineli sistemler için analitik bazı sonuçlar elde edilebilmiştir. Fakat daha çok makineli sistemler veya diğer hücre kombinasyonlarında karmaşıklık hızlı bir şekilde arttığı için analitik çözüm bulunması mümkün olmamıştır. Bu problemler için matematiksel modeller geliştirilmiştir. Karma Tamsayılı Doğrusal Olmayan yapıdaki bu modeller, çözüm süresini iyileştirmek için ikinci dereceden konik programlama formülasyonları olarak yeniden modellenmiştir. Geliştirilen bütün modeller çözdürülen örnek problemlerle doğrulanmıştır.

İkinci dereceden konik modellerin çözüm sürelerini kısalttığı gözlenmekle beraber, problem büyüklüğü arttıkça bu modellerde de makul sürelerde çözümlere ulaşmak mümkün olmamıştır. Bu sebeple, ele alınan her problem tipi için makul sürelerde kaliteli çözümler veren sezgisel/metasezgisel yöntemler geliştirilmiştir. Geliştirilen sezgiseller C++ ve Java dillerinden kodlanmış ve yapılan denemelerle doğrulanmış ve kapsamlı deneysel çalışmalarla performans testleri yapılmıştır. Ayrıca, daha maliyetli olan fakat üretim hızını artırma kapasitesi daha yüksek olan çift tutuculu robotlarla tek tutuculu robotlar enerji tüketimleri ve üretim hızları açılarından birbirleriyle karşılaştırılmıştır.

Anahtar Kelimeler: Robotik hücre, çizelgeleme, üretimde enerji optimizasyonu, matematiksel programlama, sezgisel yöntemler

ABSTRACT

In this study we consider robotic cells, which became widespread in today's industry. In parallel with this trend the number of academic studies considering robotic cells increases at the same rate. Serial production systems consisting of a number of machines and a material-handling robot that serves these machines is called a robotic cell. The robot does not make any operation on the parts but loads/unloads the machines and transfers the parts between the machines.

Almost all studies in the literature consider the maximization of the throughput as the single objective. In all studies in this field it is assumed that the robot performs every activity in the highest possible speed. Therefore, it is assumed that robot move times are fixed problem parameters. However, energy consumption of the robots depends on their speeds and higher speeds mean higher consumption rates. On the other hand, using the maximum speed sometimes leads to idle times for the robot before starting the execution of the next activity.

In this project, the part sequencing, robot move sequencing problems and the determination of the robot move speeds are considered simultaneously. Within the scope of the project, i) Single gripper robotic cells producing identical parts; ii) Dual gripper robotic cells producing identical parts, and iii) Single gripper robotic cells producing multiple parts are considered separately. Maximization of the throughput rate and minimization of the energy consumption are considered simultaneously for these systems. Thus, a bi-criteria model is developed for each problem and the set of non-dominated solutions are determined.

We attained several analytical results for the single gripper robotic cells producing identical parts. However, as the complexity increases rapidly, it was not possible to attain such results for cells with higher number of machines or with different configurations. Therefore, for these systems, general mathematical programming formulations are developed. To improve the solution times, these Mixed Integer Nonlinear Programming models are reformulated as second order cone programming formulations. All models are verified by solving several test problems.

Although it is observed that the second order conic formulations reduce the solution time, it is still not possible to find optimal solutions for large problem instances in reasonable times. Hence, we developed heuristic/metaheuristic algorithms that determines high quality solutions in reasonable times. The developed heuristics are coded in C++ and Java programming languages, verified using test problems, and their performance are tested with extensive computational studies. Dual gripper robots require higher investment costs but have a higher potential for improving the throughput rate. We also compared these robots are with single gripper ones in terms of their energy consumptions and throughput rates.

Keywords: Robotic cell, scheduling, energy optimization in production systems, mathematical programming, heuristics

1. GİRİŞ

Proje kapsamında belirli sayıda seri makinenin yer aldığı robotik hücrelerdeki çizelgeleme problemi ele alınmıştır. Bu makinelerde işlenecek olan parçaların makineler arasında taşınması ve makinelerin yüklenmesi ve boşaltılması robot tarafından yapılmaktadır. Problem, üretim hızını ve enerji tüketimini eş zamanı optimize edecek şekilde parça sıralamasını, robot hareket sıralamasını ve robot hareket hızlarını belirlemektir. Robotik hücre çizelgelemesi literatüründe robot hareket sıralaması, döngüsel çizelgeler arasından belirlenmiştir. Yani, robotun aynı hareket dizisini sürekli tekrar ettiği çizelgeler ele alınmıştır. Bunun sebebi, bu döngülerin pratikte uygulamalarının ve bilgisayarda kodlamalarının kolay olması yanında, optimal çözümün bu tür çizelgeler arasında olduğunun da ispatlanmış olmasıdır. Bu proje kapsamında da döngüsel çizelgeler ele alınmıştır. Bir döngünün bir tekrarında sisteme n adet yeni parça giriyor ve sistemden n adet parça üretilmiş olarak çıkıyorsa bu döngüye n -birim döngüler denmektedir.

Proje kapsamında üç farklı problem ele alınmıştır. Her problemde amaç çevrim zamanı ve enerji tüketimi amaçları açısından başatlanmayan çözümler, diğer bir deyişle Pareto önyüz üzerinden çözümler türetmektir. Ele alınan problemler şu şekildedir:

Tek tip parça üreten tek tutuculu robotların kullanıldığı sistemler:

Bu problemde sistemdeki robotun tek tutucusu olduğu dolayısıyla da bir anda tek bir parça taşıyabildiği varsayılmıştır. Üretilen parçaların tek tip olduğu bu problemde parça sıralaması problemi ortadan kalkmaktadır. Dolayısıyla, problem, robot döngüsünün ve döngü içerisindeki hareket hızlarının belirlenmesidir. Bu problem için yapılan çalışmalar özetle şu şekildedir:

1. Proje önerisinde de belirtildiği üzere problemler ele alınırken basitten karmaşığa doğru bir yok izlenmiştir. Bu problem için öncelikle 2 makineli hücreler ve 1-birim döngüler ele alınmıştır. Problemin 1-birim döngüler ile sınırlandırılmasının nedeni, bu döngülerin literatürde en sık kullanılan döngüler olması, anlaşılması ve uygulamasının kolay olması, hız kontrolü olmayan sistemlerde optimal çözümü garanti etmeleridir. 2 makineli sistemlerde tanımlı olan iki 1-birim döngüsü ayrı ayrı ele alınmış, her birisi için optimal hareket hızları analitik olarak belirlenmeye çalışılmıştır. S2 olarak tanımlanan döngüde hız belirleme probleminin daha karmaşık olduğu görülmüş ve parametrelerin belirli değerleri için analitik çözümler sunulurken, diğer bazı durumlar için de sezgisel algoritma geliştirilmiştir.
2. Geliştirilen algoritma C++ dilinde kodlanmıştır. Hız kontrolü varsayımının enerji tüketimi açısından faydasını değerlendirmek üzere kapsamlı hesaplamalı çalışmalar yapılmıştır. Bu çalışmalarda, literatürde yapılan robotun her zaman en yüksek hızda hareket ettiği varsayımı ile hız kontrolünün yapılabildiği varsayımı altındaki çözümler karşılaştırılmıştır. Sonuçların, problem parametrelerine duyarlılığını ölçmek için farklı parametre kombinasyonları ile deneyler yapılmıştır.
3. Aynı problem için makine sayısının 2'den büyük olduğu durumlarda kesin çözüm elde etmek üzere matematiksel programlama formülasyonu geliştirilmiştir. Bu formülasyon sadece 1-birim döngüleri değil, genel n -birim döngüler arasından optimal bulacak şekilde geliştirilmiştir. Geliştirilen model Pareto önyüzde farklı farklı noktaları türetebilmek

üzere epsilon-kısıtlı yaklaşımla ele alınmıştır. Çevrim zamanı minimizasyonu amacı, bir üst limit eklenerek kısıtlara taşınmıştır. Böylece tek amaç olarak enerji minimizasyonu kullanılmıştır. Hız fonksiyonunun üstel kuvvet içermesi ve çeşitli kısıtlardaki doğrusal olmayan ifadelerden dolayı bu model Karma Tamsayılı Doğrusal Olmayan (KTDO) yapıdadır. Bu model GAMS arayüzü ile BARON ve DICOPT çözdürücüleri ile doğrulanmış ve denemeler yapılmıştır.

4. Geliştirilen KTDO formülasyonunun çözüm sürelerinin çok uzun olması nedeniyle, formülasyondaki doğrusal olmayan ifadeler yeniden ele alınarak İkinci Dereceden Konik Programlama (İDKP) formülasyonu geliştirilmiştir. Geliştirilen bu model de GAMS arayüzü ile kodlanmış ve CPLEX 12.6.2 kullanılarak doğrulanmış ve testler yapılmıştır.
5. İDKP modeli çözüm süresini kısaltsa da problem büyüklüğü arttıkça o da makul sürelerde çözüm türetememektedir. Bu durumları ele alabilmek için sezgisel bir algoritma geliştirilmiş ve C++ programlama dilinde kodlanmıştır. Geliştirilen sezgiselle birçok test problemi çözdürülmüş ve çözüm kalitesi ve çözüm süresi açısından İDKP ile elde edilen optimal çözümlerle karşılaştırılmıştır. Problem parametrelerinin çözüm süresi ve kalitesine etkisi incelenmiştir. Ayrıca, hız kontrolünün enerji tüketimine etkisi incelenmiştir.

Tek tip parça üreten çift tutuculu robotların kullanıldığı sistemler: Bu problemde de bir öncekine benzer şekilde robot hareket sıralaması ve hareket hızlarının belirlenmesi problemleri ele alınmıştır. Fakat çift tutuculu robotlar ele alınan problemin yapısını önemli derecede farklılaştırmaktadır. Dolayısıyla, bir önceki problem için geliştirilen formülasyonlar ve çözüm yöntemleri bu problem için kullanılamamaktadır. Örneğin, tek tip robotlu sistemler için yapısal robot hareket tanımları çift tutuculu sistemleri tanımlamak için yetersiz kalmaktadır. Ayrıca problem karmaşıklığı da büyük bir hızla artmaktadır. Örneğin, 2-makineli tek tutuculu hücrede sadece iki tane 1-birim döngüsü varken çift tutuculu bir hücrede bu sayı 52'ye çıkmaktadır. Bu problem için aşağıdaki çalışmalar yapılmıştır:

1. m -makineli genel problemler için kesin çözüm bulmak üzere matematiksel programlama formülasyonu geliştirilmiştir. KTDO yapısındaki bu model yine GAMS arayüzü altında kodlanmış ve BARON ve DICOPT çözdürücüleri ile bilgisayar testler yapılmıştır.
2. Doğrusal olmayan kısıtlar yeniden formüle edilerek İDKP formülasyonu geliştirilmiştir. GAMS arayüzü ile kodlanarak CPLEX 12.6.2. çözdürücüsü ile doğrulanmış ve testler yapılmıştır.
3. İDKP ile problem büyüklüğü arttıkça makul sürelerde çözüm elde edilemediğinden sezgisel bir algoritma geliştirilmiştir. Geliştirilen bu algoritma Java programlama dilinde kodlanmıştır.
4. Geliştirilen sezgiselin performansını test etmek için kapsamlı hesaplamalı çalışmalar yapılmıştır. Sezgiselden elde edilen çözümler İDKP ile elde edilen optimal çözümlerle karşılaştırılmıştır. Problem parametrelerinin çözüm süresi ve kalitesine etkisi incelenmiştir. Ayrıca, hız kontrolünün enerji tüketimine etkisi incelenmiştir.

5. Son olarak, proje önerisinde de belirtildiği üzere, tek tutuculu robotlara göre daha yüksek yatırım maliyeti gerektiren çift tutuculu robotların üretim hızı ve enerji tüketimi açılarından faydasını ortaya koymak üzere deneysel çalışmalar yapılmıştır. Tek tutuculu ve çift tutuculu robotların kullanıldığı sistemler farklı test problemleri ve farklı parametrelerle birbirleriyle karşılaştırılmıştır.

Değişik tip parça üreten tek tutuculu robotların kullanıldığı sistemler: Bu problemde öncekilerden farklı olarak parça sıralamasının da belirlenmesi gerekmektedir. Bu durum problemin karmaşıklığını önemli ölçüde artırmaktadır. Ayrıca, 1-birim döngülerinin değişik tip parça üreten sistemlerde 2 makine için bile optimallik garantisi olmadığı önceden ispatlanmıştır. Bu sebeple bu problemde genel n -birim döngüler ele alınmıştır. Yapılan çalışmalar kısaca şu şekildedir:

1. Öncekilere benzer şekilde problem için öncelikle KTDO matematiksel programlama formülasyonu geliştirilmiş ve GAMS arayüzü altında kodlanmış ve BARON ve DICOPT çözdürücüleri ile testler yapılmıştır.
2. Modeldeki doğrusal olmayan ifadeler yeniden formüle edilerek İKDP modeli geliştirilmiştir. GAMS arayüzü ile kodlanmış ve CPLEX 12.6.2 çözdürücüsü ile testler yapılmıştır.
3. Karmaşıklığı oldukça fazla olan bu problem için makine sayısı ve iş sayısının ufak artışı bile İKDP ile makul sürelerde çözüme ulaşılamamasına sebep olmaktadır. Bu sebeple parça sıralamasının Genetik Algoritma robot hareket sıralamasının ve hareket hızlarının da Tabu Arama ile belirlendiği hibrit bir sezgisel geliştirilerek C++ programlama dilinde kodlanmıştır.
4. Hibrit algoritmada yer alan algoritma parametrelerinin kalibrasyonu için kapsamlı deneysel çalışmalar yapılmış ve en uygun parametreler belirlenmiştir.
5. Hibrit metasezgiselin performans testleri İKDP formülasyonu ile karşılaştırılarak yapılmıştır. Bu amaçla akış atölyesi çizelgelemesi literatüründen alınan test problemlerine mevcut probleme özel parametreler için farklı değerler türetilerek veri kümeleri oluşturulmuştur. Bu verilerle kapsamlı hesaplamalı çalışmalar yapılmıştır.

2 makineli tek tip robotların kullanıldığı ve tek tip parça üreten sistemler üzerine yapılan çalışmalar makale olarak hazırlanmış ve alandaki en iyi dergilerden bir tanesi olan IISE Transaction dergisine gönderilmiştir. Bu taslak makale raporun ekinde de sunulmuştur. Aynı problemin m -makineli versiyonu için de taslak bir makale hazırlanmıştır. Henüz bazı eksikleri olan bu makalenin 1-2 ay içerisinde tamamlanarak SCI indeksli bir dergiye gönderilmesi hedeflenmektedir. Bu taslak da raporun ekinde sunulmuştur. Ayrıca tek tip parça üreten çift tutuculu robotlu sistemler ve değişik tip parça üreten tek tutuculu robotlu hücreler için de makaleler hazırlanacak ve gönderilecektir. Son olarak, proje kapsamında elde edilen sonuçlar YAEM 2018'de başkanlığı proje yürütücüsü ve araştırmacısı tarafından yapılacak bir oturumda 4 farklı bildiri olarak sunulacaktır.

Proje kapsamında yapılan çalışmaların detayları raporun bundan sonraki bölümlerinde sunulmuştur. Bölüm 2'de 2 makineli tek tip parça üreten tek tutuculu hücre problemi ve Bölüm

3'de aynı problemin genel m -makine versiyonu ele alınmıştır. Bölüm 4'de tek tip parça üreten çift tutuculu hücre problemi ele alınmıştır. Tek ve çift tutuculu robotların birbirleriyle karşılaştırmaları da bu bölümde yapılmıştır. Son olarak değişik tip parça üreten tek tutuculu hücre problemi Bölüm 5'de ele alınmıştır.

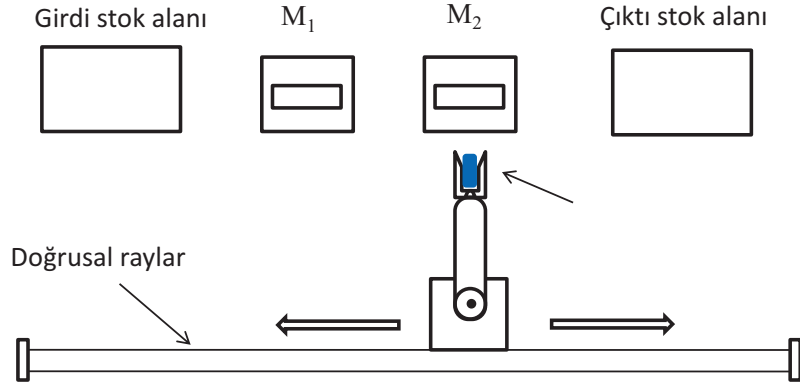
2. TEK TİP PARÇA ÜRETEK TEK TUTUCULU ROBOTLU HÜCRELERDE ROBOT HAREKET SİRALAMASININ VE HIZLARININ BELİRLENMESİ

2.1 GİRİŞ

Günümüzde, çevrim zamanının (bir parçanın uzun dönemdeki ortalama üretim zamanı), yayılma zamanının (sistemden çıkan en son parçanın tamamlanma zamanı) ve gecikme zamanının (bir ürünün tamamlanma zamanı ile teslim zamanı arasındaki fark) azaltılması özellikle seri üretim yapan firmalar için önem arz etmektedir. Söz konusu zamanların hepsi çizelgelemede farklı bir başlangıç noktası ile bitiş noktası arasında geçen zamanla ilgilidir. Bu yüzden firmalar, operasyonel verimliliklerini arttırmak ve stratejik amaçlarına ulaşmak için bu zaman kriterlerini optimize eden çizelgeler hazırlamaya çalışmaktadır.

Panek vd. [1] ile Kobetski ve Fabian'ın [2] çalışmalarında bahsedildiği gibi bir sistemdeki en iyi çizelgelemenin bulunabilmesi için sistemin çevrim zamanına odaklanması gerekir. Son yıllarda, sera gazı salınımının ortadan kaldırılması ve gelecekte daha sürdürülebilir bir çevreye sahip olunması için daha fazla çalışma ortaya konulmuştur [3, 4]. Buna paralel olarak, endüstriyel otomasyon sistemlerinde enerji tüketiminin iyileştirilmesine yönelik konulara verilen önem artmaktadır. İngiltere'deki toplam enerji tüketiminin 16%'sı fabrikalardan kaynaklanmaktadır. Bu ise 194 milyon tonluk CO₂ salınımına, bir başka deyişle 451 milyon varillik petrolün sera gazı salınımına karşılık gelmektedir [5]. Dünya çapında fabrikalardan kaynaklanan CO₂ salınımı, toplam CO₂ salınımının 36%'sını oluşturmaktadır [6]. Bütün bunların ışığında, yöneticiler için zaman odaklı amaçların iyileştirilmesinin yanında enerji tüketimiyle ilgili amaçların iyileştirilmesi de önem kazanmaktadır. Sanayi sektöründe tüketilen enerji miktarı 1998'den beri azalmıştır. Örneğin, Amerikan endüstrisindeki enerji tüketimi 2002'den 2005'e 20% [3], 2002'den 2010'a ise 17% [7] oranında azalmıştır. Buna rağmen, gereksiz tüketilen enerji miktarı 20-40% oranlarını bulabilir [8]. Bu yüzden, Avrupa Birliği'nde yer alan birçok fabrikada sürdürülebilir enerji konusuna gösterilen ilgi ve bu konuda yapılan araştırmaların sayısı giderek artmaktadır [4].

Bu çalışmada endüstride kullanımı yaygınlaşan robotik hücreler konu alınmıştır. Robotik hücre; bir girdi noktası, belirli sayıda (m) operasyon (M_1, M_2, \dots, M_m) veya makine, bir çıktı noktası ve bir elleçleme robotundan oluşan seri üretim sistemlerine verilen addır. Robotik hücrelerde tek veya çift tutuculu robotlar kullanılabilir. Firmalar, robotları kullanarak hem maliyetlerini azaltmayı hem de verimliliklerini arttırmayı amaçlar. Bu noktada, kullanılan robotların hareketleri tekrar edebilirliği, duyarlılığı, hızı, uyarlanabilirliği ve verimliliği önem kazanmaktadır [9]. Günümüzde, sera gazı salınımıyla ilgili mevcut politika ve artan enerji fiyatları neticesinde robotların harcadığı enerjinin azaltılması daha fazla önem arz etmektedir [10–13]. Seri üretim sistemlerinde ortaya çıkan enerji tüketiminin yaklaşık 8%'i robotlardan kaynaklanmaktadır [14]. Bu yüzden, robotların harcadığı enerji miktarının azaltılmasıyla operasyon maliyetleri de azaltılabilir. Öte yandan, robotik hücrelerde enerji tasarrufu yapılması da mümkündür [15–17]. Robotik hücreler; endüstri, makine, yazılım, kontrol ve otomasyon mühendisliklerinin yoğun etkileşim içerisinde olmaları sebebiyle oldukça karmaşık sistemlerdir [18, 19]. Sonuç olarak, etkin bir enerji optimizasyon yöntemi bu alanların hepsini ihtiva edecek



Şekil 1. 2 makineli tek tutuculu robotik hücre

şekilde olmalıdır. Bu tip sistemlerin performansı, enerji tüketim oranları dahil, farklı araştırma alanlarının hepsi göz önüne alınarak ölçülür [19].

Bu bölümde Şekil 1'deki gibi bir 2-makineli robotik hücreler ele alınmıştır. Burada robot doğrusal raylar üzerinde hareket eder ve sistem akış tipi (flow shop) yani her parçanın makinelerde aynı işlem sırasıyla ($M_1 - M_2$) ilerlediği üretim yapısına sahiptir. Sistemde robot makineleri yükleme/boşaltma ve makineler arası parça transferlerini gerçekleştirir. Burada robot aktivitelerinin sırası değişebilir ve verilecek kararlardan biri döngü olarak adlandırılan tekrarlanabilir aktivite sırasının belirlenmesidir. Döngüsel üretim yaklaşımında, robot makinelerin durumlarının (dolu veya boş) belli olduğu bir başlangıç pozisyonunda başlar ve belirli aktiviteleri sırasıyla yerine getirir. Sonunda robotun ve makinelerin durumu başlangıç pozisyonuna döner. Başlangıç ve bitiş pozisyonları aynı olduğundan bir döngü sonsuz sayıda tekrar edilebilir. Robotun bir döngüyü tamamlaması sonucunda n adet parçanın üretildiği döngülere n -birim döngü adı verilir. 1-birim döngüler literatürde en çok ele alınan döngülerdir; çünkü bu döngülerin uygulanması ve anlaşılması kolaydır [20]. Ayrıca, 1-birim döngüler robot hareket hızlarının sabit olduğu (değiştirilemediği) tek tip parça üreten 2-makineli ve 3-makineli robotik hücreler için en küçük çevrim zamanını verir, yani optimaldir [20, 21]. Daha fazla sayıda makine içeren robotik hücreler için de bu döngüler etkin çözümler verir [22]. Ek olarak, döngüsel üretim daha kısa değişim zamanı (change-over times), daha yüksek hat hızı (line speeds), güvenilir üretim ve teslimatlar gibi birçok farklı avantaj da sağlar [23].

Robotik hücre çizelgeleme problemlerinde genellikle robotların bütün hareketleri en yüksek hızda gerçekleştirdiği varsayılır ve amaç çıktı oranını en büyükmektir (eşdeğer olarak çevrim zamanını en küçükmektir). Ancak robotların hareketleri en yüksek hızda gerçekleştirdiği durumda enerji tüketimi de en yüksek seviyededir. Öte yandan, robotun hareketlerini en yüksek hızda gerçekleştirmesi bir sonraki aktiviteye başlamadan önce boşta beklemesine sebep olabilir. Dolayısıyla, enerji tasarrufu için kayda değer bir potansiyel mevcuttur [24]. Alternatif enerji tasarruflu ekipmanlar veya yeni enerji tasarruflu teknolojiler geliştirilerek bu kayıplar azaltılabileceği gibi sadece enerjinin aktiviteler sırasında nasıl kullanıldığının yeniden değerlendirilmesiyle de enerji tüketiminin azaltılması sağlanabilir ve bu çalışmanın odak noktası da budur.

Robotik hücreler de dahil üretim sistemlerinin çoğunda enerji tüketimi, üretim sırasındaki enerji tüketimi ve üretime yardımcı enerji tüketiminden oluşmaktadır [25]. Üretim sırasındaki enerji tüketimi; döküm, imalat veya montaj (kaynak, lehimleme veya diğer tutturma yöntemleri) gibi üretim süreçlerinde tamamen makinelerden kaynaklanan enerji tüketimidir. Üretime yardımcı enerji tüketimi (auxiliary energy) ise robot hareketlerinin gerçekleşmesi sırasındaki enerji tüketimi gibi operasyonların tamamlanması için yapılması gereken aktivitelerde kullanılan enerjidir. Bu çalışma robotun enerji tüketiminin en küçüklenmesini amaçlamaktadır [8]. Çalışmadaki robotik hücrede robot parça üzerinde herhangi bir işlem yapmaz, sadece makine yükleme/boşaltma ve makineler arası parça transferlerini gerçekleştirir. Barili vd. [26] otonom bir robot için robotun hızı değiştirilerek enerji tasarrufu yapılması konusunu ele almıştır. Bahsedilen çalışmada robotların bir kaynaktan bir veya daha fazla varış noktasına gideceği yolun/patikanın planlaması yapılırken, planlanan yol/patika ve hız kontrolü arasındaki ilişki açıklanmamıştır. Oysaki, mevcut çalışmada robotik hücrelerdeki robot aktivitelerinin sıralanmasının yanı sıra bu sıralama için en iyi hızların bulunması da yer almaktadır. Burada enerji ve işletme hedeflerinin (çevrim zamanı, yayılma zamanı vb.) her ikisini de aynı zamanda ele almak için çok kriterli optimizasyon yaklaşımı kullanılabilir.

Bu proje, hem robotun enerji tüketimini hem de çevrim zamanını en iyilemeyi amaçlayan 2-kriterli akış tipi çizelgeleme problemini ele almaktadır. Her iki amaç için farklı döngülerin enerji tüketimleri birbirleriyle karşılaştırılmakta ve en iyi çözümler belirlenmektedir. Çalışmada verilen bir çevrim zamanı üst sınırı altında, gerçekleşen farklı robot hareketleri için gereken robot hızlarına karşılık ortaya çıkan enerji tüketimi analiz edilmektedir.

2.2 LİTERATÜR ÖZETİ

Çalışmanın bu kısmında ele alınan robotik hücre, doğrusal hareket eden tek tutuculu bir robot içermektedir. Ayrıca, her aşamada bir makine bulunmakta ve makineler arasında ara stok alanı bulunmamaktadır. Projenin literatürdeki pozisyonunu görebilmek için problemin yakın ilişki içerisinde olduğu konu başlıkları ayrı ayrı ele alınacaktır. Bunlar "Robotik Hücre Çizelgelemesi", "Robot Hızları ve Enerji Tüketimi" ve "Çevrim Zamanı ve Enerji Tüketimini Ele Alan 2-Kriterli Optimizasyon" literatürleridir. Her üç alanda projeye yakından ilgili olan çalışmalar detaylandırılacak ve mevcut projenin farkları ve özgünlüğü üzerinde durulacaktır.

2.2.1 Robotik Hücre Çizelgelemesi

Robotik Hücre Çizelgeleme literatüründe incelenen problemler robotik hücrelerin özelliklerine göre farklılaşmaktadır. Bu yüzden öncelikle Dawande vd. tarafından önerilen farklı robotik hücre sınıflandırmalarına değinilmesi faydalı olabilir [27]. Bu sınıflandırmalar aşağıdaki faktörlere göre farklılaşmaktadır:

- **Makine Sayısı:** Her aşamanın tek makineden oluştuğu robotik hücrelerin çizelgenmesi basit robotik hücre çizelgenmesi olarak adlandırılırken, iş istasyonlarının en az birinde iki veya daha fazla özdeş makine içeren paralel makineli akış tipi çizelgenmesi hibrit akış tipi olarak adlandırılır. Ele alınan problemdeki hücre, Şekil 1'de verildiği gibi girdi ve çıktı noktalarının hücrenin iki ucunda yer aldığı ve her aşamanın tek makineden oluştuğu basit robotik hücredir.

• **Sistemdeki robot sayısı:** Bir robotik hücrede tek bir robot kullanılabilmesi gibi, birden fazla sayıda robot da kullanılabilir. Bu durumda, robot hareketleri çarpışma ihtimalini ortadan kaldıracak şekilde planlanmalıdır. Bu çalışmada bir robotun kullanıldığı robotik hücreler ele alınmıştır.

• **Sistemdeki robot özellikleri:** Tek tutuculu robotlar aynı anda tek bir parça tutabilen robotlar iken, çift tutuculu robotlar aynı anda iki parça tutabilen robotlardır. Öte yandan, kendi üzerinde belirli sayıda parça taşıyabilecek bir stok alanı olan robotlar ise tek tutuculu olmasına rağmen birden fazla sayıda parçayı aynı anda taşıyabilir. Bir başka robot tipi ise, her birinde tek tutucu olan ve aynı anda farklı yönlerde hareket edebilen çift kollu robotlardır. Bu çalışmada normal tek tutuculu bir robotun olduğu bir robotik hücre ele alınmaktadır. Burada robot M_{i+1} boş olmadığı sürece M_i 'yi, $\forall i = 0, \dots, m$ (m toplam makine sayısıdır. Girdi ve çıktı noktaları ise sırasıyla 0 ve $m + 1$ ile gösterilmektedir) boşaltamaz.

• **Parçaların işlem sonrası makinede bekleme (kalma) süreleri:** Parçaların işlem sonrası makinede bekleme sürelerine göre robotik hücreler üçe ayrılır: serbest (free pickup) hücreler, beklemesiz (no-wait) hücreler ve zaman pencereci (interval) hücreler. Serbest hücrelerde, işlemi tamamlanan parçanın makinede kalma süresi ile ilgili herhangi bir kısıt yoktur. Aksine, beklemesiz hücrelerdeki parça ilgili makinede işlemi bittikten hemen sonra bir sonraki makineye aktarılmalıdır. Bu tip sistemlere genellikle üretimde kullanılan parçaların belli sıcaklıklarda kalmasının gerektiği çelik üretimi ya da plastik kalıplama örnek olarak verilebilir [28–33]. Zaman pencereci hücrelerde ise, bir parçanın bir makinede kalma süresi belli bir alt ve üst limit ile belirlenir. Burada robot makinedeki işlem sona erdikten sonra belirli bir zaman aralığında makineyi boşaltmalıdır. Çalışmada da ele alınan hücre serbest ve depolama alanı içermeyen bir hücredir, yani bütün parçalar ya girdi/çıkış noktasında, ya makinelerden birinde ya da robot üzerindedir ve makinelerde işlemi biten parçanın bekleme süresi hakkında bir kısıt yoktur.

• **Makineler arası robot hareket zamanları:** Robotun makineler arasındaki hareket zamanı hücre performansını etkilemektedir. Şekil 1'deki gibi bir robotik hücre için herhangi iki makine M_i ve M_j , $0 \leq i < j \leq m + 1$, arasındaki mesafe $d(M_i, M_j)$ ile gösterilir ve $d(M_i, M_{i+1}) + d(M_{i+1}, M_{i+2}) + \dots + d(M_{j-1}, M_j)$ ile hesaplanır. Bu tip hareket zamanına toplanabilir hareket zamanı adı verilir. Eğer bütün komşu makineler, M_{i-1} ve M_i , arasındaki hareket zamanı eşit ve δ ise, herhangi iki makine M_i , M_j arasındaki hareket zamanı $d(M_i, M_j) = |i - j|\delta$ ile hesaplanır. Bir başka uygulaması ise, herhangi iki makine arasında sabit bir (δ) kadar hareket zamanı olduğu durumdur. Son olarak, öklid hareket zamanı olarak adlandırılan ve herhangi iki makine M_i ve M_j arasındaki hareket zamanının δ_{ij} ile gösterildiği durum vardır. Bu durumda, üçgen eşitsizliğinin sağlandığı varsayılır. Yani, $\delta_{ij} \leq \delta_{ik} + \delta_{kj} \forall k \in \{0, 1, \dots, m + 1\}$ sağlanır. En genel durumda bu varsayım da ortadan kalkar. Bu çalışmada, öklid hareket zamanında bahsedilen varsayımın olduğu ve olmadığı durumların her ikisi de ele alınmıştır.

• **Üretilen parça tipi sayısı:** Aynı tip parçaların üretildiği hücreler, tek tip parça üreten hücre olarak adlandırılır. Bu durumda parçaların çizelgelenmesi gerekmez, sadece robot aktiviteleri çizelgelenir. Birden fazla sayıda parça tipinin üretildiği ve parçaların makinelerdeki operasyon sürelerinin de farklılaşabildiği hücreler ise farklı tip parça üreten hücre olarak adlandırılır. Farklı tip parça üreten hücrelerde, döngüsel üretim varsayımı altında, üretilen En Küçük Parça Kümesi (MPS) belirlenir. En küçük parça kümesi, üretilen parçaların toplam

satıştaki oranına göre belirlenen en küçük üretim miktarlarından oluşan kümedir [34]. Bu kısımda tek tip parça üreten hücreler ele alınmaktadır.

Robotik hücreler hakkında yapılan çalışmalar, 1990'lardan sonra sayıları gün geçtikçe artarak literatürdeki yerini almıştır. Crama vd. [35], Hall ve Sriskindarajah [31] ve Dawande vd. [36] tarafından bu alandaki en kapsamlı çalışmalardan birkaçı yapılmıştır.

Bu alandaki ilk çalışmalardan biri Baumann vd. [37] tarafından yapılmış ve makinelerin bir robot tarafından beslendiği bir yapı için kaynak kullanımını analiz eden modeller geliştirilmiştir. Birden fazla robotun kullanıldığı robotik hücrelere ise Medeiros vd. [38] ve Nof ve Hannah [39] çalışmalarında yer vermiştir. Bu çalışmalarda çevrim zamanı ile ilgili amaçların karşılanması için benzetim yaklaşımları kullanılmıştır. Kondoleon [40] benzetim yaklaşımını kullanarak farklı yerleşim tiplerinin çevrim zamanı üzerine etkisini araştırmıştır. Claybourne [41] ise robot aktivitelerinin çizelgelenmesinin çıktı miktarına olan etkisini analiz etmiştir.

Blazewicz vd. [42] tarafından 2-makinelikli bir robotik hücrede mümkün olan iki farklı 1-birim döngü (S_1 ve S_2) tanımlanmış ve robotik hücrelerdeki çevrim zamanı hesabı için analitik yöntemler geliştirilmiştir. Ardından, Sethi vd. [20] 2 ve 3-makinelikli tek tip parça üreten hücrelerde robot aktivitelerinin çizelgelenmesi için analitik çözüm yöntemleri geliştirmiştir. Logendran ve Sriskindarajah [43] tarafından üç farklı yerleşim tipi için bu çalışma genelleştirildi: robot merkezli hücre, hareketli robotlu hücre ve doğrusal robotik hücre. Robot merkezli hücrede, makineler robotun merkezine denk geleceği çembersel bir yörünge üzerine dizilidir. Hareketli robotlu hücrede, yere veya tavana monte edilmiş bir raylı sistem gibi yardımcı bir taşıma sistemi robota destek olur. Doğrusal robotik hücrede, robot doğrusal bir taşıma sisteminde yer almaktadır. Brauner ve Finke [44–47] tarafından yapılan çalışmalarda, 1-birim döngüler diğer döngülerle karşılaştırıldı. Crama ve van de Klundert [21] toplanabilir hareket zamanının kullanıldığı tek tutuculu robot içeren ve tek tip parça üreten bir sistemde robot aktivitelerini çizelgeleme probleminin NP-Zor olduğunu gösterdi. Dawande vd. [22] ve Brauner vd. [48] ise benzer problemin sabit hareket zamanı ve öklid hareket zamanı için de NP-Zor olduğunu kanıtladı. Bahsi geçen çalışmaların hepsinde tek tip parça üreten robotik hücrelerde robot aktivitelerinin sıralanması üzerine çalışılmaktadır. Farklı tip parça üreten robotik hücrelerdeki parça çizelgeleme problemi ise, Hall vd. [49, 50] ve Sriskindarajah vd. [51] tarafından ele alınmıştır. Bu çalışmalarda genellikle serbest robotik hücreler üzerinde çalışılmış ve problemin karmaşıklığı analiz edilip, uygun kesin (mümkünse) ve sezgisel çözüm önerileri ortaya konulmuştur.

Akturk vd. [52] 2-makinelikli robotik hücrelerde bir parçanın üretimi için gerekli toplam işlem süresinin sabit olduğu fakat bunun ne kadarının hangi makinede üretileceğinin bir karar değişkeni olduğu problemi ele almıştır. Her operasyonun her makinede yapılabildiği varsayılmıştır. Takip eden çalışmada Gültekin vd. [53] ise her işlemin her makinede yapılabileceği varsayımını kaldırmış, bazı işlemlerin belirli makinelerde, diğerlerinin ise herhangi bir makinede işlenebileceği durumu ele almışlardır. Problemden amaç kalan operasyonların makinelere atanıp karşılık gelen en iyi robot hareket döngüsünün bulunmasıdır ve yapılan çalışmada en iyi çözümün 1-birim ya da 2-birim döngülerden birinde olduğu kanıtlanmıştır. Levner vd. [33] beklemez tek tip parça üreten robotik hücrelerde en iyi çevrim zamanını veren bir algoritma geliştirmiştir. Agnetis [28] tarafından 2 ve 3-makinelikli

beklemesiz hücreler üzerinde çalışmalar yapılmıştır. Agnetis ve Pacciarelli [29] 3-makineli beklemesiz robotik hücrede parça sıralama problemi için 6 mümkün döngüden 2'sinin NP-Tam (complete) olduğunu kanıtladı. Lei ve Wang [54] zaman pencereli robotik hücreler için bir dal-sınır algoritması geliştirmiştir. Chen vd. [55] ve Chen vd. [56] sırasıyla 1-birim ve n -birim döngüler ele alındığında optimal çözümleri bulmak için dal-sınır algoritması, doğrusal programlama ve iki-değerli grafikleri kullanmıştır. Sonuç olarak, 2-makineli robotik hücrelerde robotik hücre çizelgeleme problemleri polinom zamanlı çözülebilir [20]. Benzer şekilde 3-makineli tek tip parça üreten hücreler de polinom zamanda çözülebilirken, aynı hücrede farklı tip parça üretildiği durum NP-Zordur [49]. Öte yandan, m -makinelik ($m \geq 2$) bir robotik hücrede olası tüm 1-birim döngüler için parça sıralaması problemini belirli sınıflara ayırmışlardır. Bu sınıflandırmanın sonucunda olası $m!$ tane 1-birim döngü için parça sıralaması probleminin, $2m - 2$ tanesinin polinom zamanlı çözülebildiği ve geriye kalan problemlerin NP-Zor oldukları belirtilmiştir [51]. Tek tip parça üreten hücrelerdeyse 1-birim döngüler, 2 ve 3-makineli sistemler için optimaldir ve m -makinelik bir hücre için var olan en iyi 1-birim döngü de polinom zamanlı bulunabilir [20].

Robotik hücre çizelgelemesi için literatürdeki çalışmaların amacının çevrim zamanı minimizasyonu olduğu görülebilir. Oysa, sadece çevrim zamanı amaç olarak alındığında robot hızları göz önüne alınmamaktadır. Bu çalışmalarda genellikle robotların en yüksek hızlarında çalıştıkları varsayılır ve bu yüzden robot enerji tüketimi en yüksek seviyede gerçekleşir. Bu projede farklı olarak hem çevrim zamanı minimizasyonu hem de enerji tüketimi dikkate alınmaktadır

2.2.2 Robot Hızları ve Enerji Tüketimi

Son yıllarda, enerji fiyatlarındaki ve kullanımındaki artış sonucunda enerji tüketiminin azaltılması gerektiği fark edilmiştir. Bunun bir sonucu olarak da üretim sistemlerinde etkin enerji yönetimi ve enerji-etkin çizelgeleme konusundaki araştırmaların sayısı da giderek artmıştır.

Rehman vd. [57] tarafından robotların kullanıldığı operasyonlar da harcanan enerjinin azaltılmasının önemine değinilmiş ve robot çalışma alanında mevcut kısıtlarla baş edecek en iyi yol/patika planı bulunarak enerji tüketimi minimize edilmiştir. Bryan vd. [58] verilen bir zaman için enerji tüketimini azaltan optimal robot hız ve ivmelerini bulan bir yöntem geliştirmiştir. Smetanova [59] da hız, ivme, taşınan yükün ağırlığı gibi faktörlerin tüketilen enerji miktarına olan etkisini incelemiş ve robotun optimal hızı, ivmesi ve ani hareketleri (jerk) üzerinde çalışmalarda bulunmuştur. Meike ve Ribickis [60] ise enerji tüketiminin yaklaşık 8%'inin robotlardan kaynaklandığı otomotiv sektörünü baz alarak bu konuda daha kapsamlı bir çalışma yapmış ve robotların enerji tasarrufu yapmasının sağlanması için bazı yöntemler öne sürmüştür. Şu ana kadar bahsedilen tüm çalışmalarda mekatronik açıdan enerji tasarrufu yapılabilmesi için bir yol/patika planlanması (path planning programming) üzerinde çalışılmıştır. Bu çalışmalardan farklı olarak projemizde farklı robot aktiviteleri çizelgenirken her bir döngüde tanımlanan aktiviteler için gerekli robot hareket hızları düşürülerek enerji tüketiminin minimizasyonu amaçlanmaktadır.

Pellicciari vd. [61] tarafından toplama ve yerleştirme robotları ele alınarak hareket

planlamada (motion planning) sabit zamanlı ölçeklendirme yoluyla önceden planlanmış yollardan başlanıp toplam enerji tüketiminin azaltılmasına çalışılmıştır. Vergnano vd. [62] ise, birden fazla robotun yer aldığı üretim sistemlerinde enerji tasarrufu konusunu ele alarak önceki çalışmayı geliştirmiştir. Ele alınan sistemde iş istasyonlarında bulunan birden fazla robot parça üzerinde belirli sayıda işlem gerçekleştirmektedir. Robotların önceden belirlenmiş bir yol üzerindeki hız değişimleri ve ivmeleri belirlenmiştir. Önceki paragrafta da belirtildiği gibi, bahsedilen iki çalışmada da bir noktadan başka bir noktaya yol/patika planlaması için bazı algoritmalar ortaya atılmıştır. Kobetski ve Fabian [63] tarafından birden çok robotun bulunduğu bir sistemde çizelgedeki boş zamanları ortadan kaldıracak şekilde hareketli robotların hızlarını ve ivmelerini azaltmak için iki farklı yöntem geliştirilmiştir. Önerilen yöntemlerde, sistem tümüyle ele alınmış ve enerji tüketimi göz önüne alınmadan elde edilen çözüm üzerinde değişiklikler yapılarak enerji tasarrufu yapılması amaçlanmaktadır. Bahsedilen çalışmanın aksine, bu çalışmada çevrim zamanı en iyi değerinde iken bir çizelgeye ait enerji tüketimi azaltılmaya çalışılmıştır.

Paralel makineli [64–67], akış tipi [68–70] ve hibrit akış tipi [71–74] gibi farklı üretim tiplerindeki enerji tüketimini konu alan çok sayıda çalışma bulunmaktadır. Ancak, bu çalışmaların hepsinde bu projedeki gibi hem enerjiyle hem de zamanla ilgili amaçlar beraber ele alınmamakta, sadece enerji ile ilgili amaçlara odaklanılmaktadır. Ayrıca, bahsedilen çalışmaların hiçbirinde robotik hücreler ele alınmamıştır. Bundan dolayı, bahsedilen çalışmaların sonuçları hakkında daha fazla detaya yer verilmeyecektir. Bu konuda detaylı bilgi için Gahm vd. [75] ve Giret vd. [76] tarafından yapılmış iki çalışma incelenebilir.

2.2.3 Çevrim Zamanı ve Enerji Tüketimini Ele Alan 2-Kriterli Optimizasyon

Literatürde hem çevrim zamanı hem de enerji tüketimi ile ilgili amaçları birlikte ele alan birkaç çalışma bulunmaktadır. Fang ve Lin [77] paralel makinelerin çizelgelenmesi probleminde hem gecikme cezalarını hem de güç tüketim maliyetlerini minimize etmeye çalışmıştır. Burada makinelere işler atanmış ve her iş-makine eşleşmesi için optimal frekanslar belirlenmiştir. Mansouri vd. [68] 2-makineli akış tipi çizelgeleme problemlerinde yeşil çizelgeleme kavramını tanımlamış ve çevrim zamanı ile enerji tüketimi arasındaki optimal ödünleşim miktarını belirleyen bir sezgisel algoritma geliştirmiştir. Hibrit akış tipi için, tamamlanma zamanını minimize ederken enerji-etkinliğini arttıracak bir karınca koloni algoritması Du vd. [78] tarafından ortaya atılmıştır. Dai vd. [79] bir akıştaki enerji tüketimini minimize etmek için tavlama benzetimi ve genetik algoritmasını bir arada kullanarak bir meta-sezgisel yaklaşım geliştirmiştir. Subai vd. [80] yüzey işlemleri için Vinç Çizelgeleme Problemi'nde (Hoist Scheduling Problem) enerji tüketiminin azaltılması üzerine çalışmıştır. Vinç Çizelgeleme Problemi, iş istasyonları arasında taşıma yapan vinçlerin çizelgelenmesini ele alan problemlerdir. Yapılan çalışmada sistemin verimliliği belirli bir seviyede tutulurken vinçlerin boş zamanları minimize edilmiştir. Zhang vd. [81] esnek üretim sistemlerinde enerji tüketimini azaltırken, çizelgeleme etkinliğini arttıran bir hedef programlama yaklaşımı ortaya atmıştır. Mouzon ve Yildirim [82] toplam gecikmenin (tardiness) ve toplam enerji tüketiminin minimizasyonunu amaçlayan 2-kriterli optimizasyon problemini ele almıştır. Daha sonraki çalışmalarında [83] ise, aynı problemi

enerji tüketimi ve üretim zamanı minimizasyonu amaçlarıyla tekrar ele almıştır. Liu vd. [84] tek makineli çizelgeleme probleminde toplam üretim zamanını ve CO_2 salınımını minimize etmeye çalışmıştır. Öte yandan, Gultekin vd. [85] tarafından ele alınan ve verilen bir çevrim zamanına karşılık üretim maliyetlerini en küçükleyen 2-kriterli modeller de mevcuttur. Burada robotik hücrede kullanılan CNC makinelerinin hızlarının belirli maliyetlere katlanılarak artırılıp azaltılabileceği (dolayısıyla işlem zamanlarının artırılıp azaltılabileceği); ama robot hızlarının değiştirilemediği varsayılmaktadır. Halbuki, şimdi ele alınan problemde robot hızlarının kontrol edilebildiği varsayılmakta ve verilen bir çevrim zamanına karşılık toplam enerji tüketiminin en küçüklenmesi amaçlanmaktadır. Bir başka 2-kriterli optimizasyon probleminde ise, Gultekin vd. [86] 2-makineli bir robotik hücrede çevrim zamanı ve üretim maliyetlerinin optimizasyonunu ele almıştır. Bahsedilen çalışmada, optimal çevrim zamanı için mevcut üretim maliyetleri en küçüklenmektedir. Akturk ve İlhan [87] yaptıkları çalışmada inceledikleri kontrol edilebilir işlem süresine sahip tek bir CNC makine çizelgeleme probleminde, toplam gecikme ile takım ve işlem maliyetlerini en küçükleme amaçlarını birlikte ele almıştır. Uruk vd. [88] 2-makineli robotik hücrelerde işlemlerin optimal atamasını ve makinelerin işlem sürelerini belirlemiştir. Böylece, toplam üretim maliyeti ve çevrim zamanı en küçüklenmiştir.

Sonuç olarak, literatürdeki çalışmalarda robotik hücredeki çıktı oranının en büyüklenmesi (çevrim zamanı minimizasyonu) ve robotun hızlarına karşılık ortaya çıkan enerji tüketiminin en küçüklenmesi amaçları birlikte ele alınmamıştır.

2.3 PROBLEM TANIMI VE MATEMATİKSEL MODELLER

Bu çalışmada, robotik hücredeki çevrim zamanının ve robotun enerji tüketiminin minimize edilmesi amaçlarını birlikte ele alan bir problem yer almaktadır. Problemden ele alınan iki amaç birbirleriyle çelişmektedir; yani, amaçlardan birinin iyileştiği durumda diğer amaç kötüleşmektedir. Diğer bir ifadeyle, çevrim zamanının düşürülmesi için daha fazla enerji tüketmek ve enerji tüketiminin azaltılması için çevrim zamanının daha yüksek olması gerekmektedir.

İki kriterli problemin çözümünde ϵ -kısıt yöntemi kullanılmıştır. Bu yöntemde; amaçlardan biri, bir üst sınır değeriyle modele kısıt olarak yazılır. Farklı üst sınır değerleri denenerek, başatlanmayan (etkin, nondominated) çözümler kümesi oluşturulur. Bu çalışmada da, çevrim zamanıyla ilgili amaç kısıt olarak ele alınmıştır. Dolayısıyla, ele alınan problemde çevrim zamanının belirli bir üst sınır tarafından kısıtlandığı durumda robotun enerji tüketiminin en küçüklenmesi amaçlanmaktadır.

İlerleyen bölümlerde, ilk olarak çevrim zamanı ve robotun enerji tüketim fonksiyonları detaylı olarak tanımlanacaktır. Çalışmada incelenen 2-makineli robotik hücreler için iki farklı 1-birim döngüsü (S_1 ve S_2) yer almaktadır. Her iki döngü için de çevrim zamanları hesaplanabilir. Bir döngüdeki bazı robot hareketleri sırasında tutucu boş iken, diğer hareketler esnasında tutucuda bir makineye yüklenmek üzere bir parça taşınır. Sonuç olarak, robotun bir döngüdeki toplam enerji tüketimi, ilgili döngüde bulunan tüm boş ve dolu hareketler sırasındaki enerji tüketimlerinin belirli sabitlerle çarpılarak toplanmasıyla elde edilir. Burada robot hareketleri için kullanılan sabitler, birim mesafe başına karşılaşılan sürtünme kuvvetine ve robotun ağırlığına bağlı olarak

belirlenir. Problemdede; verilen bir döngüdeki robot aktivitelerinin sırasına uyularak, her aktivite için uygun robot hızları belirlenmeye çalışılmaktadır. Öte yandan, hem çevrim zamanı hem de robot hızı için verilen üst sınır değerleri sağlanmalıdır. Probleme ilgili detaylardan önce, problemde kullanılan notasyon aşağıda yer almaktadır:

Notasyon

Küme ve Parametreler:

$i, j = \{0, 1, 2, 3\}$: Makineler (M) için kullanılan indisler, burada 0 girdi, 3 ise çıktı noktasını temsil eder.

$h = \{e, f\}$: Robotun durumunu belirten indis: boş (e) ve dolu (f) robot hareketi

ε : Makineleri yükleme / boşaltma zamanı

$d_{i,j,h}$: Robotun M_i makinesinden M_j makinesine ($i \neq j$), h durumundayken katetmesi gereken mesafe. Burada M_i makinesi ve M_j makinesi arasındaki mesafenin, tutucunun durumundan (boş veya dolu) bağımsız olarak her zaman sabit olduğuna dikkat edilmelidir. Buna rağmen, problemdeki denklemlerin daha anlaşılır olması için bu parametrede üçüncü bir indis olarak h kullanılmıştır.

C_h : Enerji tüketim fonksiyonu için kullanılan sabit. h durumunda iken yapılan robot hareketinde, robotun ağırlığı ve birim mesafe başına katlanılan sürtünme kuvveti dikkate alınarak hesaplanır.

P_i : i makinesinde bir parçanın operasyon süresi, $i \in \{1, 2\}$

w_i : Robotun i makinesi önünde bekleme zamanı, $i \in \{1, 2\}$

\overline{Ct} : Çevrim zamanı üst sınırı

LB/UB : Robot hızları için alt / üst sınırlar

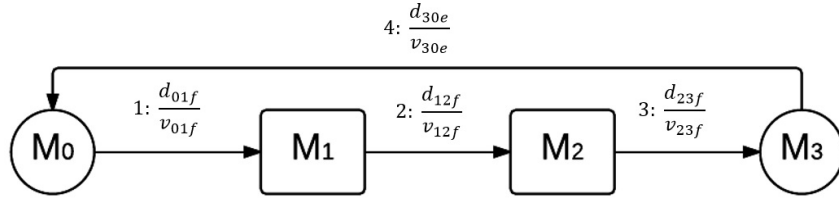
Karar Değişkenleri:

$v_{i,j,h}$: Robotun M_i makinesinden M_j makinesine ($i \neq j$), $h \in \{e, f\}$ durumunda yaptığı hareketin hızı

Yukarıda verilen notasyonlar kullanılarak takip eden bölümlerde problemdeki amaç fonksiyonları ve matematiksel model detaylandırılacaktır.

2.3.1 2-Makineli Robotik Hücreler İçin Çevrim Zamanı Hesabı

Problemdede döngülerin her zaman girdi noktasından başladığı varsayılmıştır. Burada, döngüsel bir sıralama olduğundan genellikle bir şey kaybetmeksizin yapılan varsayım doğru olur. Çalışmada, 2-makineli robotik hücrede bulunan iki farklı 1-birim döngü (S_1 and S_2) ele alınmaktadır (Sethi vd. [20]). Aşağıda her iki döngü için aktivitelerin sırası bulunmaktadır. Burada, her bir robot hareketine karşılık gelen zaman parantez içinde verilmiştir.



Şekil 2. S_1 döngüsü

S_1 : Robot girdi noktasından bir parçayı alır (ε), M_1 makinesine hareket eder $\left(\frac{d_{01f}}{v_{01f}}\right)$, M_1 makinesine parçayı yükler (ε), makine önünde parçaya yapılan operasyonun bitmesini bekler (P_1), M_1 makinesini boşaltır (ε), M_2 makinesine hareket eder $\left(\frac{d_{12f}}{v_{12f}}\right)$, parçayı bu makineye yükler (ε), M_2 makinesi önünde operasyon sona erene kadar bekler (P_2), makineyi boşaltır (ε), çıktı noktasına hareket eder $\left(\frac{d_{23f}}{v_{23f}}\right)$, parçayı çıktı noktasına bırakır (ε) ve tutucu boş durumda iken girdi noktasına geri döner $\left(\frac{d_{30e}}{v_{30e}}\right)$. Sonuç olarak, Şekil 2'de de gösterildiği gibi, verilen döngüde aktivitelerin sırasına göre 4 farklı robot hareketi mevcuttur.

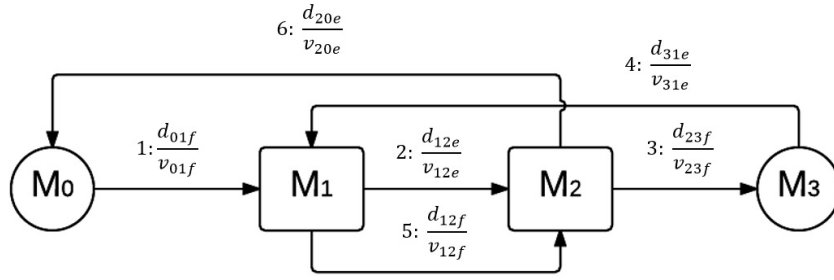
Sonuç olarak, S_1 döngüsünün çevrim zamanı:

$$Ct^{S_1} = 6\varepsilon + P_1 + P_2 + \frac{d_{01f}}{v_{01f}} + \frac{d_{12f}}{v_{12f}} + \frac{d_{23f}}{v_{23f}} + \frac{d_{30e}}{v_{30e}} \quad (1)$$

Dolu ve boş durumdaki hareketlere ait indis kümeleri sırasıyla $\mathcal{D}_1^f = \{(01f), (12f), (23f)\}$ ve $\mathcal{D}_1^e = \{(30e)\}$ şeklinde tanımlanabilir. Sonuç olarak, tüm robot hareketlerine ait indis kümesi ise $\mathcal{D}_1 = \{(01f), (12f), (23f), (30e)\}$ şeklindedir. Tanımlanan kümeler, formülasyonlarda kullanılmıştır.

S_2 : Döngünün başlangıç durumunda ikinci makine (M_2) dolu durumdadır, yani bir döngü bittiğinde ikinci makineye bir parça yüklenmiş olur. Burada robot bir başka parçayı (2. parça) girdi noktasından (M_0) alır (ε), M_1 makinesine hareket eder $\left(\frac{d_{01f}}{v_{01f}}\right)$, 2. parçayı M_1 makinesine yükler (ε), M_2 makinesine hareket eder (tutucu boş durumda iken) $\left(\frac{d_{12e}}{v_{12e}}\right)$, 1. parçanın makinedeki işlemi bitmediyse M_2 makinesi önünde bekler (w_2), 1. parçayı makineden alır (ε), çıktı noktasına hareket eder $\left(\frac{d_{23f}}{v_{23f}}\right)$, 1. parçayı çıktı noktasına bırakır (ε), M_1 makinesine hareket eder (tutucu boş durumda iken) $\left(\frac{d_{31e}}{v_{31e}}\right)$, M_1 makinesi önünde 2. parçanın işlemi bitmediyse bitmesini bekler (w_1), 2. parçayı alır (ε), M_2 makinesine hareket eder $\left(\frac{d_{12f}}{v_{12f}}\right)$, 2. parçayı M_2 makinesine yükler (ε), tutucu boş durumda iken yeni bir parça (3. parça) almak için girdi noktasına (M_0) hareket eder $\left(\frac{d_{20e}}{v_{20e}}\right)$. Sonuç olarak, Şekil 3'de gösterildiği gibi, bu döngüde toplam 6 farklı robot hareketi bulunur.

S_1 döngüsünde, robotun M_1 ve M_2 makinelerinin önündeki bekleme zamanları bir parçanın makinelerdeki operasyon sürelerine (sırasıyla P_1 ve P_2) eşittir. Bununla beraber, S_2 döngüsünde robot parçayı bir makineye yükledikten sonra (makinede operasyon başladıktan



Şekil 3. S_2 döngüsü

sonra) farklı aktiviteleri yerine getirir, yani operasyon bitene kadar makine önünde beklemez. Dolayısıyla, S_2 döngüsündeki çevrim zamanı hesabı, S_1 döngüsünde bulunandan farklıdır. Sonuç olarak, S_2 döngüsünde çevrim zamanı aşağıdaki gibi hesaplanabilir:

$$Ct^{S_2} = 6\varepsilon + \frac{d_{01f}}{v_{01f}} + \frac{d_{12e}}{v_{12e}} + w_2 + \frac{d_{23f}}{v_{23f}} + \frac{d_{31e}}{v_{31e}} + w_1 + \frac{d_{12f}}{v_{12f}} + \frac{d_{20e}}{v_{20e}} \quad (2)$$

Burada w_1 ve w_2 , robotun sırasıyla M_1 ve M_2 makineleri önünde bekleme zamanıdır. S_2 döngüsünde ilk makinedeki operasyon, robotun d_{01f} hareketini gerçekleştirip parçayı M_1 makinesine yükledikten sonra (ε) başlar. Operasyon başladıktan, yani robot parçayı ilk makineye yükledikten, P_1 kadar zaman geçince operasyon sona erer. Eğer robot d_{31e} hareketiyle M_1 makinesi önüne tekrar geldiğinde operasyon sona ermemişse, robot M_1 makinesi önünde bekler. Aksi takdirde, robot makine önünde beklemeyeceğinden w_1 sıfır olur. O halde, w_1 'in hesaplanması için aşağıdaki formül kullanılır:

$$w_1 = \max \left\{ 0, P_1 - \frac{d_{12e}}{v_{12e}} - w_2 - \varepsilon - \frac{d_{23f}}{v_{23f}} - \varepsilon - \frac{d_{31e}}{v_{31e}} \right\} \quad (3)$$

Benzer olarak, $d_{12f} + \varepsilon$ sonunda robotun M_2 makinesine parçayı yüklemesiyle P_2 kadar sürecek operasyon başlar. Burada da robot M_2 makinesinin önüne d_{12e} hareketiyle tekrar geldiğinde makine önünde beklemesi gerekebilir. O zaman w_2 aşağıdaki gibi hesaplanır:

$$w_2 = \max \left\{ 0, P_2 - \frac{d_{20e}}{v_{20e}} - \varepsilon - \frac{d_{01f}}{v_{01f}} - \varepsilon - \frac{d_{12e}}{v_{12e}} \right\} \quad (4)$$

S_2 döngüsündeki dolu ve boş robot hareketlerine ait indislerin kümesi sırasıyla $\mathcal{D}_2^f = \{(01f), (12f), (23f)\}$ ve $\mathcal{D}_2^e = \{(12e), (31e), (20e)\}$ olarak tanımlanabilir. Ayrıca, S_2 döngüsündeki tüm robot hareketlerine ait indislerin kümesi ise $\mathcal{D}_2 = \{(01f), (12e), (23f), (31e), (12f), (20e)\}$ şeklindedir.

Herhangi bir döngüde, robot tüm aktiviteleri yerine getirdikten sonra, hem robot hem de robotik hücre başlangıç durumuna geri döner. Dolayısıyla, bir döngü sonsuz sayıda tekrarlanabilir. Döngülerde ilk aktivitenin değişmesi çevrim zamanını etkilemez. Bundan dolayı, çalışmada anlaşılabilirliği arttırmak için bütün döngülerin robotun girdi noktasından bir parçayı alıp ilk makineye yüklemesiyle başladığı varsayımı yapılmıştır. Ayrıca, S_1 ve S_2 döngülerinden optimal olanı belirlemek için her döngüye ait amaç fonksiyonu değerleri hesaplanıp birbiriyle

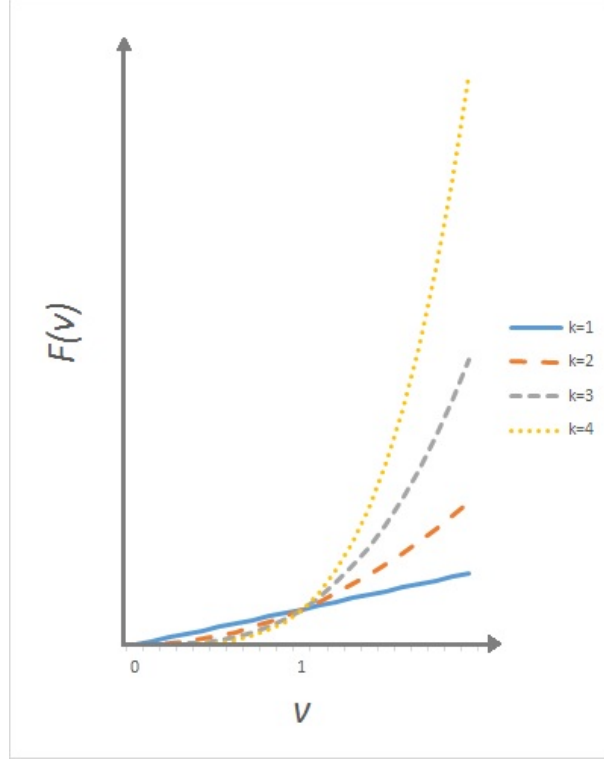
karşılaştırılmalıdır. Ele alınan problemde iki farklı amaç fonksiyonu bulunduğu için, döngülerin her iki amaç fonksiyonu da düşünülerek karşılaştırılması gerekir. Verilen sabit bir çevrim zamanı için, daha düşük enerji tüketimi olan döngü diğerinden daha iyidir. Problemde, robotun toplam enerji tüketimini en küçükleyen robot hızları belirlenecektir.

2.3.2 Robotun Enerji Tüketim Fonksiyonu

Enerji tüketim fonksiyonu, robotun katettiği mesafelere göre hesaplanır. Robot Şekil 1'deki gibi doğrusal bir rota üzerinde bir makineden bir başka makineye hareket eder. Problemde, robotun herhangi iki makine arasında yaptığı hareketin hızının sabit olduğu ve bir kez belirlendiği varsayılmıştır (hızlanma ve yavaşlamalar göz ardı edilebilir). Bununla beraber, farklı makine çiftleri için belirlenen robot hızları aynı olmak zorunda değildir. Yükleme/boşaltma sürecinde tüketilen enerji miktarı, robot hareketleri sırasındaki enerji tüketiminin yanında göz ardı edilebilecek bir seviyededir. Robotun toplam enerji tüketimi her bir hareketi için gereken enerji miktarları toplanarak hesaplanır. Robot hareketleri sırasında robotun tutucusu boş ya da dolu olabilir. Bu durumda, parçanın ağırlığına bağlı olarak, robotun enerji tüketimi tutucunun boş ve dolu olduğu hareketler için farklılaşabilir. Bu çalışmada daha genel olan durum ele alınmıştır ve her bir robot hareketi esnasında tüketilen enerji miktarının aşağıda verilen ifadeyle bulunduğu varsayılmıştır:

$$F(v) = Cdv^k \quad (5)$$

Burada C robotun ağırlığı ve sürtünme kuvveti hesaba katılarak belirlenmiş bir sabittir. Çalışmada, C_e ve C_f sırasıyla boş ve dolu robot hareketlerine karşılık gelen sabitler olarak kullanılacaktır. Verilen fonksiyonda; d hareket süresince katedilen mesafeyi, v hareketin hızını, k ise robotun tipine/modeline bağlı olarak değişen ve robotun hızıyla enerji tüketimi arasındaki ilişkiyi gösteren sabiti temsil eder. Yani, k farklı tip robotlar için değişiklik gösterebilir. Robotlar, hızlarını arttırabilmek için daha fazla enerjiye ihtiyaç duyarlar. Bu nedenle, robotun daha hızlı hareket etmesi robotun enerji tüketiminin artmasıyla sonuçlanır. Şekil 4'de enerji tüketim fonksiyonu verilmiştir. Bu çalışmada, $k \geq 1$ olduğu varsayılmıştır. Verilen fonksiyon, $k \geq 1$ olduğu durumda Şekil 4'de de görülebileceği gibi dışbükeydir. Takip eden bölümde, problemde kullanılan matematiksel formülasyon yer alacaktır. Problemde, doğrusal ($k = 1$) ve doğrusal olmayan ($k > 1$) amaç fonksiyonları için kullanılan yöntemler aynıdır.



Şekil 4. Enerji fonksiyonu alternatifleri

2.3.3 Matematiksel Modeller

Ele alınan problem için karşılaşılan modelin genel hali aşağıdaki gibidir:

min Robotun Enerji Tüketimi

min Ct (Çevrim Zamanı)

Subject to

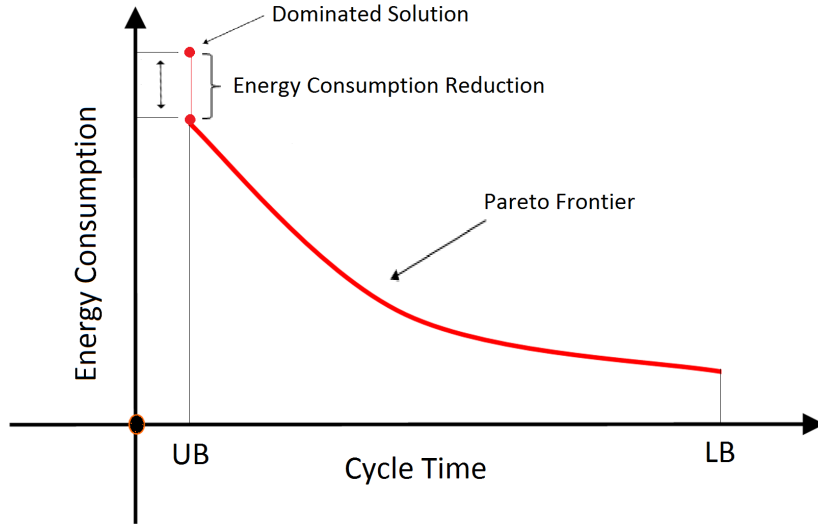
$$Ct = \min(Ct^{S_1}, Ct^{S_2})$$

$$Ct^{S_1} = 6\varepsilon + P_1 + P_2 + \frac{d_{01f}}{v_{01f}} + \frac{d_{12f}}{v_{12f}} + \frac{d_{23f}}{v_{23f}} + \frac{d_{30e}}{v_{30e}}$$

$$Ct^{S_2} = 6\varepsilon + w_1 + w_2 + \frac{d_{01f}}{v_{01f}} + \frac{d_{12e}}{v_{12e}} + \frac{d_{23f}}{v_{23f}} + \frac{d_{31e}}{v_{31e}} + \frac{d_{12f}}{v_{12f}} + \frac{d_{20e}}{v_{20e}}$$

$$LB \leq v_{ijh} \leq UB \quad \forall (ijh) \in \mathcal{D}_1 \cup \mathcal{D}_2$$

Ct^{S_1} ve Ct^{S_2} 'nin en düşük ve en yüksek değerleri, robot hızlarının sırasıyla üst (UB) ve alt (LB) sınırlara eşit olduğu durumlarda bulunur. Ayrıca, Ct (çevrim zamanı), Ct^{S_1} ve Ct^{S_2} değerlerinden küçük olana eşittir. Burada, hızlardan herhangi birinin sıfır olduğu durumda bulunan çevrim zamanı sonsuz olacağından böyle bir durumun gerçek hayatta gerçekleşmesi mümkün değildir. Dolayısıyla, robot hareket hızları için $LB > 0$ olacak şekilde bir alt sınır bulunmalıdır. ε -kısıt yöntemi kullanılarak, modelin genel halinin her bir döngü için düzenlenmiş halleri aşağıda verilmiştir:



Şekil 5. Çevrim zamanına karşın robotun enerji tüketimi

S_1 Modeli:

min Robotun Enerji Tüketimi

Subject to

$$6\varepsilon + P_1 + P_2 + \frac{d_{01f}}{v_{01f}} + \frac{d_{12f}}{v_{12f}} + \frac{d_{23f}}{v_{23f}} + \frac{d_{30e}}{v_{30e}} \leq \overline{Ct}$$

$$LB \leq v_{ijh} \leq UB \quad \forall (ijh) \in \mathcal{D}_1$$

S_2 Modeli:

min Robotun Enerji Tüketimi

Subject to

$$6\varepsilon + w_1 + w_2 + \frac{d_{01f}}{v_{01f}} + \frac{d_{12e}}{v_{12e}} + \frac{d_{23f}}{v_{23f}} + \frac{d_{31e}}{v_{31e}} + \frac{d_{12f}}{v_{12f}} + \frac{d_{20e}}{v_{20e}} \leq \overline{Ct}$$

$$LB \leq v_{ijh} \leq UB \quad \forall (ijh) \in \mathcal{D}_2$$

Problemde ele alınan amaçlar birbiriyle çeliştiği için, \overline{Ct} değerinin artırılmasının enerji tüketimini düşüreceği aşikârdır. Öte yandan, robot hızları üst ve alt sınır değerleri ile kısıtlanmaktadır. Sonuç olarak, en düşük ve en yüksek enerji tüketim değerleri bahsedilen iki uç noktada bulunur. Ayrıca, olurlu çözümler de ilgili aralıkta yer alır (Şekil 5). Bununla birlikte, aynı çevrim zamanını sağlayan ve daha az enerji tüketen çözümler, diğer çözümlere baskın gelebilir (dominated). Örneğin, bir robot hareket hızının üst sınırına eşit olduğu durum ele alınırsa, robotun makine önünde bekleme süresinin olacağı açıktır. Bu yüzden, robotun hızı azaltılarak makinenin önüne daha geç gelmesi ve beklediği sürenin azalması sağlanabilir. Böylece, bu robot hareket hızının azaltılmasıyla hareket esnasında tüketilen enerji miktarı ve dolayısıyla toplam enerji tüketimi azalacaktır. Yani, robotik hücrede aynı çevrim zamanı robotun daha düşük hızlarla hareket ettiği durumda da gerçekleşebilir.

Daha önce de bahsedildiği üzere, 2-makineli robotik hücrede iki farklı olası 1-birim döngüsü (S_1 ve S_2) yer alır ve bu Sethi vd. tarafından kanıtlanmıştır [20]. Ayrıca, Sethi vd. 1-birim döngülerin tek tip parça üreten 2-makineli robotik hücrelerde optimal olduğunu kanıtlamıştır. Döngülerde yer alan hareketlere ait indis kümeleri farklı olduğundan, problemde döngüleri ele alan iki farklı model oluşturulmuştur. Oluşturulan iki farklı formülasyon aşağıda sırasıyla yer almaktadır.

S_1 Döngüsü İçin Oluşturulan Model:

S_1 döngüsü için çevrim zamanı $6\varepsilon + P_1 + P_2 + \frac{d_{01f}}{v_{01f}} + \frac{d_{12f}}{v_{12f}} + \frac{d_{23f}}{v_{23f}} + \frac{d_{30e}}{v_{30e}}$ formülüyle hesaplanmaktadır. Buna göre, çevrim zamanının verilen bir üst sınırla kısıtlandığı durumda toplam enerji tüketimini en küçüklemeyi amaçlayan model aşağıdaki gibidir:

Formülasyon 1 (S_1 Modeli)

$$\text{Minimize } C_f \left[d_{01f} v_{01f}^k + d_{12f} v_{12f}^k + d_{23f} v_{23f}^k \right] + C_e \left[d_{30e} v_{30e}^k \right] \quad (6)$$

Subject to

$$6\varepsilon + P_1 + P_2 + \frac{d_{01f}}{v_{01f}} + \frac{d_{12f}}{v_{12f}} + \frac{d_{23f}}{v_{23f}} + \frac{d_{30e}}{v_{30e}} \leq \overline{Ct} \quad (7)$$

$$LB \leq v_{ijh} \leq UB \quad \forall (ijh) \in \mathcal{D}_1 \quad (8)$$

Burada, (6) numaralı denklem amaç fonksiyonudur. Amaç fonksiyonunda, boş ve dolu hareketlere ait sabitler sırasıyla C_e ve C_f olarak adlandırılmıştır. ε -kısıt yöntemine göre, problemin çözülmesiyle bulunan çevrim zamanı verilen çevrim zamanı üst sınırını (\overline{Ct}) geçemez. Bunu sağlamak için ilk kısıt (7) yazılmıştır. Son olarak, herhangi bir robot hareket hızı alt ve üst sınırların arasında olmak zorundadır (8). Problemde hızlar için verilen üst ve alt sınırların hareketlere göre farklılık göstermediği varsayılmıştır.

Burada, verilen (7)'daki \overline{Ct} değeri ε , P_1 , ve P_2 değerlerine bağlı olarak değişen bir eşik değerinden daha düşük ise, bu kısıt hiçbir zaman gerçekleşemez. Yani, S_1 döngüsü için \overline{Ct} değerine karşılık gelecek olurlu bir çözüm bulunamaz. Öte yandan, robot hızının alt ve üst sınırları teknolojik şartlara bağlıdır ve hızları sonsuza kadar arttırmak mümkün değildir. O yüzden, hız için bir üst sınır olmalıdır. Bu çalışmada, $LB = 0$ olduğu varsayılmıştır. Buna rağmen, hızların birinin sıfır olduğu durumda döngü sonlanmayacağından böyle bir çözüm olması mümkün değildir. Problemde doğrusal olmayan amaç fonksiyonu ve kısıtlar bulunduğu için, ele alınan problem Doğrusal Olmayan Programlama Problemidir (NLP). Problemin analitik çözümü için Karush-Kuhn-Tucker (KKT) koşulları uygulanmıştır.

S_2 Döngüsü İçin Oluşturulan Model:

Bu bölümde, S_2 döngüsü için oluşturulan matematiksel model yer almaktadır. Bu döngüde de, analitik çözüm bulunması için KKT koşulları kullanılarak farklı durumlar analiz edilmiş ve S_2 döngüsünde her hareket için optimal hız değerleri belirlenmiştir.

(2)'de bulunan (3) ve (4) denklemlerini kullanarak, aşağıdaki formül elde edilmiştir:

$$\begin{aligned}
& 6\varepsilon + \frac{d_{01f}}{v_{01f}} + \frac{d_{12e}}{v_{12e}} + \frac{d_{23f}}{v_{23f}} + \frac{d_{31e}}{v_{31e}} + \frac{d_{12f}}{v_{12f}} + \frac{d_{20e}}{v_{20e}} \\
& + \max \left\{ 0, P_1 - \frac{d_{12e}}{v_{12e}} - w_2 - \varepsilon - \frac{d_{23f}}{v_{23f}} - \varepsilon - \frac{d_{31e}}{v_{31e}}, \right. \\
& \left. P_2 - \frac{d_{20e}}{v_{20e}} - \varepsilon - \frac{d_{01f}}{v_{01f}} - \varepsilon - \frac{d_{12e}}{v_{12e}} \right\} \leq \overline{Ct}
\end{aligned} \tag{9}$$

Matematiksel modelde, (9) ile hesaplanan çevrim zamanı verilen çevrim zamanı üst sınırını aşmayacak şekilde, toplam enerji tüketiminin en küçüklmesi amaçlanmaktadır. Ek olarak, maksimum fonksiyonlarını doğrusallaştırmak için, aşağıdaki üç kısıt yazılmıştır:

$$\begin{aligned}
6\varepsilon + \frac{d_{01f}}{v_{01f}} + \frac{d_{12e}}{v_{12e}} + \frac{d_{23f}}{v_{23f}} + \frac{d_{31e}}{v_{31e}} + \frac{d_{12f}}{v_{12f}} + \frac{d_{20e}}{v_{20e}} &\leq \overline{Ct} \\
P_1 + 4\varepsilon + \frac{d_{01f}}{v_{01f}} + \frac{d_{12f}}{v_{12f}} + \frac{d_{20e}}{v_{20e}} &\leq \overline{Ct} \\
P_2 + 4\varepsilon + \frac{d_{23f}}{v_{23f}} + \frac{d_{31e}}{v_{31e}} + \frac{d_{12f}}{v_{12f}} &\leq \overline{Ct}
\end{aligned}$$

Sonuç olarak, oluşturulan matematiksel model aşağıda yer almaktadır:

Formulasyon 2 (S₂ Modeli):

$$\text{Minimize } C_f[d_{01f}v_{01f}^k + d_{23f}v_{23f}^k + d_{12f}v_{12f}^k] + C_e[d_{12e}v_{12e}^k + d_{31e}v_{31e}^k + d_{20e}v_{20e}^k]$$

Subject to

$$6\varepsilon + \frac{d_{01f}}{v_{01f}} + \frac{d_{12e}}{v_{12e}} + \frac{d_{23f}}{v_{23f}} + \frac{d_{31e}}{v_{31e}} + \frac{d_{12f}}{v_{12f}} + \frac{d_{20e}}{v_{20e}} \leq \overline{Ct} \tag{10}$$

$$P_1 + 4\varepsilon + \frac{d_{01f}}{v_{01f}} + \frac{d_{12f}}{v_{12f}} + \frac{d_{20e}}{v_{20e}} \leq \overline{Ct} \tag{11}$$

$$P_2 + 4\varepsilon + \frac{d_{23f}}{v_{23f}} + \frac{d_{31e}}{v_{31e}} + \frac{d_{12f}}{v_{12f}} \leq \overline{Ct} \tag{12}$$

$$LB \leq v_{ijh} \leq UB \quad \forall (ijh) \in \mathcal{D}_2 \tag{13}$$

Amaç fonksiyonunda, her bir hareket için oluşan enerji tüketimi ona karşılık verilen sabitle çarpılarak toplam enerji tüketimi hesaplanmıştır. (10)-(12) kısıtları çevrim zamanı üst sınırı için yazılan kısıtın doğrusallaştırılmış halidir. Karar değişkenleri olan robot hızları için bulunan üst ve alt sınırlar ise son kısıtta (13) ele alınmıştır. Bu problemin analitik çözümü için, yine KKT koşulları uygulanmıştır. İncelenmesi gereken toplam sekiz farklı durum bulunmaktadır. Bunlardan dördünün olurlu olmadığı ispatlanmış, kalan dört durum ise ayrıntılı şekilde incelenmiştir.

KKT koşulları kullanılarak ispatlanan öntoem ve teoremlerin ve geliştirilen çözüm yönteminin detayları, proje bursiyeri Vahid Eghbal Akhlaghi'nin ekte sunulan yüksek lisans tezinde yer almaktadır. Yine aynı tezde, geliştirilen çözüm yöntemi kullanılarak problem üzerinde detaylı hesaplamalı çalışmalar yapılmıştır. Bu çalışmaların bütün detayları ekte sunulan tezde yer almaktadır. Bir sonraki bölümde hesaplamalı çalışmalarda izlenen yöntem

ve elde edilen sonuçlar özetlenmiştir.

2.4 HESAPLAMALI DENEYLER VE ANALİZLER

Bu bölümde robot hız kontrolünün sağlayabileceği enerji tasarrufunu örnekler üzerinde hesaplamalı olarak göstermeyi amaçlıyoruz. Bunun için oluşturduğumuz test problemleri üzerinde robot hızının kontrol edilebildiği durum için tanımladığımız problemleri çözecek ve robot hızının kontrol edilemediği durumla karşılaştırmalar yapacağız. Test problemleri sonuç üzerinde etkisi olan farklı parametrelere alternatif değerler atayarak elde edildi. Bu şekilde farklı durumlarda enerji tasarrufunun nasıl etkileneceği görülmeye çalışıldı. Bu bölümde ayrıca farklı çevrim süresi değerleri için elde edilen çözümlerle etkin çözüm kümesi içinde enerji tüketiminin değişimi de gözlemlendi.

2.4.1 Kontrol Edilebilir Hız ve Sabit Hız Çözümlerinin Karşılaştırılması

Önceki bölümde verilen çevrim süresi için enerji tüketimini minimize eden matematiksel modeller önermiştik. Bu modellerin sınanacağı örnek problemleri oluştururken farklı durumları dikkate aldık. Bunlardan ilki makineler arası ve stok alanları arasındaki mesafeler için farklı durumlar dikkate alındı. Burada dört farklı durum sözkonusu eklenebilir-eşit, eklenebilir-genel, sabit ve genel mesafe durumlarıdır. Her durum için robotun toplam katettiği eşit olacak şekilde bu dört duruma uygun mesafeler türetildi.

Sabit için bir değer kümesi, Eklenebilir-Eşit için bir değer kümesi diğer iki durum için beşer değer kümesi oluşturuldu. Bu değerler Tablo 1’de verilmiştir.

Tablo 1. Test Problemlerinde Kullanılan Mesafe Değerleri (metre)

Distance	Set	d_{01f}	d_{12e}	d_{23f}	d_{31e}	d_{12f}	d_{20e}
Constant	1	2	2	2	2	2	2
Additive-identical	1	1.5	1.5	1.5	3	1.5	3
Additive-general	1	2	1	2	3	1	3
	2	1	2	1	3	2	3
	3	1	1	3	4	1	2
	4	3	1	1	2	1	4
	5	0.5	2.5	0.5	3	2.5	3
General	1	1	1	3	1	5	1
	2	2	2	2	3	2	1
	3	4	1	2	1	2	2
	4	0.5	2	0.5	2	0.5	6.5
	5	1	3	1.5	3.5	2.5	0.5

Üretim hücresinde makineler arası mesafeler dışında işlem süreleri de problemin sonucunu etkileyen önemli bir etkidir. Tablo 2’de test problemlerinde kullanılan işlem süreleri verilmektedir. İşlem süreleri için üç seviye (kısa, orta, uzun) belirlenmiştir. Bunun dışında yine Tablo 2’de görüldüğü gibi enerji tüketim fonksiyonunda kullanılan k için iki farklı değer

kullanılarak fonksiyonun nonlineer davranışının az ve çok olduğu iki durumun karşılaştırılması düşünülmüştür. Tablo 2’de diğer parametre değerleri de verilmiştir. Hız üst sınırları için iki farklı değer kullanılmıştır. Bu değerler literatürdeki bazı çalışmalarda [59, 89, 90], kullanılan hız değerleri (0.05 m/s - 2.2 m/s) düşünülerek belirlenmiştir. Enerji tüketim fonksiyonundaki çarpanlar C_e, C_f için iki durum dikkate alınmıştır. İlkinde bu çarpanların eşit olduğu, ikincisinde robotun boş hareket ettiği durumda daha düşük bir çarpan değeri kullanılmıştır.

Tablo 2. Test Problemlerinde Kullanılan Parametre Değerleri

Parametre	Değer
P_i	$P_1 = 1, P_2 = 3$
	$P_1 = 10, P_2 = 10$
	$P_1 = 22, P_2 = 19$
k	2
	3
UB	1
	2
C_h	$C_e = C_f = 3$
	$C_e = 2, C_f = 4$

Yukarıda verilen parametre değerleri kullanılarak oluşturulan problemler için her S1 ve S2 çevrimleri için önerilen matematiksel formülasyonlar çözülmüştür. Modellerdeki değişken ve kısıt sayıları çok az olduğundan modeller GAMS/MINOS ile ihmal edilebilecek kadar kısa sürelerde çözülmüştür.

Öncelikle robot hareketlerinde hızın üst limite eşit olduğu düşünülerek S1 ve S2 çevrilerinin verdiği en iyi çevrim süresi bulunmuştur. Pratikte genellikle robotlar en yüksek hızda çalışmaktadır. Daha sonra bulunan çevrim süresi için önerdiğimiz formülasyonlar çözülerek bu çevrim süresini veren en iyi çevrim ve o çevrimdeki robot hızları bulunmuştur. Elde edilen çözümde eğer daha düşük bir enerji tüketimi elde edilmişse robotun üst limitte hareket etmediği durumlar vardır ve hız kontrolü ile enerji tasarrufu sağlanmış demektir.

Tablo 3’da eklemeli-eşit mesafelerin olduğu durumlar için hesaplama sonuçları verilmektedir. Bu tabloda Tablo 2’de verilen alternatif parametre değerleri için toplam 24 problemin çözümü verilmiştir. Tabloda ilk sütunlarda parametre değerleri verilmiştir. S_1C_t ve S_2C_t sütunlarında S1 ve S2 çevrimlerinde hız üst limitteyken elde edilen çevrim süreleri verilmektedir. w_1 ve w_2 sütunlarında robot hızı üst limitteyken makine 1 ve makine 2 önünde robotun bekleme süreleri verilmektedir. US* robot hızı üst limitteyken ortaya çıkan enerji tüketimini CS** ise önerdiğimiz modeller çözülerek elde edilen optimal enerji tüketimini vermektedir. Bu değerler üzerinden son sütunda (Dev.%) robot hızını kontrol etmenin sağlayacağı tasarruf % olarak verilmektedir. Tablo 4 mesafelerin sabit olduğu durum için sonuçları vermektedir. Benzer şekilde Tablo 5 genel mesafeler için sonuçları vermektedir. Tablo 6 ise Eklemeli-Genel mesafeler için sonuçları vermektedir. Genel ve Eklemeli-Genel için verilen sonuçlar Tablo 1’de verilen mesafe değerleri için alınan sonuçların ortalamalarını

göstermektedir.

Tablolardan da görüldüğü üzere robot hızını kontrol etmenin sağladığı enerji tasarrufu ancak robotların makinelerin önünde beklemek yerine yavaş hareket ederek daha az enerji harcamaları ile mümkün olmuştur. Eklemeli-Eşit durum için en yüksek tasarruf %47.05 düzeyinde gerçekleşmiştir. İşlem sürelerinin kısa olduğu durum için herhangi bir tasarruf elde edilmemiştir. Bunun nedeni kısa işlem süreleri nedeniyle robotun maksimum hızda gitse bile makineler önünde bekleme yapmamasıdır.

Tablo 3. Eklemeli-Eşit Mesafeler için Sonuçlar

No.	P_1	P_2	C_e	C_f	UB	k	$S_1 Ct$	$S_2 Ct$	w_1	w_2	US*	CS**	Dev. (%)
1	10	10	3	3	1	2	38	20	0	2	36	32.33	10.20
2	10	10	3	3	1	3	38	20	0	2	36	31.85	11.52
3	10	10	3	3	2	2	32	17	0	5	144	126.31	12.29
4	10	10	3	3	2	3	32	17	0	5	288	252.08	12.47
5	10	10	2	4	1	2	38	20	0	2	33	30.55	7.42
6	10	10	2	4	1	3	38	20	0	2	33	30.24	8.38
7	10	10	2	4	2	2	32	17	0	5	132	120.20	8.94
8	10	10	2	4	2	3	32	17	0	5	264	240.05	9.07
9	1	3	3	3	1	2	22	18	0	0	36	36.00	0.00
10	1	3	3	3	1	3	22	18	0	0	36	36.00	0.00
11	1	3	3	3	2	2	16	12	0	0	144	144.00	0.00
12	1	3	3	3	2	3	16	12	0	0	288	288.00	0.00
13	1	3	2	4	1	2	22	18	0	0	33	33.00	0.00
14	1	3	2	4	1	3	22	18	0	0	33	33.00	0.00
15	1	3	2	4	2	2	16	12	0	0	132	132.00	0.00
16	1	3	2	4	2	3	16	12	0	0	264	264.00	0.00
17	22	19	3	3	1	2	59	32	3	11	36	22.92	36.32
18	22	19	3	3	1	3	59	32	3	11	36	20.92	41.88
19	22	19	3	3	2	2	53	29	3	14	144	81.96	43.08
20	22	19	3	3	2	3	53	29	3	14	288	152.51	47.05
21	22	19	2	4	1	2	59	32	3	11	33	22.20	32.73
22	22	19	2	4	1	3	59	32	3	11	33	20.49	37.92
23	22	19	2	4	2	2	53	29	3	14	132	80.52	39.00
24	22	19	2	4	2	3	53	29	3	14	264	151.24	42.71

Sabit mesafe, genel mesafe ve eklemeli -genel mesafe durumları için de benzer sonuçlar gözlenmektedir. Farklı mesafe durumlarında önerdiğimiz modellerin ortalama performansları Tablo 7'de verilmektedir.

Tablo 7'te görüldüğü üzere hesaplamalı deneylerimiz sonucunda işlem süreleri yeterince kısa olduğunda robot hızlarını kontrol etmenin tasarrufa yol açmayabileceğini gözledik. İşlem süreleri uzadıkça ise elde edilecek tasarruf oranı artmakta ve ortalamada %35.53'e kadar çıkmaktadır. Verilen enerji tüketim fonksiyonları için robot hız üst limiti arttığında elde edilen tasarrufun arttığı görülmektedir. Bunun enerji tüketim fonksiyonunun doğrusal olmaması ve konveks olması ile açıklanabileceği düşünülmektedir. Enerji tüketim fonksiyonunda k değerinin artması yani fonksiyonun nonlineer davranışının artması ile önerdiğimiz modellerin sağlayacağı

Tablo 4. Sabit Mesafeler için Sonuçlar

No.	P_1	P_2	C_e	C_f	UB	k	$S_1 Ct$	$S_2 Ct$	w_1	w_2	US*	CS**	Dev. (%)
1	10	10	3	3	1	2	38	20	0	2	36	31.50	12.50
2	10	10	3	3	1	3	38	20	0	2	36	30.75	14.58
3	10	10	3	3	2	2	32	17	0	5	144	120.67	16.20
4	10	10	3	3	2	3	32	17	0	5	288	240.22	16.59
5	10	10	2	4	1	2	38	20	0	2	36	33.00	8.33
6	10	10	2	4	1	3	38	20	0	2	36	32.50	9.72
7	10	10	2	4	2	2	32	17	0	5	144	128.44	10.80
8	10	10	2	4	2	3	32	17	0	5	288	256.15	11.06
9	1	3	3	3	1	2	22	18	0	0	36	36.00	0.00
10	1	3	3	3	1	3	22	18	0	0	36	36.00	0.00
11	1	3	3	3	2	2	16	12	0	0	144	144.00	0.00
12	1	3	3	3	2	3	16	12	0	0	288	288.00	0.00
13	1	3	2	4	1	2	22	18	0	0	36	36.00	0.00
14	1	3	2	4	1	3	22	18	0	0	36	36.00	0.00
15	1	3	2	4	2	2	16	12	0	0	144	144.00	0.00
16	1	3	2	4	2	3	16	12	0	0	288	288.00	0.00
17	22	19	3	3	1	2	59	32	3	11	36	22.06	38.72
18	22	19	3	3	1	3	59	32	3	11	36	20.26	43.72
19	22	19	3	3	2	2	53	29	3	14	144	79.79	44.59
20	22	19	3	3	2	3	53	29	3	14	288	150.16	47.86
21	22	19	2	4	1	2	59	32	3	11	36	23.86	33.71
22	22	19	2	4	1	3	59	32	3	11	36	22.16	38.45
23	22	19	2	4	2	2	53	29	3	14	144	87.46	39.27
24	22	19	2	4	2	3	53	29	3	14	288	165.89	42.40

enerji tasarrufunun artacağı görülmektedir. Son olarak farklı mesafe durumlarında elde edilecek tasarruf oranları görülmektedir. Hesaplamalı deneylerde ele aldığımız problemlerde ortalama tasarruf %18.73 olarak gerçekleşmiştir. Bu da robotik hücre çizelgelemede kontrol edilebilir robot hızlarını dikkate alarak optimal hız değerlerini bulmanın robotların enerji tüketimini ciddi oranda düşürebileceğini göstermektedir. Elde edilen en yüksek tasarru %54.30 olarak gerçekleşmiştir.

2.4.2 Etkin Çözüm Kümeleri

Robotik hücre çizelgeleme problemlerinde literatürdeki çalışmalar ve pratik uygulamalar çevrim sürelerini minimize etmeye odaklanmıştır. Bunun nedeni minimum çevrim süresinin maksimum üretim hızını veriyor olmasıdır. Bu projede çevrim süresi ile birlikte robot enerji tüketimini de minimize edilmesi gereken bir kriter olarak ele alıyoruz. Önceki bölümde maksimum robot hızı ile yaptığımız karşılaştırmalarda robot enerji tüketiminin ciddi oranda düşürülebileceğini gösterdik. Bu bölümde ise farklı çevrim süresi değerleri için önerdiğimiz modelleri çözerek çevrim süresi ve enerji tüketimi hedefleri için etkin çözüm kümesini inceleyeceğiz. Seçtiğimiz çevrim süreleri için minimum enerji tüketim değerlerini bularak etkin çözümlerde hedef fonksiyon değerlerinin değişimini gözlemleyeceğiz.

Tablo 5. Genel Mesafeler için Sonuçlar

No.	P_1	P_2	C_e	C_f	UB	k	S_2	w_1	w_2	US	CS	Ave.*	Max.**
1	10	10	3	3	1	2	21.8	1.3	3.2	36	27.22	24.39	25.63
2	10	10	3	3	1	3	21.8	1.3	3.2	36	25.55	29.02	31.49
3	10	10	3	3	2	2	17.9	0.75	6.3	144	101.25	29.69	34.52
4	10	10	3	3	2	3	17.9	0.75	6.3	288	194.51	32.46	36.66
5	10	10	2	4	1	2	21.8	1.3	3.2	35.8	28.05	21.66	25.55
6	10	10	2	4	1	3	21.8	1.3	3.2	35.8	26.62	25.64	30.59
7	10	10	2	4	2	2	17.9	0.75	6.3	143.2	106.67	25.51	33.56
8	10	10	2	4	2	3	17.9	0.75	6.3	286.4	205.75	28.16	35.90
9	1	3	3	3	1	2	18	0	0	36	36.00	0.00	0.00
10	1	3	3	3	1	3	18	0	0	36	36.00	0.00	0.00
11	1	3	3	3	2	2	12	0	0	144	144.00	0.00	0.00
12	1	3	3	3	2	3	12	0	0	288	288.00	0.00	0.00
13	1	3	2	4	1	2	18	0	0	35.8	35.80	0.00	0.00
14	1	3	2	4	1	3	18	0	0	35.8	35.80	0.00	0.00
15	1	3	2	4	2	2	12	0	0	143.2	143.20	0.00	0.00
16	1	3	2	4	2	3	12	0	0	286.4	286.40	0.00	0.00
17	22	19	3	3	1	2	32.4	3.1	13.8	36	25.58	28.96	35.53
18	22	19	3	3	1	3	32.4	3.1	13.8	36	24.46	32.05	37.04
19	22	19	3	3	2	2	29.15	3	17.1	144	87.83	39.01	47.94
20	22	19	3	3	2	3	29.15	3	17.1	288	165.16	42.65	54.30
21	22	19	2	4	1	2	32.4	3.1	13.8	35.8	26.95	24.71	35.10
22	22	19	2	4	1	3	32.4	3.1	13.8	35.8	25.86	27.77	36.58
23	22	19	2	4	2	2	29.15	3	17.1	143.2	94.44	34.05	39.36
24	22	19	2	4	2	3	29.15	3	17.1	286.4	179.03	37.49	45.00

Bu kısımda çözeceğimiz problemlerde şu parametre değerlerini kullanacağız: $d_{01f} = 1d_{12e} = d_{23f} = d_{12f} = 1$, $d_{31e} = d_{20e} = 2$, $k = 3$, $\varepsilon = 1$, $C_f = 4$, $C_e = 2$, $P_1 = 13$ and $P_2 = 11$. S_1 ve S_2 çevrimleri için minimum enerji tüketimlerini inceleyeceğiz.

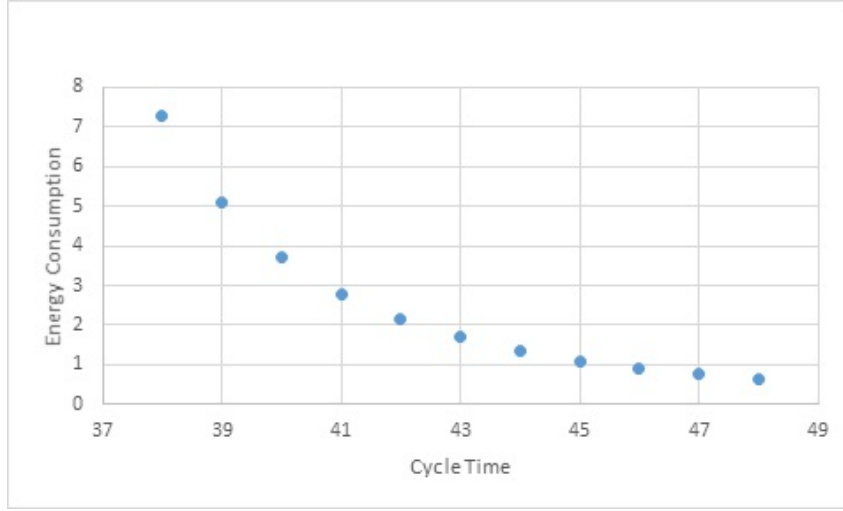
Tablo 8'da farklı çevrim süresi değerleri için robotun ayarlanması gereken hız seviyeleri ve buna karşılık gelen enerji tüketim değerleri verilmektedir. Şekil 6'de ise seçilen çevrim süreleri için çizilmiş etkin noktalar kümesi görülmektedir.

Şekil 6'de görüldüğü gibi çevrim süresi arttıkça enerji tüketimi azalmaktadır. Çevrim süresi arttıkça enerji tüketimindeki değişimin azaldığı bir başka deyişle düşük çevrim sürelerinde robot hız kararlarının enerji tüketimine etkisinin önem kazandığı görülmektedir.

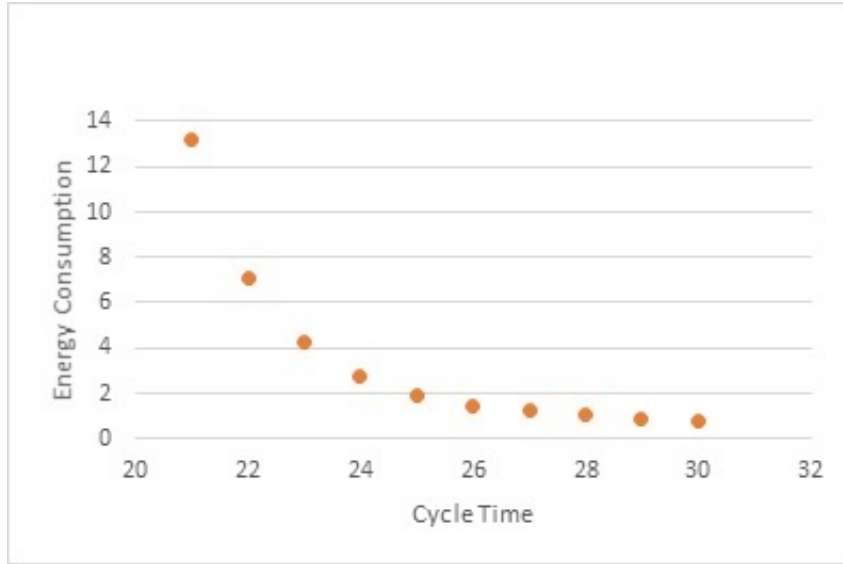
S_2 çevrimi için benzer sonuçlar Tablo 9 ve Şekil 7'de görülmektedir.

Şekil 8'de ise farklı çevrim sürelerinde S_1 , S_2 çevrimlerinin optimal enerji tüketim seviyelerini görmekteyiz. Şekilden anlaşıldığı gibi çevrim süresinin belli bir seviyesine kadar S_1 çevrimi daha sonra ise S_2 çevrimi en düşük enerji tüketimini veriyor. Bunun nedeni S_1 çevriminde robotun kat ettiği mesafenin daha kısa olması ve kısa çevrim süreleri için S_2 'nin gerektirdiği yüksek robot hızlarının yüksek enerji tüketimine neden olması.

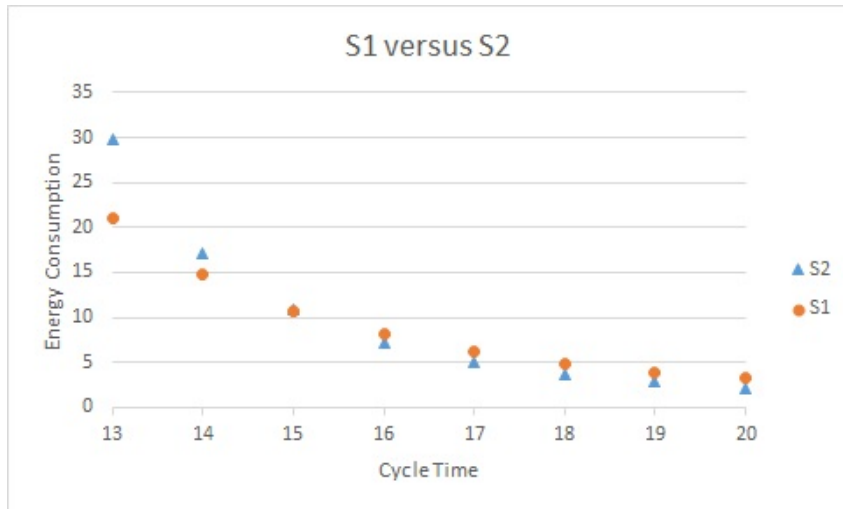
Bu kısımda iki makinalı robotik hücre çizelgeleme probleminde elde ettiğimiz başlıca sonuçları verdik. Daha detaylı sonuçlar proje bursiyeri Vahid Egbal Akhlaghi'nin ekte sunulan



Şekil 6. S_1 çevrimi için seçilen çevrim sürelerinde etkin çözüm kümesi



Şekil 7. S_2 çevrimi için seçilen çevrim sürelerinde etkin çözüm kümesi



Şekil 8. Farklı çevrim süresi değerlerinde S_1 ve S_2 döngülerinin enerji tüketim değerleri

Tablo 6. Eklemeli Genel Mesafeler için Sonuçlar

No.	P_1	P_2	C_e	C_f	UB	k	S_2	w_1	w_2	US	CS	Ave.*	Max.**
1	10	10	3	3	1	2	20.8	0.8	2	36	29.93	16.86	25.00
2	10	10	3	3	1	3	20.8	0.8	2	36	28.93	19.64	29.17
3	10	10	3	3	2	2	17.4	0.4	5	144	114.70	20.35	28.64
4	10	10	3	3	2	3	17.4	0.4	5	288	225.62	21.66	31.35
5	10	10	2	4	1	2	20.8	0.8	2	33	28.32	14.18	22.31
6	10	10	2	4	1	3	20.8	0.8	2	33	27.57	16.46	25.87
7	10	10	2	4	2	2	17.4	0.4	5	132	109.76	16.85	25.18
8	10	10	2	4	2	3	17.4	0.4	5	264	216.44	18.02	27.63
9	1	3	3	3	1	2	18	0	0	36	36.00	0.00	0.00
10	1	3	3	3	1	3	18	0	0	36	36.00	0.00	0.00
11	1	3	3	3	2	2	12	0	0	144	144.00	0.00	0.00
12	1	3	3	3	2	3	12	0	0	288	288.00	0.00	0.00
13	1	3	2	4	1	2	18	0	0	33	33.00	0.00	0.00
14	1	3	2	4	1	3	18	0	0	33	33.00	0.00	0.00
15	1	3	2	4	2	2	12	0	0	132	132.00	0.00	0.00
16	1	3	2	4	2	3	12	0	0	264	264.00	0.00	0.00
17	22	19	3	3	1	2	32.2	3.2	11	36	24.24	32.68	40.83
18	22	19	3	3	1	3	32.2	3.2	11	36	22.73	36.86	45.31
19	22	19	3	3	2	2	29	3	14	144	87.78	39.04	45.90
20	22	19	3	3	2	3	29	3	14	288	164.32	42.94	48.53
21	22	19	2	4	1	2	32.2	3.2	11	33	23.60	28.47	34.08
22	22	19	2	4	1	3	32.2	3.2	11	33	22.35	32.26	38.09
23	22	19	2	4	2	2	29	3	14	132	85.99	34.86	38.87
24	22	19	2	4	2	3	29	3	14	264	162.18	38.57	43.36

yüksek lisans tezinde verilmiştir.

Elde ettiğimiz hesaplamalı sonuçlar robot hız kararlarının özellikle kısa çevrim sürelerinde ve işlem sürelerinin uzun olduğu durumlarda daha önemli olduğunu ve %50'ler seviyesinde enerji tasarrufu sağlayabildiğini göstermektedir. Robotik üretim hücrelerinde parça elleçleme maliyetlerinin önemli oranda azaltılabileceği görülmüştür.

2.5 Sonuç

Projenin bu iş paketinde 2 makinalı, tek tip parça üreten robotik hücre ele alınmıştır. Böyle bir hücrede çevrim zamanı ve robot enerji tüketimi hedeflerinin minimize edilmesi çalışılmıştır. ϵ -kısıt yaklaşımıyla matematiksel programlama modelleri geliştirilmiştir. Modeller test için oluşturulan problemler için çözülmüştür. Test sonuçları robot hız kontrol fikrinin robot enerji tüketiminde sağlayabileceği faydaları göstermektedir. Karar vericinin seçtiği bir çevrim süresi değeri için en uygun çevrim tipi ve robotun her hareketi için en iyi hız değeri bulunabilmektedir.

Projenin bu kısmından bir master tezi kabul edilmiştir. Ayrıca bir makale yazılmış ve değerlendirme için IIE Transactions dergisine gönderilmiştir. Tez çalışması ve makalede bu rapora ek analizler ve hesaplamalı deney sonuçları yer almaktadır. Tez ve makale taslağı proje raporunun ekinde sunulmaktadır.

Tablo 7. Özet Sonuçlar

Levels	Energy saving (%)
Small processing times	0.00
Medium processing times	20.65
Large processing times	35.53
$UB = 1$	16.97
$UB = 2$	20.49
$k = 2$	17.71
$k = 3$	19.75
Additive-identical	16.71
Constant	18.86
General	20.13
Additive-general	17.91

Tablo 8. S_1 çevrimi için farklı \overline{Ct} Seviyeleri için En İyi Hız ve Enerji Tüketimi

\overline{Ct}	Enerji Tüketimi	v_{01f}	v_{12f}	v_{23f}	v_{30e}
38	7.27	0.69	0.69	0.69	0.82
39	5.10	0.61	0.61	0.61	0.73
40	3.72	0.55	0.55	0.55	0.66
41	2.80	0.50	0.50	0.50	0.60
42	2.15	0.46	0.46	0.46	0.55
43	1.69	0.43	0.43	0.43	0.51
44	1.36	0.39	0.39	0.39	0.47
45	1.10	0.37	0.37	0.37	0.44
46	0.91	0.35	0.35	0.35	0.41
47	0.76	0.33	0.33	0.33	0.39
48	0.64	0.31	0.31	0.31	0.37

Bir sonraki kısımda ikiden fazla makine için aynı hedeflerle en iyi çevrim ve robot hızlarını bulma problemi ele alınacaktır.

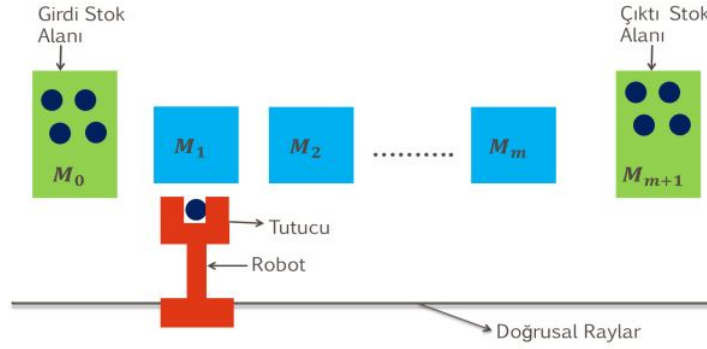
Tablo 9. S_2 çevrimi için farklı \overline{Ct} Seviyeleri için En İyi Hız ve Enerji Tüketimi

\overline{Ct}	Enerji Tüketimi	KKT Durum	v_{01f}	v_{12e}	v_{23f}	v_{31e}	v_{12f}	v_{20e}
20	30.04	4	1.22	0.15	0.64	0.76	1.22	1.45
21	13.23	4	0.91	0.17	0.54	0.65	0.94	1.09
22	7.03	4	0.73	0.19	0.47	0.56	0.76	0.87
23	4.21	4	0.61	0.22	0.42	0.50	0.64	0.72
24	2.75	4	0.52	0.26	0.37	0.45	0.55	0.62
25	1.93	4	0.46	0.32	0.34	0.40	0.48	0.54
26	1.47	2	0.41	0.38	0.32	0.38	0.41	0.49
27	1.20	2	0.37	0.38	0.32	0.38	0.37	0.44
28	1.02	2	0.34	0.38	0.32	0.38	0.36	0.40
29	0.89	1	0.31	0.37	0.31	0.37	0.31	0.37
30	0.78	1	0.30	0.36	0.30	0.36	0.30	0.36

3. TEK TİP PARÇA ÜRETEEN M-MAKİNELİ GENEL ROBOTİK HÜCRELER

Robotik hücre çizelgeleme problemleri, literatürde 1990 yılından sonra gün geçtikçe daha fazla ele alınmaya başlanmıştır. Bu alanda yapılan ilk çalışmalardan biri Sethi vd. [20] tarafından 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 -makinelili bir robotik hücre için var olan en iyi 1-birim döngünün de polinom zamanlı bulunabildiği gösterilmiştir. Ardından, Crama ve van de Klundert [21] toplanabilir hareket zamanının kullanıldığı tek tutuculu robot içeren ve tek tip parça üreten bir sistemde robot aktivitelerini çizelgeleme probleminin NP-Zor olduğunu göstermiştir. Burada bahsedilen toplanabilir hareket zamanının kullanıldığı sistemlerde, herhangi iki makine M_i ve M_j , $0 \leq i < j \leq (m + 1)$, arasındaki mesafe $\delta_{i,j}$ ile gösterilir ve $d(M_i, M_{i+1}) + d(M_{i+1}, M_{i+2}) + \dots + d(M_{j-2}, M_{j-1}) + d(M_{j-1}, M_j)$ ile hesaplanır. Aynı çalışmada, bu sistem için 1-birim döngüler arasından en iyi 1-birim döngüyü bulan $O(m^3)$ zamanlı bir algoritma da geliştirilmiştir. Robotik hücre çizelgelenmesi literatüründeki çalışmaların amacının genellikle çevrim zamanını minimize etmek olduğu görülebilir. Oysa, sadece çevrim zamanı amaç olarak ele alındığında robot hızlarının ayarlanabilir olduğu ve Chemnitz vd. [91] tarafından yapılan çalışmada belirtildiği gibi çevrim zamanına bağlı olarak karşılık gelen enerji tüketimlerinin değiştirilebileceği göz önüne alınmamaktadır. Bu çalışmalarda genellikle robotların en yüksek hızlarında çalıştıkları varsayılır ve robotların boş beklediği zamanlar oluşabilir. Bu yüzden robotun daha düşük bir enerji tüketimi ile aynı çevrim zamanını sağlaması olasıdır. Yani, robotun enerji tüketiminin düşürülmesi için bir potansiyel mevcuttur.

Bu çalışmada yer alan problemde hem sistemin çevrim zamanı hem de robotun enerji tüketimi dikkate alınmaktadır. Literatür araştırmamızın sonucuna göre, literatürde bahsedilen iki amacı birlikte ele alan ve robot hareket hızlarının karar değişkeni olduğu problemler ilk kez çalışılmaktadır. Projenin bu kısmında ele alınan problemde amaç; Şekil 3'deki gibi tek tutuculu bir robotun olduğu, tek tip parça üreten, m -makinelili robotik hücrelerde herhangi bir n değeri için n -birim döngüleri ele alarak sistemin çevrim zamanını ve robotun enerji tüketimini minimize edecek robot aktivite sıralamasını ve hareket hızlarını belirlemektir. Problemin ana kısıtları, robotun aynı anda tek aktivite yapabileceğini belirten aktivite atama kısıtları ve makinelerin doluluk/boşluklarına göre olurlu aktivite sıralamasını sağlayan kısıtlardır. Ayrıca, modelde çevrim zamanının hesaplanması ve bir makinede işlem tamamlanmadan sonraki makinede işlem başlayamamasını sağlayan sıralama kısıtları mevcuttur.



Şekil 3. m -makineli tek tip parça üreten robotik hücre

Bu problemde, robot aktivite sıralaması ve hareket hızlarının belirlenmesi için farklı yöntemler kullanılarak çözüm aranmıştır. Öncelikle, bu problem için bir matematiksel model geliştirilmiştir. Bu modelde ε -kısıtlı yaklaşım kullanılarak çevrim zamanı amacının bir üst sınırla kısıtlarda yer aldığı enerji tüketimini minimize eden bir matematiksel model kurulmuştur. Matematiksel modelde, farklı çevrim zamanı üst sınırları kullanılarak etkin çözümler kümesi elde edilmiştir. Geliştirilen bu matematiksel modelin çözüm süresinin küçük boyutlu problemler için bile oldukça uzun olması nedeniyle, ilgili modelin hızlandırılması için yeniden formüle edilmesi üzerine çalışılmıştır ve bir ikinci derece konik model oluşturulmuştur. Ancak, ikinci derece konik modelin çözüm süresi büyük boyutlu problemler için hala uzun olduğundan, kısa sürede iyi çözüm bulunabilmesi için bir sezgisel algoritma geliştirilmiştir. Bahsedilen sezgisel algoritmada etkin çözüm kümesinin ilk çözümünün bulunmasında Crama ve van de Klundert [21] tarafından geliştirilen algoritmadan da yararlanılmıştır. Crama ve van de Klundert tarafından geliştirilen bu algoritma ile verilen bir probleme ait en iyi 1-birim döngü bulunmaktadır.

Bu bölümde, bahsedilen problem için geliştirilen çözüm yöntemleri sırasıyla ele alınmış ve bu yöntemler birbirleriyle karşılaştırılarak avantaj ve dezavantajları sıralanmıştır. İlk olarak Bölüm 3.2'de çözüm için geliştirilen matematiksel model ve ikinci dereceden konik programlama yaklaşımı kullanılarak geliştirilen ikinci derece konik model detaylı olarak anlatılmıştır. Ardından, bu iki model oluşturulan test problemleri kullanılarak çözüm süresi açısından karşılaştırılmıştır. Son olarak, Bölüm 3.3'de geliştirilen sezgisel algoritmaya ait detaylar ve bu algoritmanın çözüm süresi ve çözüm kalitesi açısından farklı problemler için yapılan testlerin sonuçlarına ve bu sonuçlar doğrultusunda yapılan değerlendirmelere yer verilmiştir.

3.1 LİTERATÜR TARAMA

Çalışmanın bu kısmında ele alınan robotik hücre, doğrusal ray üzerinde hareket eden tek tutuculu bir robot ve toplam m tane makine içermektedir. Her aşamada bir makine bulunmakta ve makineler arasında ara stok alanı bulunmamaktadır. Projenin literatürdeki pozisyonunu görebilmek için problemin yakın ilişki içerisinde olduğu konu başlıkları ayrı ayrı ele alınacaktır. Bunlar "Robotik Hücre Çizelgelemesi", "Robot Hızları ve Enerji Tüketimi" ve "Çevrim Zamanı ve Enerji Tüketimini Ele Alan 2-Kriterli Optimizasyon" literatürleridir. Her üç alanda projeye yakından ilgili olan çalışmalar detaylandırılacak ve mevcut projenin farkları ve özgünlüğü

üzerinde durulacaktır.

3.1.1 Robotik Hücre Çizelgelemesi

Robotik Hücre Çizelgeleme literatüründe incelenen problemler robotik hücrelerin sahip olduğu özelliklere göre farklılaşmaktadır. Bu yüzden, öncelikle Dawande vd. tarafından önerilen farklı sınıflandırmalara değinilmesi faydalı olabilir [27]. Bu sınıflandırmalar aşağıdaki faktörlere göre farklılaşmaktadır:

- **Makine Sayısı:** Her iş istasyonunda tek bir makine bulunan ve her makinenin bir işlem gerçekleştirdiği akış tipi robotik hücrelerin çizelgelenmesi basit robotik hücre çizelgelenmesi olarak adlandırılırken, iş istasyonlarının en az birinde iki veya daha fazla özdeş makine içeren paralel makineli akış tipi çizelgelenmesi hibrit akış tipi olarak adlandırılır. Ele alınan problemdeki hücre, Şekil 3'de verildiği gibi girdi ve çıktı noktalarının hücrenin iki ucunda yer aldığı ve her aşamanın tek makineden oluştuğu basit robotik hücredir.

- **Sistemdeki robot sayısı:** Bir robotik hücrede tek bir robot kullanılabileceği gibi, birden fazla sayıda robot da kullanılabilir. Bu durumda, robot hareketleri çarpışma ihtimalini ortadan kaldıracak şekilde planlanmalıdır. Bu çalışmada bir robotun kullanıldığı robotik hücreler ele alınmıştır.

- **Sistemdeki robot özellikleri:** Tek tutuculu robotlar aynı anda tek bir parçayı üzerinde bulundurabilen robotlar iken, çift tutuculu robotlar aynı anda iki parça taşıyabilen robotlardır. Öte yandan, kendi üzerinde belirli sayıda parça taşıyabilecek bir stok alanı olan robotlar ise tek tutuculu olmasına rağmen birden fazla sayıda parçayı aynı anda taşıyabilir. Bir başka robot tipi ise, her birinde tek tutucu olan ve aynı anda farklı yönlerde hareket edebilen çift kollu robotlardır. Bu çalışmada normal tek tutuculu bir robotun olduğu bir robotik hücre ele alınmaktadır. Burada robot M_{i+1} boş olmadığı sürece M_i 'yi, $\forall i = 0, \dots, m$ (m toplam makine sayısıdır. Girdi ve çıktı noktaları ise sırasıyla 0 ve $(m + 1)$ ile gösterilmektedir) boşaltamaz.

- **Parçaların işlem sonrası makinede bekleme (kalma) süreleri:** Parçaların işlem sonrası makinede bekleme sürelerine göre robotik hücreler üçe ayrılır: serbest (free pickup) hücreler, beklemesiz (no-wait) hücreler ve zaman pencereli (interval) hücreler. Serbest hücrelerde, işlemi tamamlanan parçanın makinede kalma süresi ile ilgili herhangi bir kısıt yoktur. Aksine, beklemesiz hücrelerdeki parça ilgili makinede işlemi bittikten hemen sonra bir sonraki makineye aktarılmalıdır. Bu tip sistemlere genellikle üretimde kullanılan parçaların belli sıcaklıklarda kalmasının gerektiği çelik üretimi ya da plastik kalıplama endüstrileri örnek olarak verilebilir [28–33]. Zaman pencereli hücrelerde ise, bir parçanın bir makinede kalma süresi belli bir alt ve üst limit ile belirlenir. Burada robot makinedeki işlem sona erdikten sonra belirli bir zaman aralığında makineyi boşaltmalıdır. Çalışmada ele alınan hücre serbest ve depolama alanı içermeyen bir hücredir, yani bütün parçalar ya girdi/çıkış noktasında, ya makinelerden birinde ya da robot üzerindedir ve makinelerde işlemi biten parçanın bekleme süresi hakkında bir kısıt yoktur.

- **Makineler arası robot hareket zamanları:** Robotun makineler arasındaki hareket zamanı hücre performansını etkilemektedir. Şekil 3'deki gibi bir robotik hücre için herhangi iki makine M_i ve M_j , $0 \leq i < j \leq m + 1$, arasındaki mesafe $d(M_i, M_j)$ ile gösterilir ve $d(M_i, M_{i+1}) + d(M_{i+1}, M_{i+2}) + \dots + d(M_{j-1}, M_j)$ ile hesaplanır. Bu tip hareket zamanına toplanabilir hareket

zamanı adı verilir. Eğer bütün komşu makineler, M_{i-1} ve M_i , arasındaki hareket zamanı eşit ve δ ise, herhangi iki makine M_i, M_j arasındaki hareket zamanı $d(M_i, M_j) = |i - j|\delta$ ile hesaplanır. Bir başka uygulaması ise, herhangi iki makine arasında sabit bir (δ) kadar hareket zamanı olduğu durumdur. Son olarak, öklid hareket zamanı olarak adlandırılan ve herhangi iki makine M_i ve M_j arasındaki hareket zamanının δ_{ij} ile gösterildiği durum vardır. Bu durumda, üçgen eşitsizliğinin sağlandığı varsayılır. Yani, $\delta_{ij} \leq \delta_{ik} + \delta_{kj} \forall k \in \{0, 1, \dots, m + 1\}$ sağlanır. En genel durumda bu varsayım da ortadan kalkar. Bu çalışmada, öklid hareket zamanında bahsedilen varsayımın olduğu ve olmadığı durumların her ikisi de ele alınmıştır.

• **Üretilen parça tipi sayısı:** Aynı tip parçaların üretildiği hücreler, tek tip parça üreten hücre olarak adlandırılır. Bu durumda parçaların çizelgelenmesi gerekmez, sadece robot aktiviteleri çizelgelenir. Birden fazla sayıda parça tipinin üretildiği ve parçaların makinelerdeki operasyon sürelerinin de farklılaşabildiği hücreler ise farklı tip parça üreten hücre olarak adlandırılır. Farklı tip parça üreten hücrelerde, döngüsel üretim varsayımı altında, üretilen En Küçük Parça Kümesi (MPS) belirlenir. En küçük parça kümesi, üretilen parçaların toplam satıştaki oranına göre belirlenen en küçük üretim miktarlarından oluşan kümedir [34]. Çalışmanın bu kısmında tek tip parça üreten hücreler ele alınmaktadır.

Robotik hücreler hakkında yapılan çalışmalar, 1990'lardan sonra sayıları gün geçtikçe artarak literatürdeki yerini almıştır. Crama vd. [35], Hall ve Sriskandarajah [31] ve Dawande vd. [36] tarafından bu alandaki en kapsamlı çalışmalardan birkaçı yapılmıştır.

Bu alandaki ilk çalışmalardan biri Baumann vd. [37] tarafından yapılmış ve makinelerin bir robot tarafından beslendiği bir yapı için kaynak kullanımını analiz eden modeller geliştirilmiştir. Birden fazla robotun kullanıldığı robotik hücrelere ise Medeiros vd. [38] ve Nof ve Hannah [39] çalışmalarında yer vermiştir. Bu çalışmalarda çevrim zamanı ile ilgili amaçların karşılanması için benzetim yaklaşımları kullanılmıştır. Kondoleon [40] benzetim yaklaşımını kullanarak farklı yerleşim tiplerinin çevrim zamanı üzerine etkisini araştırmıştır. Claybourne [41] ise robot aktivitelerinin çizelgelenmesinin çıktı miktarına olan etkisini analiz etmiştir.

Blazewicz vd. [42] tarafından 2-makineli bir robotik hücrede mümkün olan iki farklı 1-birim döngü (S_1 ve S_2) tanımlanmış ve robotik hücrelerdeki çevrim zamanı hesabı için analitik yöntemler geliştirilmiştir. Ardından, Sethi vd. [20] 2 ve 3-makineli tek tip parça üreten hücrelerde robot aktivitelerinin çizelgelenmesi için analitik çözüm yöntemleri geliştirmiştir. Logendran ve Sriskandarajah [43] tarafından üç farklı yerleşim tipi için bu çalışma genelleştirildi: robot merkezli hücre, hareketli robotlu hücre ve doğrusal robotik hücre. Robot merkezli hücrede, makineler robotun merkezine denk geleceği çembersel bir yörünge üzerine dizilidir. Hareketli robotlu hücrede, yere veya tavana monte edilmiş bir raylı sistem gibi yardımcı bir taşıma sistemi robota destek olur. Doğrusal robotik hücrede, robot doğrusal bir taşıma sisteminde yer almaktadır. Brauner ve Finke [44–47] tarafından yapılan çalışmalarda, 1-birim döngüler diğer döngülerle karşılaştırıldı. Crama ve van de Klundert [21] toplanabilir hareket zamanının kullanıldığı tek tutuculu robot içeren ve tek tip parça üreten bir sistemde robot aktivitelerini çizelgeleme probleminin NP-Zor olduğunu gösterdi. Yine aynı çalışmada, tek tutuculu robot içeren ve tek tip parça üreten robotik hücreler için çevrim zamanı açısından en iyi 1-birim döngüyü bulan bir algoritma önerilmiştir. Dawande vd. [22] ve Brauner vd. [48] ise benzer problemin sabit hareket zamanı ve öklid hareket zamanı için de NP-Zor olduğunu

kanıtladı. Bahsi geçen çalışmaların hepsinde tek tip parça üreten robotik hücrelerde robot aktivitelerinin sıralanması üzerine çalışılmaktadır. Farklı tip parça üreten robotik hücrelerdeki parça çizelgeleme problemi ise, Hall vd. [49, 50] ve Sriskandarajah vd. [51] tarafından ele alınmıştır. Bu çalışmalarda genellikle serbest robotik hücreler üzerinde çalışılmış ve problemin karmaşıklığı analiz edilip, uygun kesin (mümkünse) ve sezgisel çözüm önerileri ortaya konulmuştur.

Aktürk vd. [52] 2-makineli robotik hücrelerde bir parçanın üretimi için gerekli toplam işlem süresinin sabit olduğu fakat bunun ne kadarının hangi makinede üretileceğinin bir karar değişkeni olduğu problemi ele almıştır. Her makinede tüm operasyonların yapılabildiği varsayılmıştır. Takip eden çalışmada Gültekin vd. [53] ise tüm işlemlerin her makinede yapılabileceği varsayımını kaldırmış, bazı işlemlerin belirli makinelerde, diğerlerinin ise herhangi bir makinede işlenebileceği durumu ele almışlardır. Problemden amaç kalan operasyonların makinelere atanıp karşılık gelen en iyi robot hareket döngüsünün bulunmasıdır ve yapılan çalışmada en iyi çözümün 1-birim ya da 2-birim döngülerden birinde olduğu kanıtlanmıştır. Levner vd. [33] beklemesiz tek tip parça üreten robotik hücrelerde en iyi çevrim zamanını veren bir algoritma geliştirmiştir. Agnetis [28] tarafından 2 ve 3-makineli beklemesiz hücreler üzerinde çalışmalar yapılmıştır. Agnetis ve Pacciarelli [29] 3-makineli beklemesiz robotik hücrede parça sıralama problemi için 6 mümkün döngüden 2'sinin NP-Tam (complete) olduğunu kanıtladı. Lei ve Wang [54] zaman pencereli robotik hücreler için bir dal-sınır algoritması geliştirmiştir. Chen vd. [55] ve Chen vd. [56] sırasıyla 1-birim ve n -birim döngüler ele alındığında optimal çözümleri bulmak için dal-sınır algoritması, doğrusal programlama ve iki-değerli grafikleri kullanmıştır. Sonuç olarak, 2-makineli robotik hücrelerde robotik hücre çizelgeleme problemleri polinom zamanlı çözülebilir [20]. Benzer şekilde 3-makineli tek tip parça üreten hücreler de polinom zamanda çözülebilirken, aynı hücrede farklı tip parça üretildiği durum NP-Zordur [49]. Öte yandan, m -makineli ($m \geq 2$) bir robotik hücrede olası tüm 1-birim döngüler için parça sıralaması problemini belirli sınıflara ayırmışlardır. Bu sınıflandırmanın sonucunda olası $m!$ tane 1-birim döngü için parça sıralaması probleminin, $2m - 2$ tanesinin polinom zamanlı çözülebildiği ve geriye kalan problemlerin NP-Zor oldukları belirtilmiştir [51]. Tek tip parça üreten hücrelerdeyse 1-birim döngüler, 2 ve 3-makineli sistemler için optimaldir ve m -makineli bir hücre için var olan en iyi 1-birim döngü de polinom zamanlı bulunabilir [20].

Robotik hücre çizelgelemesi için literatürdeki çalışmaların amacının genellikle çevrim zamanını en küçükmek olduğu görülebilir. Oysa, sadece çevrim zamanı en küçükmek amacını ele alındığında robot hızları göz önüne alınmamaktadır. Bu çalışmalarda genellikle robotların en yüksek hızlarında çalıştıkları varsayılır ve bu yüzden enerji tüketimi yüksektir. Bu projede farklı olarak hem çevrim zamanını hem de enerji tüketimi minimize etme hedefi dikkate alınmaktadır.

3.1.2 Robot Hızları ve Enerji Tüketimi

Son yıllarda, enerji fiyatlarındaki ve kullanımındaki artış sonucunda enerji tüketiminin azaltılması gerektiği fark edilmiştir. Bunun bir sonucu olarak da üretim sistemlerinde etkin enerji

yönetimi ve enerji-etkin çizelgeleme konusundaki araştırmaların sayısı da giderek artmıştır.

Rehman vd. [57] tarafından robotların kullanıldığı operasyonlarda harcanan enerjinin azaltılmasının önemine değinilmiş ve robot çalışma alanında mevcut kısıtlarla baş edecek en iyi yol/patika planı bulunarak enerji tüketimi minimize edilmiştir. Bryan vd. [58] verilen bir zaman için enerji tüketimini azaltan optimal robot hız ve ivmelerini bulan bir yöntem geliştirmiştir. Smetanova [59] da hız, ivme, taşınan yükün ağırlığı gibi faktörlerin tüketilen enerji miktarına olan etkisini incelemiş ve robotun optimal hızı, ivmesi ve ani hareketleri (jerk) üzerinde çalışmalarda bulunmuştur. Meike ve Ribickis [60] ise enerji tüketiminin yaklaşık 8%'inin robotlardan kaynaklandığı otomotiv sektörünü baz alarak bu konuda daha kapsamlı bir çalışma yapmış ve robotların enerji tasarrufu yapmasının sağlanması için bazı yöntemler öne sürmüştür. Şu ana kadar bahsedilen tüm çalışmalarda mekatronik açıdan enerji tasarrufu yapılabilmesi için bir yol/patika planlanması (path planning programming) üzerinde çalışılmıştır. Halbuki, bu projede farklı robot aktiviteleri çizelgelenirken her bir döngüde tanımlanan aktiviteler için gerekli robot hareket hızları düşürülerek enerji tüketiminin minimize edilmesi amaçlanmaktadır.

Pellicciari vd. [61] tarafından toplama ve yerleştirme robotları ele alınarak hareket planlamada (motion planning) sabit zamanlı ölçeklendirme yoluyla önceden planlanmış yollardan başlanıp toplam enerji tüketiminin azaltılmasına çalışılmıştır. Vergnano vd. [62] ise, birden fazla robotun yer aldığı üretim sistemlerinde enerji tasarrufu konusunu ele alarak önceki çalışmayı geliştirmiştir. Ele alınan sistemde iş istasyonlarında bulunan birden fazla robot parça üzerinde belirli sayıda işlem gerçekleştirmektedir. Robotların önceden belirlenmiş bir yol üzerindeki hız değişimleri ve ivmeleri belirlenmiştir. Yukarıda da belirtildiği gibi, bahsedilen iki çalışmada da bir noktadan başka bir noktaya yol/patika planlaması için bazı algoritmalar önerilmiştir. Kobetski ve Fabian [63] tarafından birden çok robotun bulunduğu bir sistemde çizelgedeki boş zamanları ortadan kaldıracak şekilde hareketli robotların hızlarını ve ivmelerini azaltmak için iki farklı yöntem geliştirilmiştir. Önerilen yöntemlerde, sistem tümüyle ele alınmış ve enerji tüketimi göz önüne alınmadan elde edilen çözüm üzerinde değişiklikler yapılarak enerji tasarrufu yapılması amaçlanmaktadır. Bahsedilen çalışmanın aksine, bu çalışmada çevrim zamanının optimalliği değiştirilmeden verilen bir çizelgeye ait enerji tüketimi azaltılmaya çalışılmıştır.

Paralel makineli [64–67], akış tipi [68–70] ve hibrit akış tipi [71–74] gibi farklı üretim tiplerindeki enerji tüketimini konu alan çok sayıda çalışma bulunmaktadır. Ancak, bu çalışmaların hiçbirinde bu projedeki gibi enerjiyle ve zamanla ilgili amaçlar beraber ele alınmamakta, sadece enerji ile ilgili amaçlara odaklanılmaktadır. Ayrıca, bahsedilen çalışmaların hiçbirinde robotik hücreler ele alınmamıştır. Bundan dolayı, bahsedilen çalışmaların sonuçları hakkında daha fazla detaya yer verilmeyecektir. Bu konuda detaylı bilgi için Gahm vd. [75] ve Giret vd. [76] tarafından yapılmış iki çalışma incelenebilir.

3.1.3 Çevrim Zamanı ve Enerji Tüketimini Ele Alan 2-Kriterli Optimizasyon

Literatürde hem çevrim zamanı hem de enerji tüketimi ile ilgili amaçları birlikte ele alan birkaç çalışma bulunmaktadır. Fang ve Lin [77] paralel makinelerin çizelgelenmesi probleminde hem

gecikme cezalarını hem de güç tüketim maliyetlerini minimize etmeye çalışmıştır. Burada makinelere işler atanmış ve her iş-makine eşleşmesi için optimal frekanslar belirlenmiştir. Mansouri vd. [68] 2-makinelik akış tipi çizelgeleme problemlerinde yeşil çizelgeleme kavramını tanımlamış ve çevrim zamanı ile enerji tüketimi arasındaki optimal ödünleşim miktarını belirleyen bir sezgisel algoritma geliştirmiştir. Hibrit akış tipi için, tamamlanma zamanını minimize ederken enerji-etkinliğini arttıracak bir karınca koloni algoritması Du vd. [78] tarafından önerilmiştir. Dai vd. [79] bir akıştaki enerji tüketimini minimize etmek için tavlama benzetimi ve genetik algoritmayı bir arada kullanarak bir meta-sezgisel yaklaşım geliştirmiştir. Subai vd. [80] yüzey işlemleri için Vinç Çizelgeleme Problemi'nde (Hoist Scheduling Problem) enerji tüketiminin azaltılması üzerine çalışmıştır. Vinç Çizelgeleme Problemi, iş istasyonları arasında taşıma yapan vinçlerin çizelgelenmesini ele alan problemlerdir. Yapılan çalışmada sistemin verimliliği belirli bir seviyede tutulurken vinçlerin boş zamanları minimize edilmiştir. Zhang vd. [81] esnek üretim sistemlerinde enerji tüketimini azaltırken, çizelgeleme etkinliğini arttıran bir hedef programlama yaklaşımı ortaya atmıştır. Mouzon ve Yildirim [82] toplam gecikmenin (tardiness) ve toplam enerji tüketiminin minimize edilmesini amaçlayan 2-kriterli optimizasyon problemini ele almıştır. Daha sonraki çalışmalarında [83] ise, aynı problemi enerji tüketimi ve üretim zamanını minimize etme amaçlarıyla tekrar ele almıştır. Liu vd. [84] tek makinelik çizelgeleme probleminde toplam üretim zamanını ve CO_2 salınımını minimize etmeye çalışmıştır.

Ek olarak, Gültekin vd. [85] tarafından ele alınan ve verilen bir çevrim zamanına karşılık üretim maliyetlerini minimize eden 2-kriterli modeller de mevcuttur. Burada robotik hücrede kullanılan CNC makinelerinin hızlarının belirli maliyetlere katlanılarak artırılıp azaltılabileceği (dolayısıyla parçaların işlem zamanlarının artırılıp azaltılabileceği); ama robot hızlarının değiştirilmediği varsayılmaktadır. Halbuki, şimdi ele alınan problemde robot hızlarının kontrol edilebildiği varsayılmakta ve verilen bir çevrim zamanına karşılık toplam enerji tüketiminin minimize edilmesi amaçlanmaktadır. Bir başka 2-kriterli optimizasyon probleminde ise, Gültekin vd. [86] 2-makinelik bir robotik hücrede çevrim zamanı ve üretim maliyetlerinin optimizasyonunu ele almıştır. Bahsedilen çalışmada, optimal çevrim zamanı için mevcut üretim maliyetleri minimize edilmektedir. Aktürk ve İlhan [87] ise yaptıkları çalışmada inceledikleri kontrol edilebilir işlem süresine sahip tek bir CNC makine çizelgeleme probleminde, sistemdeki toplam gecikme ve takım-işlem maliyetlerini minimize etme amaçlarını birlikte ele almıştır. Uruk vd. [88] 2-makinelik robotik hücrelerde işlemlerin optimal atamasını ve makinelerin işlem sürelerini belirlemiştir. Böylece, toplam üretim maliyeti ve çevrim zamanı minimize edilmiştir.

Sonuç olarak, literatürdeki çalışmalarda robotik hücredeki çıktı oranının maksimize edilmesi (çevrim zamanının en küçüklenmesi) ve robotun hızlarının değiştirilmesiyle enerji tüketiminin minimize edilmesi amaçları birlikte ele alınmamıştır.

3.2 Geliştirilen Matematiksel Modeller

Geliştirilen modellerde, robotik hücrelerde daha önce bahsedildiği gibi sistemin çevrim zamanını ve sistemdeki robotun enerji tüketimini minimize etme amaçları birlikte ele alınmıştır. Bu iki amacı birlikte ele alabilmek için, ϵ -kısıt yöntemi kullanılmıştır. ϵ -kısıt yönteminde, ele

alınacak amaçlardan biri modele kısıt olarak eklenir ve bu kısıt altında diğer amaç için en iyi çözüm aranır. Bu çalışmada, çevrim zamanı amacı bir üst sınır kullanılarak modellere kısıt olarak eklenmiş ve bu çevrim zamanı üst sınırını sağlayacak en düşük enerji tüketimine sahip robot aktivite döngüsü ve hareket hızlarını veren matematiksel modeller geliştirilmiştir.

Geliştirilen modellerde ilk olarak en sıkı çevrim zamanı üst sınırını elde etmek için, çevrim zamanı amaç fonksiyonu minimize edilmek üzere model çözdürülmüştür. Ardından, son çevrim zamanı üst sınırı olarak en düşük enerji tüketimine sahip olan çözüm için çevrim zamanı bulunmuştur. En düşük enerji tüketimine sahip çözüm, tüm kullanılan robot hareket hızlarının alt sınıra eşit olduğu ve robotun sisteme giren her parçayı girdi noktasından alıp, sırasıyla tüm makinelerdeki işlemlerin tamamlanmasını bekleyip, tüm işlemler tamamlandığında çıktı noktasına götürdüğü döngüdür. En düşük enerji tüketimine sahip çözümün çevrim zamanı en yüksektir. En düşük çevrim zamanı üst sınırından başlanarak, en düşük enerji tüketimine karşılık gelen çevrim zamanı üst sınırında bitecek şekilde istenilen çözüm sayısı kadar enerji tüketimi amaç fonksiyonuna sahip problem eşit aralıklı çevrim zamanı üst sınırları kullanılarak çözdürülmüştür. Böylelikle, modeller sonucunda etkin çözüm kümesi elde edilmiştir. Bu bölümde; sırasıyla geliştirilen matematiksel ve ikinci derece konik modele ait detaylara yer verilecek, ardından iki modelin performanslarının karşılaştırıldığı test problemleri ve elde edilen sonuçlar verilecektir.

3.2.1 Karma Tam Sayılı Doğrusal Olmayan Model (MINLP)

Geliştirilen matematiksel model m -makineli robotik hücrelerde k -birim döngüleri ele almaktadır ve Karma Tam Sayılı Doğrusal Olmayan (MINLP) yapıdadır. Bu model GAMS arayüzü ile kodlanmış ve bir MINLP çözdürücüsü olan BARON çözdürücüsü kullanılarak testler gerçekleştirilmiştir. Aşağıda modelde kullanılan notasyon ve modele ait detaylar verilmiştir:

Notasyon

Kümeler ve Parametreler:

$\mathcal{M} = \{M_0, M_1, \dots, M_{(m+1)}\}$: Makineler kümesi (M_0 girdi noktasını ve $M_{(m+1)}$ çıktı noktasını ifade eder),

$\mathcal{I} = \{0, 1, \dots, m\}$: Robot aktivite indisleri kümesi, (A_0, A_1, \dots, A_m),

$\mathcal{N} = \{1, \dots, n\}$: Tekrarlar kümesi (Bir n -birim döngüde her bir A_i aktivitesi n defa tekrarlanır),

$\mathcal{P} = \{1, 2, \dots, n(m+1)\}$: n -birim döngüde bulunan pozisyonların kümesi. Ek olarak, notasyonu basitleştirmek için u aktivite atanabilecek son pozisyonu ifade etsin ($u = n(m+1)$),

ε : Robotun makineleri yükleme/boşaltma işlemlerini gerçekleştirmesi için geçen süre

f_i : Bir parçanın M_i makinesindeki işlem süresi ($i \in \mathcal{M} \setminus \{0, (m+1)\}$),

$d_{i,j}$: Robotun herhangi M_i makinesinden M_j makinesine katetmesi gereken mesafe ($i, j \in \mathcal{M}$)

\overline{Ct} : Çevrim zamanı için üst sınır

low/up : Robot hızları için alt ve üst sınır değerleri,

Cl, Ce : Sırasıyla robotun dolu ve boş hareketleri sırasında ağırlık ve sürtünmeden dolayı birim mesafe başına ortaya çıkan enerji tüketim değeri için belirlenen katsayılar,

a, b : Dolu ve boş robot hareketleri sırasında robotun sahip olduğu özelliklere göre robotun hareket hızı ve enerji tüketimi arasındaki ilişkiyi gösteren sabit üstel kuvvetler,

G : Herhangi iki makine arasındaki boş hareket süresinin alt sınır değerinden ve herhangi bir makinedeki işlem süresinden daha küçük bir sayı, yani, $G \leq \min \left\{ \min_{g,h \in \mathcal{M}, g \neq h} \{d_{g,h}/up\}, \min_{h \in \mathcal{M}} \{f_h\} \right\}$. Bulunan bir döngüde robot herhangi bir pozisyonda herhangi iki makine arasında boş hareket etmiyorsa, karşılık gelen boş hız değeri sıfır olmalıdır. Ancak, bu durum modelde hızların paydada bulunduğu bazı kısıtlarda (çevrim zamanı hesaplama kısıtı gibi) sıfıra bölünme hatası verebilir. G , bu durumu çözmek için kullanılmıştır,

B : Kısıtlarda kullanılan yeterince büyük bir sayı.

Karar Değişkenleri:

Ct : Çevrim zamanı,

T_p : $p \in \mathcal{P}$ pozisyonuna atanan aktivitenin başlama zamanı,

$$x_{h,i,p} : \begin{cases} 1, & \text{Eğer } p \in \mathcal{P} \text{ pozisyonuna } A_i \text{ aktivitesinin } h. \text{ tekrarı atanmışsa,} \\ 0, & \text{Diğer durumlarda,} \end{cases}$$

$$y_{i,j,p} : \begin{cases} 1, & A_i \text{ ve } A_j \text{ aktivitelerinin herhangi tekrarları sırasıyla } p \text{ ve } (p+1) \text{ pozisyonlarına atanmışsa,} \\ 0, & \text{Diğer durumlarda,} \end{cases}$$

v_p^l : $p \in \mathcal{P}$ pozisyonuna atanan aktivitede yapılan dolu hareket sırasında robotun hızı,

v_p^e : $p \in \mathcal{P}$ pozisyonundaki aktiviteyi tamamladıktan sonra yapılan boş hareket sırasında robotun hızı,

δ_p : $p \in \mathcal{P}$ pozisyonundaki aktiviteyi tamamladıktan sonra yapılan boş hareketin süresi,

d_p^l : $p \in \mathcal{P}$ pozisyonuna atanan aktivitede yapılan dolu hareket sırasında katedilen mesafe,

d_p^e : $p \in \mathcal{P}$ pozisyonundaki aktiviteyi tamamladıktan sonra yapılan boş hareket sırasında katedilen mesafe.

Her bir pozisyona atanan aktivitenin başlangıç zamanını tutan karar değişkenleri çevrim zamanı hesabında kullanılmıştır. Ayrıca sistemde robotun enerji tüketiminin hesaplanması için robot hareket hızlarının bilinmesi gerektiğinden bu hızlar için karar değişkenleri tanımlanmıştır. Ancak, robot bir pozisyona atanan aktiviteyi tamamladıktan sonra bir sonraki pozisyona atanan aktivite sırasında boşaltacağı makinenin önünde ise boş hareket etmezken, değilse

o makinenin önüne boş olarak hareket eder. Boş hareketin olmadığı durumda o pozisyondaki boş hareketin hızı sıfır olmalıdır. Bu durumun sağlanması için biri boş hareketin hızını, diğeri süresini tutan karar değişkenleri tanımlanmıştır.

Model MINLP

$$\text{En Küçükle} \quad \sum_{p \in \mathcal{P}} \left(C_l \cdot d_p^l \cdot (v_p^l)^a + C_e \cdot d_p^e \cdot (v_p^e)^b \right) \quad (14)$$

Öyle ki:

$$x_{1,0,1} = 1 \quad (15)$$

$$\sum_{p \in \mathcal{P}} x_{h,i,p} = 1 \quad \forall i \in \mathcal{I}, \forall h \in \mathcal{N} \quad (16)$$

$$\sum_{h \in \mathcal{N}} \sum_{i \in \mathcal{I}} x_{h,i,p} = 1 \quad \forall p \in \mathcal{P} \quad (17)$$

$$\sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{I}} y_{i,j,p} = 1 \quad \forall p \in \mathcal{P} \quad (18)$$

$$y_{i,j,p} \geq \sum_{h \in \mathcal{N}} x_{h,i,p} + \sum_{h \in \mathcal{N}} x_{h,j,p+1} - 1 \quad \forall i, j \in \mathcal{I}, \forall p \in \mathcal{P} \setminus u \quad (19)$$

$$y_{i,0,u} = x_{n,i,u} \quad \forall i \in \mathcal{I} \quad (20)$$

$$\sum_{p \in \mathcal{P}} p \cdot x_{h,i,p} \leq \sum_{p \in \mathcal{P}} p \cdot x_{s,i,p} \quad \forall i \in \mathcal{I}, \forall h, s \in \mathcal{N} : h < s \quad (21)$$

$$\sum_{\substack{s \in \mathcal{N}, r \in \mathcal{P}, \\ r > k, r < p}} x_{s,i-1,r} \geq x_{h,i,k} + x_{h+1,i,p} - 1 \quad \forall i \in \mathcal{I} \setminus \{0\}, \forall h \in \mathcal{N} \setminus \{n\}, \forall p, k \in \mathcal{P} : k < p \quad (22)$$

$$\sum_{\substack{s \in \mathcal{N}, r \in \mathcal{P}, \\ r > k, r < p}} x_{s,i+1,r} \geq x_{h,i,k} + x_{h+1,i,p} - 1 \quad \forall i \in \mathcal{I} \setminus \{m\}, \forall h \in \mathcal{N} \setminus \{n\}, \forall p, k \in \mathcal{P} : k < p \quad (23)$$

$$T_1 = 0 \quad (24)$$

$$\delta_p \leq G + B \cdot \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{I} \setminus \{i+1\}} y_{i,j,p} \quad \forall p \in \mathcal{P} \quad (25)$$

$$\delta_p \geq G \quad \forall p \in \mathcal{P} \quad (26)$$

$$T_p \geq T_{p-1} + 2\varepsilon + \frac{d_{p-1}^l}{v_{p-1}^l} + \delta_{p-1} \quad \forall p \in \mathcal{P} \setminus \{1\} \quad (27)$$

$$T_p \geq T_k + 2\varepsilon + \frac{d_k^l}{v_k^l} + \sum_{i \in \mathcal{I} \setminus \{0\}} f_i \left(\sum_{h \in \mathcal{N}} x_{h,i-1,k} \cdot \sum_{s \in \mathcal{N}} x_{s,i,p} \right) \quad \forall p, k \in \mathcal{P} : p > k \quad (28)$$

$$Ct + T_p \geq T_k + 2\varepsilon + \frac{d_k^l}{v_k^l} + \sum_{i \in \mathcal{I} \setminus \{0\}} f_i \left(x_{n,i-1,k} \cdot x_{1,i,p} \right) \quad \forall p, k \in \mathcal{P} : p < k \quad (29)$$

$$Ct \geq T_u + 2\varepsilon + \frac{d_u^l}{v_u^l} + \delta_u \quad (30)$$

$$Ct \leq \bar{Ct} \quad (31)$$

$$d_p^l = \sum_{i \in \mathcal{I}} \left(d_{i,i+1} \cdot \sum_{h \in \mathcal{N}} x_{h,i,p} \right) \quad \forall p \in \mathcal{P} \quad (32)$$

$$d_p^e = \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{I}} \left(d_{i+1,j} \cdot y_{i,j,p} \right) \quad \forall p \in \mathcal{P} \quad (33)$$

$$v_p^e \geq \frac{d_p^e}{\delta_p} \quad \forall p \in \mathcal{P} \quad (34)$$

$$v_p^e \leq up \cdot \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{I} \setminus \{i+1\}} y_{i,j,p} \quad \forall p \in \mathcal{P} \quad (35)$$

$$v_p^e \geq low \cdot \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{I} \setminus \{i+1\}} y_{i,j,p} \quad \forall p \in \mathcal{P} \quad (36)$$

$$v_p^l \leq up \quad \forall p \in \mathcal{P} \quad (37)$$

$$v_p^l \geq low \quad \forall p \in \mathcal{P} \quad (38)$$

$$Ct, T_p \geq 0 \quad \forall p \in \mathcal{P} \quad (39)$$

$$x_{h,i,p}, y_{i,j,p} \in \{0, 1\} \quad \forall i, j \in \mathcal{I}, \forall p \in \mathcal{P}, \forall h \in \mathcal{N} \quad (40)$$

Yukarıda verilen modelde, bir robotik hücrede robotun aktiviteler sırasındaki toplam enerji tüketimi minimize edilmiştir. Robotun dolu ve boş hareketleri için; hızların bir üssel kuvveti (sırasıyla $(v_p^l)^a$ ve $(v_p^e)^b$), o hızla katedilen mesafe (sırasıyla d_p^l ve d_p^e) ve o hız tipine ait katsayı (sırasıyla C_l ve C_e) ile çarpılarak ilgili enerji tüketim değerleri hesaplanır ve bu değerler toplanarak robotun döngüdeki toplam enerji tüketim değeri elde edilir.

Döngüsel çizelgeler ele alındığı için, aynı robot hareket sıralamasının sürekli tekrar edilmesi, yani bir döngü oluşturulması gerekmektedir. Bu tür bir döngünün başlama noktası herhangi bir nokta olarak kabul edilebilir. Bu çalışmada, genellikle bir şey kaybetmeden, her döngünün giriş noktasından yeni bir parça alınarak başladığı varsayılmış ve Kısıt (15) ile ilk pozisyona A_0 aktivitesi atanmıştır. (16) numaralı kısıt her aktivite tekrarının bir pozisyona atanması, (17) numaralı kısıt ise her pozisyona bir aktivite tekrarı atanması sağlanmaktadır. Kısıt (18) her pozisyonda birbirini takip eden tek bir aktivite çifti olmasını sağlar. Herhangi bir p pozisyonunda birbirini takip eden aktivite çifti A_i ve A_j ise, A_i aktivitesinin bir tekrarı p pozisyonuna ve A_j aktivitesinin bir tekrarı $(p + 1)$ pozisyonuna atanmış demektir. Son pozisyon dışındaki pozisyonlar için $(p < u)$ birbirini takip eden aktivite çifti (19) ile belirlenir. Kısıt (15)'deki varsayımımızdan ve döngüsel çizelgeler ele alındığından dolayı, son pozisyondaki aktivite her zaman A_0 aktivitesi tarafından takip edilir ve son pozisyon için aktivite çifti (20) numaralı kısıt ile belirlenir.

Kısıt (21), aktivite tekrarlarının artan sırada pozisyonlara atanmasını sağlamaktadır. Sistemdeki robot boş bir makineyi boşaltamaz. Dolayısıyla, A_i aktivitesinin herhangi iki tekrarı arasında A_{i-1} aktivitesinin en az bir tekrarı bulunmalıdır ve bu (22) numaralı kısıt ile sağlanır. Benzer şekilde, robot zaten dolu bir makineyi yükleyemez. Bu yüzden, Kısıt (23) ile sağlandığı gibi A_i aktivitesinin herhangi iki tekrarı arasında A_{i+1} aktivitesinin en az bir tekrarı bulunmalıdır. Böylece, elde edilen robot aktivite sıralamasının olurlu olması sağlanır.

Kısıt (24) ilk pozisyona atanan aktivitenin başlangıç zamanını sıfır yapar. Birbirini takip eden pozisyonlara atanan sıralı aktiviteler için Kısıt (25)'de verilen $\sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{I} \setminus \{i+1\}} y_{i,j,p}$ toplamı sıfıra eşit

olur ve böylelikle Kısıt (25) ve (26)'den dolayı karşılık gelen δ_p daha önce belirlenen alt sınır değerine (G) eşit olur. Bu durum kullanılarak Kısıt (27) ile birbirini takip eden pozisyonlara atanan aktivitelerin başlangıç zamanları (T_{p-1} ve T_p) arasındaki ilişki belirlenmiştir. Ardışık pozisyonlara atanan aktivitelerde; önceki pozisyona atanan aktivite tamamlanmadan, sonraki pozisyona atanan aktivite başlayamaz. Bu kısıtta, $(p - 1)$ pozisyonuna atanan A_i aktivitesi T_{p-1} anında başlar, robot M_i makinesini boşaltır (ε), parçayı M_{i+1} makinesine taşır $\left(\frac{d_{p-1}^l}{v_{p-1}^l}\right)$ ve parçayı M_{i+1} makinesine yükler (ε). Bu kısıtta, eğer $(p - 1)$ ve p pozisyonlarına atanan aktiviteler sıralı ise (sırasıyla A_{i-1} ve A_i aktiviteleri atanmış ise), robot bir sonraki aktiviteyi gerçekleştirmek için boş hareket gerçekleştirmez. Bu durumda, daha önce belirtildiği gibi δ_{p-1} değişkeni G değerine eşit olacağından başlangıç zamanları arasındaki ilişki tam olarak sağlanmamıştır. Bu durumda robot M_i makinesi önünde işlem sona erene kadar bekler ve bu ilişki sonraki kısıtlar ile sağlanmıştır.

Robot parçanın herhangi bir makinedeki işlemi sona erene kadar o makineyi boşaltamaz. (28) ve (29) numaralı kısıtlar, iki farklı olası senaryo için herhangi bir makinedeki parçanın işlemi sona ermeden ilgili makinenin boşaltılmayacağını ifade etmektedir. Kısıt (28) A_i aktivitesinin A_{i-1} aktivitesinden sonraki bir pozisyona atandığı durumu ele almaktadır. Yani, bu kısıtta A_i aktivitesinin p pozisyonuna, A_{i-1} aktivitesinin k pozisyonuna atandığı ve $(p > k)$ olduğu durum ele alınmaktadır. Robot, T_k anında M_{i-1} makinesini boşaltır (ε), M_i makinesine hareket eder $\left(\frac{d_k^l}{v_k^l}\right)$, parçayı M_i makinesine yükler. Sonuç olarak, parçanın M_i makinesindeki işlemi $T_k + 2\varepsilon + \frac{d_k^l}{v_k^l}$ anında başlar ve parçanın işlemi tamamlandığında (f_i birim zaman sonra) artık makine boşaltılabilir. Her pozisyondaki ilgili işlem süresi ise, $\sum_{i \in \mathcal{I} \setminus \{0\}} f_i \left(\sum_{h \in \mathcal{N}} x_{h,i-1,k} \cdot \sum_{s \in \mathcal{N}} x_{s,i,p} \right)$ toplamı ile belirlenir. Böylece, birbirini takip eden pozisyonlara sıralı A_{i-1} ve A_i aktiviteleri atandığı durumda, bu kısıt ile başlangıç zamanları arasındaki ilişki sağlanmış olacaktır.

Çalışmada döngüsel çizelgeler ele alındığından, belirlenen bir döngü sonsuza kadar sürekli tekrarlanır. Dolayısıyla; sistemdeki robot döngünün bir önceki tekrarında yüklediği bir makineyi, aynı döngü içerisinde değil bir sonraki döngüde boşaltabilir. Dolayısıyla, (28) numaralı kısıtta ele alınan durumun tersi Kısıt (29) ile ele alınmıştır. Yani; (29) numaralı kısıtta A_{i-1} aktivitesinin son tekrarının k pozisyonuna, A_i aktivitesinin ilk tekrarının p pozisyonuna atandığı ve $(p < k)$ olduğu durum ele alınmıştır. Kısıt (29), döngünün ardışık iki tekrarında aynı pozisyondaki aktivitelerin başlangıç zamanları arasındaki farkın döngünün çevrim zamanına eşit olduğu bilgisinden yararlanır. Yani, bu döngüde gerçekleştirilen son A_{i-1} aktivitesi ile sonraki döngüde gerçekleştirilen ilk A_i aktivitesini ele alır. Bu durumda, M_i makinesi Kısıt (28) içerisinde hesaplanan süre kadar sonra boşaltılabilir; ancak, ele alınan aktiviteler döngünün ardışık tekrarlarında yer aldığından kısıtın sol tarafı $Ct + T_p$ olarak değiştirilmelidir.

Kısıt (30) ile döngünün çevrim zamanı hesaplanmaktadır. Bir döngünün çevrim zamanı, robotun döngünün herhangi bir tekrarına başladığı andan son pozisyondaki aktiviteyi tamamlayıp sonraki tekrarına başlamak için girdi noktasına geldiği ana kadar geçen süreye eşittir. Robot son pozisyondaki aktiviteye (T_u) anında başlar, ilgili makineyi boşaltır (ε), sonraki makineye parçayı taşır $\left(\frac{d_u^l}{v_u^l}\right)$, o makineyi yükler (ε). Ardından, robot bulunduğu yerden girdi noktasına boş olarak hareket eder (δ_u). Kısıt (31) ise, ε -kısıt yönteminde kullanılan ve çevrim zamanı amacının ele alınmasını sağlayan kısıttır.

Kısıt (32) p pozisyonuna atanan aktivite sırasında yapılan dolu hareket ile katedilen mesafeyi belirlerken, Kısıtlar (33) ve (34) ise bu aktivite tamamlandıktan sonra yapılan boş hareket esnasında katedilen mesafeyi ve robotun hareket hızını belirler. Eğer p pozisyonundaki aktivite tamamlandıktan sonra robot boş hareket gerçekleştiriyorsa, yani ardışık pozisyonlara sıralı aktiviteler atanmış ise, d_p^e ve v_p^e sıfır olmalıdır. Bu durumda, Kısıt (33) ile d_p^e sıfır olur. Ayrıca, bu durumda daha önce bahsedildiği gibi δ_p karar değişkeni G değerine eşittir. Sonuç olarak, Kısıt (34) herhangi bir olursuzluğa yol açılmazken, Kısıtlar (35) ve (36) sayesinde v_p^e değeri sıfıra eşit olur. Çünkü, bu durumda kısıtlarda yer alan $\sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{I} \setminus \{i+1\}} y_{i,j,p}$ toplamı sıfır olur. Aksine p pozisyonundaki aktivite tamamlandıktan sonra robot boş hareket ediyorsa, v_p^e hızı ilgili boş hareket sırasında katedilen mesafenin hareket süresine bölünmesi ile hesaplanır. Dolayısıyla, Kısıt (34) ve robotun enerji tüketimini en küçükleme amaç fonksiyonu sayesinde doğru v_p^e değeri belirlenmiş olur. Ek olarak, Kısıtlar (35) ve (36) ile ilgili boş hareket hızının verilen hız alt ve üst limitleri arasında olması sağlanır. Kısıtlar (37) ve (38) ise, dolu hareketler için robot hızlarının alt ve üst limitler arasında olmasını sağlar. Son olarak, Kısıtlar (39) ve (40) kullanılan karar değişkenlerine ait işaret kısıtlarıdır. Bu modelde amaç fonksiyonu ve Kısıtlar (27)-(30) ve (34) doğrusal olmayan yapıdadır.

3.2.2 Tamsayılı İkinci Derece Konik Programlama (MISOCP) Modeli

Matematiksel modelde yer alan doğrusal olmayan yapıdaki kısımların ikinci derece konik eşitsizliklerle yeniden formüle edilmesiyle tamsayılı ikinci derece konik programlama modeli (MISOCP) elde edilmiştir. Model GAMS arayüzü ile kodlanmış ve bir MISOCP çözdürücüsü olan CPLEX çözdürücüsü kullanılarak testler yapılmıştır. MISOCP oluşturulurken matematiksel modelde doğrusal olmayan, yani karar değişkenlerinin birbirine bölüldüğü veya birbiriyle çarpıldığı, kısıtlar ikinci dereceden (quadratic) kısıtlar haline getirildiğinden; matematiksel modelde kullanılan notasyonda bazı değişikliklere gidilmiştir. Öncelikle, MISOCP'de $d_{i,j}$, Cl ve Ce parametreleri yerine bu parametrelerin karekökleri kullanılmıştır. Ayrıca, aşağıda verilen karar değişkenlerinin tanımında değişikliğe gidilmiştir:

$v_{i,p}^l$: $p \in \mathcal{P}$ pozisyonuna atanan A_i aktivitesinde yapılan dolu hareket sırasında robotun hızı,

$v_{i,j,p}^e$: $p \in \mathcal{P}$ pozisyonuna atanan A_i aktivitesi tamamlandıktan sonra sıradaki A_j aktivitesine başlamak için robotun yaptığı boş hareketin hızı,

$\delta_{i,p}^l$: $p \in \mathcal{P}$ pozisyonuna atanan A_i aktivitesinde yapılan dolu hareketin süresi,

$\delta_{i,j,p}^e$: $p \in \mathcal{P}$ pozisyonuna atanan A_i aktivitesi tamamlandıktan sonra sıradaki A_j aktivitesine başlamak için robotun yaptığı boş hareketin süresi,

Yeni tanımlanan karar değişkenleri kullanılarak modeldeki Kısıtlar (27)-(30) aşağıdaki şekilde güncellenmiştir.

$$T_p \geq T_{p-1} + 2\varepsilon + \sum_{i \in \mathcal{I}} \delta_{i,p-1}^l + \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{I}} \delta_{i,j,p-1}^e \quad \forall p \in \mathcal{P} \setminus \{1\} \quad (27')$$

$$T_p \geq T_k + 2\varepsilon + \sum_{i \in \mathcal{I}} \delta_{i,k}^l + \sum_{i \in \mathcal{I} \setminus \{0\}} f_i \left(\sum_{h \in \mathcal{N}} x_{h,i-1,k} \cdot \sum_{s \in \mathcal{N}} x_{s,i,p} \right) \quad \forall p, k \in \mathcal{P} : p > k \quad (28')$$

$$Ct + T_p \geq T_k + 2\varepsilon + \sum_{i \in \mathcal{I}} \delta_{i,k}^l + \sum_{i \in \mathcal{I} \setminus \{0\}} f_i \left(x_{n,i-1,k} \cdot x_{1,i,p} \right) \quad \forall p, k \in \mathcal{P} : p < k \quad (29')$$

$$Ct \geq T_u + 2\varepsilon + \sum_{i \in \mathcal{I}} \delta_{i,u}^l + \sum_{i \in \mathcal{I}} \delta_{i,0,u}^e \quad (30')$$

Güncellenen yeni kısıtlarda robotun aktivite sırasındaki toplam hareket süresi farklı şekilde yazılmış ve doğrusal olmayan dolu hareket süresini belirten terimler değiştirilmiştir. Örneğin; güncellenen Kısıt (27'), $(p - 1)$ pozisyonundaki dolu hareketin süresini $\left(\frac{d_{p-1}^l}{v_{p-1}^l} \right)$ ifadesi yerine $\left(\sum_{i \in \mathcal{I}} \delta_{i,p-1}^l \right)$ şeklinde ele almaktadır. Benzer şekilde, diğer kısıtlarda da dolu hareket süresini belirleyen karar değişkenlerinin paydada bulunduğu terimler yerine yeni tanımlanan karar değişkenleri getirilmiştir.

Ardından, boş ve dolu hareket hızları ile ilgili yazılan Kısıtlar (35)-(38) aşağıdaki şekilde değiştirilmiştir:

$$v_{i,j,p}^e \leq y_{i,j,p} \cdot up \quad \forall i, j \in \mathcal{I} : j \neq i + 1, \quad \forall p \in \mathcal{P} \quad (35')$$

$$v_{i,j,p}^e \geq y_{i,j,p} \cdot low \quad \forall i, j \in \mathcal{I} : j \neq i + 1, \quad \forall p \in \mathcal{P} \quad (36')$$

$$v_{i,p}^l \leq up \cdot \sum_{h \in \mathcal{N}} x_{h,i,p} \quad \forall i \in \mathcal{I}, \forall p \in \mathcal{P} \quad (37')$$

$$v_{i,p}^l \geq low \cdot \sum_{h \in \mathcal{N}} x_{h,i,p} \quad \forall i \in \mathcal{I}, \forall p \in \mathcal{P} \quad (38')$$

Yukarıda güncellenen hali verilen kısıtlarda o pozisyona atanan aktivitelere karşılık gelen hızların verilen hız alt ve üst limitleri arasında olması sağlanırken, o pozisyona atanmamış aktivitelere karşılık gelen hızların sıfır olması sağlanır. Bu kısıtlarda sırasıyla A_i ve A_{i+1} aktivitelerinin p ve $(p + 1)$ pozisyonlarına atandığı durum ele alınmamıştır. Bu durumda; A_i aktivitesi tamamlandıktan sonra robot boş hareket etmeyeceğinden ilgili robot hızına ve hareket süresine karşılık gelen karar değişkenleri sıfır olmalıdır, bunun için aşağıdaki kısıtlar modele eklenmiştir:

$$v_{i,i+1,p}^e = 0 \quad \forall i \in \mathcal{I} : i \leq m - 1, \quad \forall p \in \mathcal{P} \quad (41)$$

$$\delta_{i,i+1,p}^e = 0 \quad \forall i \in \mathcal{I} : i \leq m - 1, \quad \forall p \in \mathcal{P} \quad (42)$$

Ayrıca, Kısıt (34)'de kullanılan yöntemden yararlanarak robot hızlarının doğru hesaplanması için aşağıdaki kısıtlar eklenmelidir:

$$v_{i,p}^l \geq \frac{d_{i,i+1}}{\delta_{i,p}^l} \sum_{h \in \mathcal{N}} x_{h,i,p} \quad \forall i \in \mathcal{I}, \forall p \in \mathcal{P} \quad (43)$$

$$v_{i,j,p}^e \geq \frac{d_{i+1,j}}{\delta_{i,j,p}^e} y_{i,j,p} \quad \forall i, j \in \mathcal{I}, \forall p \in \mathcal{P} \quad (44)$$

Ancak, yukarıda verilen Kısıtlar (43) ve (44) doğrusal olmayan yapıda olduğundan bu kısıtlar konik ikinci dereceden kısıtlara dönüştürülmüştür. Bu dönüştürme sürecinin adımlarına ait detaylar aşağıda verilmiştir:

$$v \geq \frac{d}{\delta} \Rightarrow \delta v \geq d \Rightarrow 4\delta v \geq 4d \Rightarrow 2\delta v \geq 4d - 2\delta v \Rightarrow (\delta + v)^2 \geq (\delta - v)^2 + (2\sqrt{d})^2.$$

Yukarıda bahsedilen dönüştürmenin uygulanmasının ardından, konik modele Kısıt (43) yerine yeni tanımlanan $w_{1i,p}$, $w_{2i,p}$ ve $w_{3i,p}$ karar değişkenleri kullanılarak oluşturulan aşağıdaki kısıtlar eklenmiştir:

$$w_{1i,p} = v_{i,p}^l - \delta_{i,p}^l \quad \forall i \in \mathcal{I}, \quad \forall p \in \mathcal{P} \quad (45)$$

$$w_{2i,p} = v_{i,p}^l + \delta_{i,p}^l \quad \forall i \in \mathcal{I}, \quad \forall p \in \mathcal{P} \quad (46)$$

$$w_{3i,p} = 2 (d_{i,i+1})^{(1/2)} \sum_{h \in \mathcal{N}} x_{h,i,p} \quad \forall i \in \mathcal{I}, \quad \forall p \in \mathcal{P} \quad (47)$$

$$(w_{1i,p})^2 + (w_{3i,p})^2 \leq (w_{2i,p})^2 \quad \forall i \in \mathcal{I}, \quad \forall p \in \mathcal{P} \quad (48)$$

Benzer şekilde, konik modele Kısıt (44) yerine $w_{4i,j,p}$, $w_{5i,j,p}$ ve $w_{6i,j,p}$ karar değişkenlerinin kullanıldığı aşağıdaki kısıtlar eklenmiştir:

$$w_{4i,j,p} = v_{i,j,p}^e - \delta_{i,j,p}^e \quad \forall i, j \in \mathcal{I}, \quad \forall p \in \mathcal{P} \quad (49)$$

$$w_{5i,j,p} = v_{i,j,p}^e + \delta_{i,j,p}^e \quad \forall i, j \in \mathcal{I}, \quad \forall p \in \mathcal{P} \quad (50)$$

$$w_{6i,j,p} = 2 (d_{i+1,j})^{(1/2)} y_{i,j,p} \quad \forall i, j \in \mathcal{I}, \quad \forall p \in \mathcal{P} \quad (51)$$

$$(w_{4i,j,p})^2 + (w_{6i,j,p})^2 \leq (w_{5i,j,p})^2 \quad \forall i, j \in \mathcal{I}, \quad \forall p \in \mathcal{P} \quad (52)$$

Böylelikle MISOCP'de yer alacak kısıtlardaki doğrusal olmayan terimlerin tamamı İkinci Dereceden Konik Programlama yapısına göre yeniden düzenlenmiştir. Ardından, amaç fonksiyonundaki doğrusal olmayan $E_r = C \cdot d \cdot v^r$ şeklindeki ifadeler düzenlenmelidir. Bu ifadeler $E_r \geq C \cdot d \cdot v^r$ şeklinde kısıt olarak düşünülebilir. Kısıt olarak ele alındığında enerji tüketimini en küçükleme amaç fonksiyonundan dolayı E_r karar değişkenlerinin doğru değerlerini alması sağlanır ve bu kısıtlar konik eşitsizlikler olarak düzenlenebilir. Bunun için; q karar değişkeni

tanımlanıp, bu karar değişkeninin değeri $q = 1$ kısıtı eklenerek sabitlenmiştir. Ancak, kısıtların düzenlenme süreci r üssel kuvvetine göre farklılaşmaktadır. Bu yüzden, r değerinin sırasıyla 2, 3 ve 4 olduğu durumlar için uygulanan düzenlenme süreci örnek olarak verilmiştir.

$r = 2$ olduğunda, $E_2 \geq C \cdot d \cdot v^2$ ifadesi ele alınmaktadır. Bu ifade için aşağıda verilen düzenlenme süreci uygulanır:

$$E_2 \geq C \cdot d \cdot v^2 \Rightarrow 4 E_2 \geq 4 C \cdot d \cdot v^2 \Rightarrow 4 E_2 \cdot q \geq \left(2 \sqrt{C \cdot d \cdot v}\right)^2 \Rightarrow 2 E_2 \cdot q \geq \left(2 \sqrt{C \cdot d \cdot v}\right)^2 - 2 E_2 \cdot q \Rightarrow (E_2 + q)^2 \geq \left(2 \sqrt{C \cdot d \cdot v}\right)^2 + (E_2 - q)^2.$$

Sonuç olarak, $r = 2$ olduğu durumda $(E_2 + q)^2 \geq \left(2 \sqrt{C \cdot d \cdot v}\right)^2 + (E_2 - q)^2$ eşitsizliği kısıt olarak modele eklenir. Benzer şekilde, $r = 3$ olduğunda dönüşüm süreci $E_3 \geq C \cdot d \cdot v^3 \Rightarrow E_3 \cdot C \cdot d \cdot v \geq E_2 \cdot E_2 \Rightarrow 4 E_3 \cdot C \cdot d \cdot v \geq 4 E_2 \cdot E_2 \Rightarrow 2 E_3 \cdot C \cdot d \cdot v \geq 4 E_2 \cdot E_2 - 2 E_3 \cdot C \cdot d \cdot v$ şeklinde ilerler ve sonuç olarak $(E_3 + v \cdot C \cdot d)^2 \geq (2 E_2)^2 + (E_3 - v \cdot C \cdot d)^2$ kısıtı elde edilir. Bu kısıt, E_2 karar değişkenini tanımlayan kısıt ile beraber modele eklenir.

Son olarak, $r = 4$ durumu ele alındığında $E_4 \geq C \cdot d \cdot v^4 \Rightarrow E_4 \cdot C \cdot d \geq E_2 \cdot E_2 \Rightarrow 4 E_4 \cdot C \cdot d \geq 4 E_2 \cdot E_2 \Rightarrow 2 E_4 \cdot C \cdot d \geq 4 E_2 \cdot E_2 - 2 E_4 \cdot C \cdot d \Rightarrow 2 E_4 \cdot C \cdot d \cdot q \geq 4 E_2 \cdot E_2 - 2 E_4 \cdot C \cdot d \cdot q$ şeklinde bir dönüşüm süreci uygulanır ve $(E_4 + q \cdot C \cdot d)^2 \geq (2 E_2)^2 + (E_4 - q \cdot C \cdot d)^2$ kısıtı bulunur. $r = 3$ durumuna benzer şekilde elde edilen kısıt E_2 karar değişkenini tanımlayan kısıt ile birlikte modele eklenmelidir.

Sonuç olarak, herhangi bir r değeri için de benzer şekilde modele eklenecek kısıtlar oluşturulabilir ve dolayısıyla enerji tüketimi amaç fonksiyonu da İkinci Derece Konik Programlama'ya uygun olarak yazılabilir. Notasyonu anlaşılır hale getirmek için ele alınan r değerine karşılık modele eklenmesi gereken kısıtları (AFK) ile gösterelim. Ayrıca, her bir p pozisyonu için robotun dolu ve boş hareketinden kaynaklanan enerji tüketim değerini sırasıyla E_p^l ve E_p^e ile gösterelim.

Bu düzenlemeler sonucunda oluşturulan konik model aşağıdaki şekilde olur:

MISOCP

En küçükke $\sum_{p \in \mathcal{P}} (E_p^l + E_p^e)$

Öyle ki: Kısıtlar (15) – (26), (27') – (30'), (31) – (33), (35') – (38'), (39), (40), (41), (42), (45) – (52), (AFK)

3.2.3 MINLP - MISOCP Karşılaştırması

Önceki bölümlerde detayları verilen iki modelin çözüm süreleri açısından karşılaştırılması için bazı testler yapılmıştır. Bu testlerde 2-makineli ve 3-makineli iki farklı problem oluşturulmuş ve bu problemler her iki modelle çözdürülmüştür. Her iki test probleminde herhangi ardışık iki makine arasındaki mesafe 2 birim olarak alınmıştır. Testler sırasında; ilk olarak çevrim zamanı en küçüklenmesi amaç fonksiyonuyla her iki model çözdürülmüş, ardından enerji tüketimi ve çevrim zamanı amaçları için 5 elemandan oluşan etkin çözüm kümesi bulunmuştur. Etkin çözümlerde bulunan ilk çözüm için kullanılan çevrim zamanı üst sınırı en iyi çevrim

zamanına eşit iken, diğer çözümlerde bir önceki problemin çevrim zamanı 1 birim arttırılarak hesaplanmıştır. Testler sırasında bazı problemler 1 saat zaman limiti altında çözdürülmüştür. Aşağıda verilen Tablo 10'de, çözdürülen test problemlerine ait veriler ve yapılan testlerin ardından elde edilen sonuçlar tablo olarak verilmiştir:

Tablo 10. MINLP - MISOCP Karşılaştırması

No.	Amaç	\overline{Ct}	m	n	P	d	MINLP Sapma (%)	MINLP Süre (sn)	MISOCP Sapma (%)	MISOCP Süre (sn)
01	ÇZ	-	2	2	[5,10]	ÖT	0	25.27	0	1.22
1	ET	25.33	2	2	[5,10]	ÖT	0	27.15	0	9.10
2	ET	26.33	2	2	[5,10]	ÖT	0	27.32	0	5.23
3	ET	27.33	2	2	[5,10]	ÖT	0	28.20	0	8.16
4	ET	28.33	2	2	[5,10]	ÖT	ÇY	ZL	0	2.62
5	ET	29.33	2	2	[5,10]	ÖT	ÇY	ZL	0	2.40
02	ÇZ	41.33	3	2	[5,10]	ÖT	0	34.46	0	10.39
6	ET	41.33	3	2	[5,10]	ÖT	0	41.17	0	12.85
7	ET	42.33	3	2	[5,10]	ÖT	ÇY	ZL	0	19.08
8	ET	43.33	3	2	[5,10]	ÖT	ÇY	ZL	0	12.60
9	ET	44.33	3	2	[5,10]	ÖT	ÇY	ZL	0	23.17
10	ET	45.33	3	2	[5,10]	ÖT	ÇY	ZL	0	11.81

Yukarıda verilen tabloda “ÇZ” çevrim zamanı amaç fonksiyonunu, “ET” ise enerji tüketimi amaç fonksiyonunu temsil etmektedir. Ayrıca, işlem süreleri tabloda belirtilen aralıkta rastgele üretilmiştir. “ÖT” ardışık makineler arasındaki mesafelerin özdeş olduğu, toplanabilir mesafeli robotik hücreleri temsil etmektedir. Tabloda “ZL” zaman limitine takılarak duran problemleri göstermektedir. Zaman limitiyle duran bazı durumlarda olurlu bir çözüm bulunmuşken, bazılarında bulunamamış ve bu durumlar tabloda “ÇY” ile gösterilmiştir. Elde edilen sonuçlara bakarak MISOCP'nin MINLP'ye göre çok daha kısa sürede çözüm verdiği açıkça görülmektedir. Yani, matematiksel model ile küçük boyutlu problemlerde bile makul süreler içerisinde çözüm alınamamaktadır. MISOCP her ne kadar çözüm süresini oldukça kısaltmış olsa da makine sayısının artırılmasıyla gereken çözüm süresi çok hızlı bir şekilde artmaktadır. Bu durumlarda makul süre içerisinde çözümler türetebilmek için sezgisel bir algoritma geliştirilmiştir. Bu algoritmanın performansı, çözüm süresi ve çözüm kalitesi açısından ikinci derece konik model ile karşılaştırılarak ölçülmüştür.

3.3 Sezgisel Algoritma (META)

Bu kısımda özellikle MISOCP'nin yetersiz kaldığı büyük boyutlu robotik çizelgeleme problemleri için bir sezgisel algoritma önerilmektedir. Sezgisel algoritma en düşük çevrim zamanına sahip döngüden başlayarak paret-etkin önyüz üzerinde çözümler türetmeyi amaçlamaktadır. Algoritma sezgisel yöntemle çözümler bulunduğu için elde edilen çözümler yaklaşık etkindir. Algoritmaya m -makina robotik hücreler için etkin çözümler türetmek üzere tasarlandığından kısaca META adı verilmiştir.

Öncelikle iyi (düşük) bir çevrim zamanına karşılık gelen robot hareket sıralaması bulmak

ve bu çözümden hareketle enerji tüketimi için pareto önyüz üzerinde çeşitli sayıda nokta üretmek amacıyla geliştirilen sezgisel algoritmamız; çevrim zamanı için aç gözlü (greedy), enerji tüketimi için ise optimizasyon tabanlı bir yaklaşım kullanmaktadır. Öncelikle, başlangıç çözümü olarak kullanacağımız çevrim zamanı açısından iyi bir çözüm aranmaktadır. Bu çözüm aranırken eldeki bir çözümde çevrim zamanını en çok arttıran aktivitelerin yerleri değiştirilmektedir. Robot hareket hızları belirlenirken ise; enerji maliyet fonksiyonuna en fazla katkısı olan hız değeri öncelikli olarak ele alınıp, bu hız değerini oluşturan aktivitenin komşuları taranmaktadır. Ardından her bir komşu için hızlar hesaplanıp en iyi enerji tüketimine sahip komşu bulunmaktadır.

Eğer herhangi iki makine arasındaki süreler kesin olarak biliniyorsa m -makinelikli bir robotik hücrede çevrim zamanını minimize eden 1-birim döngüsü Crama ve van de Klundert tarafından geliştirilen $O(m^3)$ zamanlı dinamik programlama tabanlı algoritma ile bulunabilmektedir.

Geliştirilen sezgisel algoritma, öncelikle bütün dolu ve boş hızları üst limitlerine eşitleyerek bütün makineler arasındaki hareketlerin olabilecek en düşük zamanını hesaplar. Bu değerler üzerinden Crama ve van de Klundert algoritmasını kullanarak en düşük çevrim zamanını veren 1-birim döngüsü ve bu döngünün n defa tekrarlanması ile bir n -birim döngüsü belirlenir. Elde edilen n -birim döngüde çevrim zamanını en çok arttıran aktivite bulunur ve bu aktivitenin olası komşuları taranarak en düşük çevrim zamanını veren en iyi komşu robot aktivite sıralamasına gidilir. Bu işlem belirli sayıda tekrarlanır ve elde edilen son çözüm pareto önyüz üzerinde noktalar üretmek amacıyla kullanılır. Elde edilen çözüm için çevrim zamanı ve başlangıç zamanları hesaplanır. Ardından enerji tüketim fonksiyonundaki üssel kuvvete ve başlangıç zamanlarına göre en iyi hız değerleri bulunur. Bu hız değerleri arasında enerji tüketimini en fazla arttıran aktivite bulunur ve bu aktivitenin yeri değiştirilerek hem çevrim zamanını aşmayan hem de daha düşük bir enerji tüketim değerine sahip komşu çözümler aranır. Daha iyi bir komşu bulunursa ona gidilir. İşlemler sonucunda elde edilen çözüm pareto önyüzdeki ilk çözümdür. Pareto önyüzündeki son çözüme ait aktivite sıralaması ise, öncelikle her parçanın girdi noktasından alınıp makinelerin hepsinde sırasıyla işlemini tamamladıktan sonra çıktı noktasına götürüldüğü (downhill) aktivite sıralamasıdır. Ayrıca, bu aktivite sıralamasında kullanılan hızların tamamı verilen alt sınıra eşittir. Bu nokta, Teorem 1’de belirtildiği gibi gerekli şartlar sağlandığında o problem için olabilecek en düşük enerji tüketim değerine sahiptir. Sezgisel algoritmamız ile, bu iki noktanın çevrim zamanı arasında kullanıcı tarafından belirtilen sayıda pareto-etkin çözüm elde edilir. Pareto önyüzde herhangi iki çözüm arasındaki adım uzunluğunun λ ile gösterilecektir. Sezgisel algoritmamız λ değerini pareto önyüzdeki ilk ve son çözüme ait çevrim zamanlarını kullanarak kendi hesaplayıp bu λ değerine göre eşit aralıklı etkin çözümler üretmek üzere tasarlanmıştır.

Theorem 1. *Bir malzeme elleçleme robotu, m -makine, bir girdi ve bir çıktı noktasından oluşan bir robotik hücreyi ele alalım ve bu robotik hücrede tek tip parça üretimi yapılsın. Ele alınan hücrede, makineler arası mesafeler için Öklid koşulu sağlansın ve sistemde makineler arasında parça taşıma işlemleri ile parçaların makinelere yükleme/boşaltma işlemlerini sistemdeki robot yerine getirsin. Ayrıca, ilgili sistem için karar verici tarafından verilmiş bir çevrim zamanı üst sınırı olsun, yani herhangi bir döngü verilen çevrim zamanı üst sınırı içerisinde tamamlanıyorsa*

bu döngü karar verici için olası bir döngü olsun ve bu sistem için olası tüm n -birim döngüler ele alınsın. Şayet verilen çevrim zamanı üst sınırı yeterince büyükse, yani verilen çevrim zamanı üst sınırının herhangi bir bağımlılığı yoksa; bu sistemde olası tüm n -birim döngüler için bulunabilecek farklı çözümler içerisinde robotun toplam enerji tüketim değerinin en düşük olduğu çözüm;

i - Kullanılan dolu ve boş hızların tamamı problemde verilen hız alt limitine eşittir,

ii - n -birim döngülerde A_0 aktivitesi ile başlayan ve aktivitelerin indislerine göre artan sırada dizildiği 1-birim döngünün (uphill cycle) n kez tekrarlanması ile elde edilen aktivite sıralamasına sahiptir,

şartlarını sağlar ve bu çözüm sistemdeki robotun toplam enerji tüketim değerinin en küçüklenmesi problemi için optimal olur.

Kanıt. Öncelikle, herhangi bir olurlu robot hareket sıralamasını ele alalım. Bu robot hareket sıralamasını \mathcal{D} ile gösterelim ve döngüsü \mathcal{D} olan bir çözümü düşünelim. Kabul edelim ki bu çözümde, robot hareket hızlarından en az biri hız alt limitinden daha büyük olsun ve bu hızlardan birini v' ile gösterelim. Ayrıca, robot hareket hızı için verilen alt ve üst limitleri de sırasıyla l ($l > 0$) ve u ($u > l$) ile gösterelim. Problemde ele alınan amaç fonksiyonunda, yani (53) numaralı ifadede, daha önce bilindiği gibi herhangi bir hızın amaç fonksiyonuna katkısı $E_r = C \cdot d \cdot v^r$ ($d > 0$ ve $C > 0$ ve $r > 0$) fonksiyonu ile hesaplanır. E_r fonksiyonu $[l, u]$ aralığında sürekli ve (l, u) aralığında türevlenebilir olduğundan burada, E_r fonksiyonunun v hızına göre türevini alırsak; $\forall v \in (l, u)$ için $\frac{\partial E_r}{\partial v} = C \cdot d \cdot r \cdot v^{(r-1)} > 0$ olur. Dolayısıyla, E_r artan bir fonksiyon olur ve $\forall v \in [l, u]$ için $C \cdot d \cdot u^r \geq C \cdot d \cdot v^r \geq C \cdot d \cdot l^r > 0$ ($l > 0$) eşitsizliği sağlanır. Sonuç olarak, daha önce bahsedilen v' ($v' > l$) hızı için de $C \cdot d \cdot (v')^r > C \cdot d \cdot l^r > 0$ eşitsizliği sağlanacağından E_r fonksiyonu en küçük değerini $v = v'$ yerine $v = l$ olduğunda alır.

Şimdi, her pozisyondaki dolu robot hareket hızlarını v_p^l , boş robot hareket hızlarını ise v_p^e ile gösterelim. O halde, amaç fonksiyonu $\sum_{p \in \mathcal{P}} (E_p^l + E_p^e) = \sum_{p \in \mathcal{P}} (C_l \cdot d_p^l \cdot (v_p^l)^a + C_e \cdot d_p^e \cdot (v_p^e)^b)$ şeklinde ifade edilebilir. Burada her bir E_p^l ve E_p^e değerlerini birer E_r fonksiyonu olarak düşünebiliriz. O halde, amaç fonksiyonunu $\sum_{r \in \mathcal{R}} E_r$ ile gösterebiliriz (Burada \mathcal{R} , $\forall p$ için E_p^l ve E_p^e değerlerini ifade eden küme olsun). Dolayısıyla, amaç fonksiyonunda $\forall r$ için E_r fonksiyonunun minimum değerini ilgili hız değerinin alt limite eşit olduğu durumda aldığını ve bu değer sıfırdan büyük olduğunu daha önce gösterdik. Şimdi ise, robot aktivite sıralamasının sabit olduğu durumda $\forall r$ için E_r fonksiyonlarının minimum değerlerinin toplamının $\sum_{r \in \mathcal{R}} E_r$ toplamının minimum değerine eşit olduğunu göstermeliyiz. Bunu tümevarım yöntemini kullanarak ispatlayalım ve ilk olarak $|\mathcal{R}| = 2$ olduğu durum için iddiamızın doğru olduğunu gösterelim. Bu durumda, $\sum_{r \in \mathcal{R}} E_r = E_{r_1} + E_{r_2}$ ve $E_{r_1} = C_1 \cdot d_1 \cdot (v_1)^{r_1}$, $E_{r_2} = C_2 \cdot d_2 \cdot (v_2)^{r_2}$ olur. Ayrıca, $\min_{v \in [l, u]} \{E_{r_1}\} = C_1 \cdot d_1 \cdot l^{r_1}$ ve $\min_{v \in [l, u]} \{E_{r_2}\} = C_2 \cdot d_2 \cdot l^{r_2}$ olduğunu biliyoruz. O halde, adım adım ispatlarsak;

1. $\forall v \in [l, u]$ için $\min_{v \in [l, u]} \{E_{r_1}\} \leq E_{r_1}$ ve $\min_{v \in [l, u]} \{E_{r_2}\} \leq E_{r_2}$ (minimum değer tanımından)
2. $\forall v \in [l, u]$ için $\min_{v \in [l, u]} \{E_{r_1}\} + \min_{v \in [l, u]} \{E_{r_2}\} \leq E_{r_1} + E_{r_2}$ (iki terimi taraf tarafa toplarsak)

3. $\min_{v \in [l, u]} \{E_{r_1}\} + \min_{v \in [l, u]} \{E_{r_2}\} \leq \min_{v \in [l, u]} \{E_{r_1} + E_{r_2}\}$ ($\forall v \in [l, u]$ için 2'deki eşitsizlik sağlandığından)

4. $v = l$ için $E_{r_1} = C_1 \cdot d_1 \cdot l^{r_1}$ ve $E_{r_2} = C_2 \cdot d_2 \cdot l^{r_2}$

5. $E_{r_1} + E_{r_2} = C_1 \cdot d_1 \cdot l^{r_1} + C_2 \cdot d_2 \cdot l^{r_2}$ (iki terimi taraf tarafa toplarsak)

6. $\min_{v \in [l, u]} \{E_{r_1} + E_{r_2}\} \leq C_1 \cdot d_1 \cdot l^{r_1} + C_2 \cdot d_2 \cdot l^{r_2}$ ($\forall v \in [l, u]$ için $\min_{v \in [l, u]} \{E_{r_1} + E_{r_2}\} \leq E_{r_1} + E_{r_2}$)

7. $\min_{v \in [l, u]} \{E_{r_1} + E_{r_2}\} \leq \min_{v \in [l, u]} \{E_{r_1}\} + \min_{v \in [l, u]} \{E_{r_2}\}$ (6'da verilen eşitsizlikten)

8. $\min_{v \in [l, u]} \{E_{r_1} + E_{r_2}\} = \min_{v \in [l, u]} \{E_{r_1}\} + \min_{v \in [l, u]} \{E_{r_2}\}$ (3 ve 7'den)

elde edilir. Böylelikle, iddiamızın $|\mathcal{R}| = 2$ için doğruluğunu göstermiş olduk. $|\mathcal{R}| = n + 1$ olduğu durum için, $|\mathcal{R}| = n$ olduğunda iddianın sağlandığı kabul edilip, $|\mathcal{R}| = 2$ olduğu duruma benzetilerek iddianın doğruluğu kolaylıkla gösterilebilir. Sonuç olarak, tümevarım yönteminden ispat tamamlanır. Yani, herhangi bir döngü için en düşük enerji tüketimine sahip çözümde hızlar alt limitinde olmalıdır. Böylelikle *i*. şart ispatlanmış olur.

ii. şartı ispatlamak için, bu şartta bahsedilen robot aktivite sıralamasını \mathcal{S}_1 ile gösterelim. Şimdi, verilen bir problemde \mathcal{S}_1 döngüsüne sahip çözümün enerji tüketim değerinin, herhangi bir \mathcal{D} döngüsüne sahip çözümden daha düşük veya eşit olduğunu göstermeliyiz. Öncelikle, herhangi bir \mathcal{D} döngüsüne sahip bir çözümü ele alalım. Ayrıca, optimal çözümde hız değerlerinin alt limit değerine eşit olması gerektiğini gösterdiğimizden bu çözümde hızlar sabit ve alt limit değerine eşit olmalıdır. Şimdi, daha önce matematiksel modelde kullandığımız tanımladığımız $\mathcal{I} = \{0, 1, \dots, m\}$ kümesini hatırlayalım. Kabulümüze göre n -birim döngüleri ele aldığımızdan, \mathcal{D} döngüsünde $\forall i \in \mathcal{I}$ için A_i aktivitesi n defa gerçekleşmelidir. Her bir A_i aktivitesi içerisinde M_i makinesinden M_{i+1} makinesine dolu hareket gerçekleştirilir. Dolayısıyla, robot aktivite sıralamasından bağımsız olarak tüm aktiviteler n kez tekrarlanacağından ve her bir A_i aktivitesi gerçekleştiğinde robotun enerji tüketimi $C_l \cdot d_{i,i+1} \cdot l^a$ olacağından; \mathcal{D} döngüsünde robotun dolu hareketlerinden kaynaklanan toplam enerji tüketim değeri $\sum_{p \in \mathcal{P}} E_p^l = \sum_{p \in \mathcal{P}} (C_l \cdot d_p^l \cdot (v_p^l)^a) = n \sum_{i \in \mathcal{I}} C_l \cdot d_{i,i+1} \cdot l^a$ olur. Sonuç olarak, \mathcal{D} döngüsü rastgele seçildiğinden, ilgili robot hareket hızları alt limitine eşit olduğunda tüm döngüler için robotun dolu hareketinden kaynaklanan enerji tüketim değerinin aynı olduğu görülmüş olur. Dolayısıyla; herhangi bir döngüye sahip ve döngü boyunca robotun hız alt limitinde hareket ettiği çözümde robotun enerji tüketim değerini farklılaştıran, robotun boş hareketlerinden kaynaklanan enerji tüketim değeridir. Yani, \mathcal{S}_1 döngüsüne sahip çözümdeki robotun boş hareketlerinden kaynaklanan enerji tüketim değerini herhangi bir \mathcal{D} döngüsüne sahip çözümden daha düşük olduğunu göstermeliyiz.

\mathcal{S}_1 döngüsüne sahip çözüm için robotun boş hareketlerinden kaynaklanan toplam enerji tüketim değerinin $E_{\mathcal{S}_1} = \sum_{p \in \mathcal{P}} E_p^e = \sum_{p \in \mathcal{P}} (C_e \cdot d_p^e \cdot (v_p^e)^b) = n C_e \cdot d_{m+1,0} \cdot l^b$ olduğunu biliyoruz. \mathcal{S}_1 döngüsünde robot önceki parçayı çıktı noktasına bıraktıktan sonra, bir sonraki parçayı girdi noktasından almaya gider. Herhangi bir \mathcal{D} döngüsü için, robot n parçanın her birini almak için çıktı noktası dahil bir makineden girdi noktasına gitmelidir. Robotun yapacağı hareketler döngüye göre farklılaşır. Ancak, her döngü toplamda $n(m+1)$ pozisyona atanmış aktivitelerden oluşur. Burada aktiviteler kümesini de kullanarak \mathcal{D} döngüsü $\{A_{k_1}, A_{k_2}, \dots, A_{k_{n(m+1)}}\}$ şeklinde indislerle gösterilebilir. Burada, $\forall h \in \{1, 2, \dots, n(m+1)\}$ için $A_{k_h} \in \{A_0, A_1, \dots, A_m\}$ şeklindedir

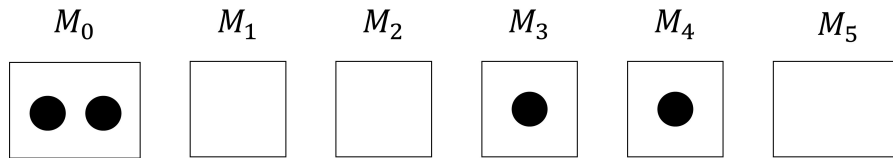
ve $\forall i \in \{0, 1, \dots, m\}$ için $T_i = \{A_{k_h} : A_{k_h} = A_i\}$ şeklinde tanımlanan kümeler için $|T_i| = n$ sağlanmalıdır. Döngünün indislerle gösterimini kullanarak \mathcal{D} döngüsünde robotun boş olarak katettiği mesafelerin kümesini de indis kullanarak $\{d_{k_1+1, k_2}, d_{k_2+1, k_3}, \dots, d_{k_{n(m+1)+1}, k_1}\}$ şeklinde ifade edebiliriz. Dolayısıyla, robotun döngü boyunca yaptığı boş hareketlerden kaynaklanan enerji tüketim değerini E ile gösterirsek, E aşağıdaki formülle hesaplarız:

$$E = \sum_{h=1}^{n(m+1)} E_h = Ce \cdot d_{k_1+1, k_2} \cdot l^b + Ce \cdot d_{k_2+1, k_3} \cdot l^b + Ce \cdot d_{k_3+1, k_4} \cdot l^b + \dots + Ce \cdot d_{k_{n(m+1)+1}, k_1} \cdot l^b = Ce \cdot l^b \cdot (d_{k_1+1, k_2} + d_{k_2+1, k_3} + d_{k_3+1, k_4} + \dots + d_{k_{n(m+1)+1}, k_1})$$

Yukarıdaki formülde Ce ve b robotun katettiği mesafeden ve robot aktivite sıralamasından bağımsızdır ve parametre olarak verilmektedir. Robotun hızı ise ispatın bu kısmı için sabit ve alt sınır değerinde eşit olarak alınmıştır. Şimdi, $d_{k_1+1, k_2} + d_{k_2+1, k_3} + d_{k_3+1, k_4} + \dots + d_{k_{n(m+1)+1}, k_1}$ toplamını daha detaylı olarak incelersek;

1. Öncelikle $\forall h \in \{1, 2, \dots, n(m+1)-1\}$ için eğer $k_h+1 = k_{h+1}$ ise ilgili mesafe $d_{k_h+1, k_{h+1}} = 0$ olur. Dolayısıyla, ilgili robot enerji tüketimi değeri $E_h = 0$ olur. Bu tip durumlarda robot boş hareket gerçekleştirmediğinden, \mathcal{D} döngüsünde oluşabilecek bu durumlar robotun enerji tüketim değerini değiştirmez.
2. Bir önceki maddede belirtilen durum A_0 aktivitesi için gerçekleşemez, yani A_0 aktivitesinden önceki pozisyona atanan aktivite her ne olursa olsun robot boş olarak girdi noktasına hareket etmek zorundadır; çünkü robot ancak önünde bulunduğu makineden daha büyük indisli makinelere dolu olarak gidebilir.

İspatı vermeden önce daha anlaşılır olması için 4-makineli robotik hücrede örnek bir döngü üzerinden durumu anlayalım. Aktivite notasyonunda küçük bir değişiklik yapalım ve l parçası için A_i aktivitesini A_i^l ile gösterelim ($l \in \{1, 2, \dots, n\}$, $i \in \{0, 1, \dots, m\}$).



$$\text{Döngü: } A_0^1 - A_3^4 - A_1^1 - A_0^2 - A_4^4 - A_2^1 - A_3^1 - A_1^2 - A_2^2 - A_0^3 \\ - A_4^1 - A_1^3 - A_3^2 - A_4^2 - A_2^3 - A_3^3 - A_0^4 - A_1^4 - A_2^4 - A_4^3$$

Şekil 3.3. m -makineli tek tip parça üreten robotik hücre için örnek döngü

Bir parçanın boş hareketlerini döngü boyunca parçanın bulunduğu makinelerden boşaltılması için yapılan boş hareketler olarak ele alırsak, verilen örnek için her bir parçanın döngü boyunca yapılan boş hareketleri sırasında katedilen toplam mesafeler sırasıyla,

1. parça için: $d_{5,0} + d_{4,1} + d_{5,2} + d_{1,4}$
2. parça için: $d_{2,0} + d_{4,1} + d_{2,3}$
3. parça için: $d_{3,0} + d_{5,1} + d_{5,2} + d_{3,4}$

4. parça için: $d_{4,0} + d_{1,3} + d_{1,4}$

O halde, döngü boyunca yapılan boş hareketlerle katedilen toplam mesafe ($d_{5,0} + d_{5,2} + d_{2,0} + d_{5,1} + d_{4,0} + d_{5,2} + d_{3,0} + 2d_{4,1} + 2d_{1,4} + d_{2,3} + d_{3,4} + d_{1,3}$) olur. Burada makineler arası mesafeler Öklid şartını sağladığından, biliyoruz ki $d_{5,0} \leq d_{5,2} + d_{2,0}$ ve $d_{5,0} \leq d_{5,1} + d_{4,0}$ ve $d_{5,0} \leq d_{5,2} + d_{3,0}$ eşitsizlikleri sağlanır. Ayrıca, S_1 döngüsü ele alınmış olsaydı, döngü boyunca yapılan boş hareketler sırasında toplam katedilen mesafe $4 \cdot d_{5,0}$ olacaktı. Dolayısıyla, hızlar aynı olduğundan S_1 döngüsü sırasında robotun enerji tüketimi verilen örnek döngüdeki enerji tüketim değerinden daha düşük olur.

Şimdi, bu durumu rastgele olurlu bir \mathcal{D} döngüsü üzerinden ele alalım. Robot bir n -birim döngü içerisinde n kez girdi noktasına ve n kez çıktı noktasına gitmelidir; çünkü sisteme n parça girmeli ve sistemden n parça çıkmalıdır. Ayrıca, robotun bulunduğu makineden daha küçük indisli bir makineye sadece boş olarak hareket edebileceğini biliyoruz. Dolayısıyla, robot her çıktı noktasına gittiğinde daha önceki bir makineye boş olarak hareket etmelidir. Bu makinelerin indislerini sırasıyla j_1, j_2, \dots, j_n ile gösterelim. Ayrıca, robot girdi noktasına her seferinde bir makineden boş olarak gelebilir. Bu makinelerin indislerini ise sırasıyla s_1, s_2, \dots, s_n ile gösterelim. Sonuç olarak, bir döngü içerisinde sırasıyla $d_{(m+1),j_1}, d_{(m+1),j_2}, \dots, d_{(m+1),j_n}$ ve $d_{s_1,0}, d_{s_2,0}, \dots, d_{s_n,0}$ boş hareketleri gerçekleşmektedir. Bu durumu iki alt durum ile ele alabiliriz:

1. $\forall h \in \{1, 2, \dots, n\}$ ve $\forall t \in \{1, 2, \dots, n\}$ ($h \neq t$) için, $g_t \neq g_h$ olmak üzere $\exists g_h \in \{1, 2, \dots, n\} \ni j_h \leq s_{g_h}$ ve $\exists g_t \in \{1, 2, \dots, n\} \ni j_t \leq s_{g_t}$ olur. Bu durumda, makineler arası mesafeler için Öklid şartı sağlandığından $\forall h$ için $d_{(m+1),0} \leq d_{(m+1),s_{g_h}} + d_{s_{g_h},0} \leq d_{(m+1),j_h} + d_{s_{g_h},0}$ ($j_h \leq s_{g_h}$) eşitsizliği sağlanır. Dolayısıyla, \mathcal{D} döngüsü boyunca yapılan boş hareketler sırasında katedilen toplam mesafe $\sum_{h \in \{1, 2, \dots, n\}} d_{(m+1),j_h} + d_{s_{g_h},0}$ değerinden büyük olur ($\forall h$ indisine karşılık gelen farklı bir g_h indisi olduğundan). Sonuç olarak, ilgili boş hareket hızları sabit ve alt sınıra eşit olduğundan \mathcal{D} döngüsünde boş hareketlerden dolayı robotun toplam enerji tüketim değeri $E_{\mathcal{D}} \geq l^b \sum_{h \in \{1, 2, \dots, n\}} (d_{(m+1),j_h} + d_{s_{g_h},0}) \geq l^b n d_{(m+1),0}$ olur. Yani, S_1 döngüsü robotun enerji tüketimi açısından bu tip \mathcal{D} döngülerine göre daha iyi olur.

2. $\exists h \in \{1, 2, \dots, n\}$ için $\nexists g_h \in \{1, 2, \dots, n\} \ni j_h \leq s_{g_h}$ olsun. Burada, robot ilk olarak s_{g_h} makinesinden daha küçük bir indisli bir makinedeki parçayı alıp s_{g_h} makinesine kadar dolu olarak hareket eder ve ardından s_{g_h} makinesinden girdi noktasına gider. Robot \mathcal{D} döngüsünde ya önce j_h aktivitesini ardından $s_{g_h} - 1$ aktivitesini, ya da önce $s_{g_h} - 1$ aktivitesini ardından j_h aktivitesini gerçekleştirebilir.

Öncelikle, j_h aktivitesinin daha önce olduğu durumu ele alalım. O halde, iki ihtimal bulunur. İlk olarak, robot $j_h + 1$ makinesinden çıktı noktasına hiç uğramadan $s_{g_h} - 1$ aktivitesini gerçekleştirebilir. Bu durumda, Öklid şartı geçerli olduğundan $d_{j_h+1, s_{g_h}}$ değeri ilgili aralıkta yapılan boş hareketler sırasında katedilen toplam mesafeden (d) daha küçük veya eşit olur. Dolayısıyla, ilgili h için yapılan boş hareketler sırasında katedilen mesafe için $d_{(m+1),0} \leq d_{(m+1),j_h} + d_{j_h, s_{g_h}} + d_{s_{g_h},0} \leq d_{(m+1),j_h} + d + d_{s_{g_h},0}$ eşitsizliği sağlanır. İkinci olarak, robot $j_h + 1$ makinesinden t için çıktı noktasına uğrayarak $s_{g_h} - 1$ aktivitesini gerçekleştirebilir. Bu durumda, öncelikle robotun yaptığı boş hareketler

sirasında $d_{j_t+1, s_{g_h}}$ değerinden daha fazla mesafe katettiğini biliyoruz. Ayrıca, robotun j_t+1 makinesindeki parçayı boşaltmak için öncelikle $j_h + 1$ makinesindeki parçayı boşaltması gerekir. Dolayısıyla, robot s_{g_h} makinesine gittikten sonra bir adımda $j_h + 1$ makinesindeki parçayı boşaltmak için boş olarak hareket etmelidir. Robot s_{g_h} makinesine gittikten sonra $j_h + 1$ makinesindeki parçayı boşaltmak için en fazla j_t makinesine kadar dolu olarak hareket edebilir; çünkü j_t+1 makinesinde işlenmekte olan bir parça bulunmaktadır. Robot, $j_t + 1$ makinesindeki parçayı boşaltmadan önce $j_h + 1$ makinesindeki parçayı boşaltmaya giderse en az d_{j_t, j_h+1} kadar boş olarak hareket etmelidir. Eğer önce $j_t + 1$ makinesindeki parçayı boşaltırsa ve dolu olarak belirli bir mesafeyi giderse, s_{g_h} makinesindeki parçanın bulunduğu son konuma gelmek için geriye doğru toplamda en az d_{j_h+1, j_t} kadar boş olarak hareket etmelidir. Dolayısıyla, robotun h için yaptığı boş hareketler sırasında katettiği toplam mesafe (d_2) için $d_{(m+1),0} \leq d_{(m+1),j_h} + d_{j_h, j_t} + d_{j_t, s_{g_h}} + d_{s_{g_h},0} \leq d_{(m+1),j_h} + d_{j_t+1, s_{g_h}} + d_{j_t, j_h+1} + d_{s_{g_h},0} \leq d_2$ eşitsizliği sağlanır. Sonuç olarak, olası iki durumda da h için gerçekleştirilen boş hareketler sırasında gidilen toplam mesafe $d_{(m+1),0}$ değerinden daha büyük veya eşit olduğundan ilgili \mathcal{D} döngüsü sırasında robotun toplam enerji tüketim değeri için $E_{\mathcal{D}} \geq l^b n d_{(m+1),0}$ eşitsizliği sağlanır. Yani, \mathcal{S}_1 döngüsü robotun enerji tüketimi açısından bu tip \mathcal{D} döngülerine göre daha iyi olur.

Şimdi, robotun önce s_{g_h} aktivitesini ardından j_h aktivitesini gerçekleştirdiği durumu ele alalım. Bu durumda, robot ilk olarak s_{g_h} makinesinden girdi noktasına gider. Ardından, robot girdi noktasından parçayı alır ve en fazla $s_{g_h} - 1$ makinesine kadar dolu olarak hareket edebilir (s_{g_h} makinesinde işlenmekte olan bir parça bulunmaktadır). Dolayısıyla, robot s_{g_h} makinesine gelip oradaki parçayı daha sonraki bir makineye taşımalıdır (Ayrıca biliyoruz ki j_h makinesinde işlenmekte olan bir parça bulunuyor). Robot girdi noktasına gittikten sonra s_{g_h} makinesindeki parçayı boşaltmadan çıktı noktasına giderse, boş olarak aldığı toplam mesafe en az $d_{s_{g_h}-1, (m+1)}$ kadar olmalıdır. Dolayısıyla, h için $d_{(m+1),0} \leq d_{s_{g_h},0} + d_{s_{g_h}, (m+1)} \leq d_{s_{g_h},0} + d_{s_{g_h}-1, (m+1)}$ eşitsizliği sağlanır. Eğer robot çıktı noktasına gitmeden önce s_{g_h} makinesindeki parçayı boşaltacaksa, robot s_{g_h} makinesindeki parçayı en fazla $j_h - 1$ makinesine kadar dolu olarak taşıyabilir. Öncelikle, robot j_h makinesine kadar aradaki bir makineye (x makinesine) kadar dolu olarak hareket edebilir. Ardından, bu makineden çıktı noktasına boş olarak hareket eder ve $d_{x, (m+1)} \leq d_{x, j_h} + d_{j_h, (m+1)}$ kadar yol alır. Robot farklı hareketler yaptıktan sonra girdi noktasından aldığı ve s_{g_h} makinesinden önceki bir makineye yüklediği parçayı almak için boş olarak hareket etmelidir. Bu parçadan sonraki ilk parça x makinesindeki ise, robot çıktı noktasından j_h makinesine geldikten sonra x 'den daha büyük indisli bir makineden boş olarak girdi noktasından alınan parçanın bulunduğu makineye hareket etmelidir. Dolayısıyla, $d_{x, s_{g_h}}$ değerinden daha fazla bir mesafeyi boş olarak hareket etmelidir. Girdi noktasından alınan parçadan sonraki ilk parça x makinesinde olmasa da (kabul edelim ki y makinesinde olsun), robotun girdi noktasındaki parçayı boşaltması için robotun en az $d_{x, s_{g_h}}$ kadar boş olarak hareket etmesi gerekir. Burada, girdi noktasından alınan parçanın dolu olarak en fazla $y - 1$ makinesine kadar taşınabilir ($y < s_{g_h} < x$). Ancak, y makinesindeki parça en fazla $x - 1$ makinesine kadar dolu olarak taşınabilir. Ardından x makinesindeki parça

boşaltılmalıdır. Bu parça boşaltıldıktan sonra ya y makinesine uğranarak ya da direk olarak robot girdi noktasından aldığı parçanın olduğu makineye boş olarak gitmelidir. Dolayısıyla, en az $d_{x,s_{gh}}$ kadar boş olarak hareket etmelidir. Buradan, h için en az $d_{(m+1),0} \leq d_{s_{gh},0} + d_{s_{gh},x} + d_{x,j_h} + d_{(m+1),j_h} \leq d_{s_{gh},0} + d_{x,(m+1)} + d_{(m+1),j_h} + d_{x,s_{gh}}$ kadar mesafe boş olarak gidilir. Sonuç olarak, olası tüm durumlarda h için gerçekleştirilen boş hareketler sırasında gidilen toplam mesafe $d_{(m+1),0}$ değerinden daha büyük veya eşit olduğundan ilgili \mathcal{D} döngüsü sırasında robotun toplam enerji tüketim değeri için $E_{\mathcal{D}} \geq l^n d_{(m+1),0}$ eşitsizliği sağlanır. Yani, \mathcal{S}_1 döngüsü robotun enerji tüketimi açısından bu tip \mathcal{D} döngülerine göre daha iyi olur.

Sonuç olarak, olası tüm \mathcal{D} döngülerinin robotun enerji tüketim değerleri için \mathcal{S}_1 döngüsünün enerji tüketim değeri daha düşük veya eşit olduğundan *ii.* şart ispatlanmış olur. \square

Pareto önyüzdeki en düşük çevrim zamanına sahip çözüm bulunduğundan sonra, eldeki çevrim zamanı ve aktivite sıralaması kullanılarak robot hareket hızları bulunur. Bu robot hareket hızları kullanılarak çözümün robot enerji tüketim değeri hesaplanır. Ardından, elde edilen çözümün hız değerleri arasında enerji tüketimini en fazla arttıran aktivite bulunur ve bu aktivitenin yeri değiştirilerek hem çevrim zamanını aşmayan hem de daha düşük bir enerji tüketim değerine sahip komşu çözümler aranır. Komşuluk arama tamamlandıktan sonra pareto önyüzde yeni bir nokta bulunmuş olur. Pareto önyüzde bulunan sonraki elemanın çevrim zamanı değeri eldeki çözümün çevrim zamanına λ eklenerek hesaplanır. Bu işlem pareto önyüzde istenilen sayıda eleman elde edilene kadar devam eder.

Pareto-etkin çözümler türeten sezgisel algoritmamıza ait adımlar şu şekilde olacaktır:

M-makina Etkin Çözüm Kümesi Türetme Algoritması (META):

Adım 0 Kabul edelim ki ω Pareto-etkin çözümlerin kümesi, θ bu kümedeki eleman sayısı ve λ ise ω 'da bulunan iki ardışık pareto-çözümün çevrim zamanları arasındaki fark olsun. Ayrıca, D_j ve B_j sırasıyla j . Pareto-etkin çözümün dolu ve boş hızlarının kümesini, \mathcal{V}_l ve \mathcal{V}_e ise sırasıyla elimizdeki herhangi bir çözümde bulunabilecek tüm olası dolu ve boş hızların kümesini gösterecek şekilde kullanılır. Son olarak, t tavlama benzetiminde kullanılan sıcaklık değeri, $s(\cdot)$ ise tavlama benzetiminde kullanılan fonksiyon olsun.

Adım 1 Döngü $\forall i \in \{0, 1, \dots, m\}$ **o zaman**

\Rightarrow İlgili A_i aktivitesi için ilgili dolu hız v_i^l hızını verilen üst limit değerine eşitle.

\Rightarrow İlgili A_i aktivitesi için ilgili aktiviteden sonra yapılabilecek olası tüm boş hareketler için v_i^e hızını verilen üst limit değerine eşitle.

Bitir Döngü

Adım 2 Probleme olası tüm robot hareketleri için makineler arasındaki mesafeler ve karşılık gelen hareket hızlarını kullanarak, \mathcal{D} hareket zamanı matrisini oluştur.

Adım 3 Probleme ait \mathcal{D} hareket zamanı matrisi ve Crama'nın algoritmasını [21] kullanarak problem için en iyi 1-birim döngüyü bul. Bulunan 1-birim döngüyü n kez tekrarlayarak başlangıç çözümü olarak kullanılacak n -birim döngüyü oluştur. Bu çözümün robot aktivite sıralamasını \mathcal{C} ile gösterelim.

Adım 4 Çevrim Zamanı Hesaplama Algoritması'nı kullanarak ve kullanılan hızları üst sınır değerinde alarak çözümün çevrim zamanını (Ct) hesapla.

Adım 5 $Ct^* \leftarrow Ct$.

Adım 6 Hız Hesaplama Algoritması'na göre, çözümün \mathcal{C} robot aktivite sıralamasını ve çevrim zamanı Ct^* 'yi kullanarak robot hareket hızlarını ve aktivitelerin başlangıç zamanlarını hesapla. Bu hesaplamalara göre \mathcal{V}_l ve \mathcal{V}_e kümelerini güncelle.

Adım 7 \mathcal{C} döngüsünde aktiviteleri çevrim zamanına olan etkilerine göre sırala. Sıralama yapılırken, p pozisyonundaki aktivitenin başlangıç zamanı ile bir sonraki pozisyonda bulunan aktivitenin başlangıç zamanlarına bakılır ve aktivitelerin buldukları pozisyonlar bu farklara göre büyükten küçüğe sıralanır (Son pozisyona atanan aktivite için çevrim zamanı ile bu aktivitenin başlangıç zamanı arasındaki fark dikkate alınır). Bu sıralamaya göre, aktivitelerin bulunduğu pozisyonların sıralandığı liste \mathcal{P}_c ile gösterilir.

Adım 8 $sayac \leftarrow 0$.

Adım 9 Döngü $sayac < \text{Adım Sayısı } 1 \text{ o zaman}$

\Rightarrow Komşuluk Arama Algoritması'nı kullanarak \mathcal{P}_c listesindeki ilk elemanın tüm olurlu komşularını tara.

\Rightarrow Bu komşuların kümesini NB , bu kümenin ilk elemanını ise b harfiyle gösterelim. Ayrıca, bu b komşusunun çevrim zamanını Ct_b ve robot aktivite sırasını \mathcal{P}_b ile ifade edelim.

Eğer $Ct_b < Ct^*$ **o zaman**

$\mathcal{P} \leftarrow \mathcal{P}_b$. Adım 6'ya git.

Değilse

$NB \leftarrow NB \setminus \{b\}$.

Bitir Eğer

$\mathcal{P}_c \leftarrow \mathcal{P}_c \setminus \{a\}$. $sayac++$.

Bitir Döngü

Adım 10 Bulunan son çözümün enerji tüketim değeri ET 'yi hesapla.

Adım 11 $ET^* \leftarrow ET$.

Adım 12 \mathcal{P}_c listesine benzer şekilde oluşturulan ancak sıralamanın aktivitelerin enerji tüketimine katkısına göre sıralandığı listeye \mathcal{P}_e diyelim.

Adım 13 $sayac \leftarrow 0$.

Adım 14 Döngü $sayac < \text{Adım Sayısı } 2 \text{ o zaman}$

\Rightarrow Komşuluk Arama Algoritması'nı kullanarak \mathcal{P}_e listesindeki ilk elemanın (a) tüm olurlu komşularını tara.

\Rightarrow Bu komşuların kümesini NB ile, bu kümenin ilk elemanını ise b harfiyle

gösterelim. Ayrıca, b komşusunun çevrim zamanını Ct_b , enerji tüketimini ET_b ve robot aktivite sırasını \mathcal{P}_b ile ifade edelim.

Eğer $Ct_b \leq Ct^*$ o zaman

$\Rightarrow \mathcal{P}_b$ döngüsünde kullanılan hızları Hız Hesaplama Algoritması'nı kullanarak hesapla ve bu hızlardan yararlanarak ET_b değerini hesapla.

Eğer $ET_b < ET^*$ o zaman

$\mathcal{P} \leftarrow \mathcal{P}_b$. $ET^* \leftarrow ET_b$. \mathcal{V}_l ve \mathcal{V}_e kümelerini güncelle ve Adım 12'ye git.

Değilse

$NB \leftarrow NB \setminus \{b\}$.

Bitir Eğer

Değilse

$\Rightarrow r \in [0, 1]$ sayısını rastgele üret.

Eğer $r \leq e^{-\frac{|ET_b - ET^*|}{t}}$ o zaman

Eldeki çözümü b olarak güncelle.

Bitir Eğer

$t \leftarrow s(t)$, $NB \leftarrow NB \setminus \{b\}$ ve Adım 12'ye git.

Bitir Eğer

$\mathcal{P}_e \leftarrow \mathcal{P}_e \setminus \{a\}$. *sayac* ++.

Bitir Döngü

Adım 15 $j \leftarrow |\omega|$.

Adım 16 $D_j \leftarrow \mathcal{V}_l$ ve $B_j \leftarrow \mathcal{V}_e$.

Adım 17 Eğer $|\omega| < \theta$ o zaman

$Ct \leftarrow Ct + \lambda$. Adım 6'ya git.

Değilse

Dur.

Bitir Eğer

Aktivite sıralaması ve kullanılan robot hareket hızları belliyken, bu çözümün çevrim zamanı tekrarlı bir algoritma ile hesaplanır. Algoritmada ilk olarak her bir pozisyona atanan aktivitelerin başlangıç zamanları ve çözümün çevrim zamanı sıfıra eşitlenir. Şimdi, bu algoritmanın adımları aşağıda Çevrim Zamanı Hesaplama Algoritması başlığının altında verilecektir:

Çevrim Zamanı Hesaplama Algoritması:

META'nın bulduğu bir çözümde robot aktivite sırası ve hareket hızları verildiğinde, matematiksel model kullanılmadan ilgili sistemin çevrim zamanını hesaplayan ve tekrarlanan adımlardan oluşan bir algoritma kullanılmıştır. Algoritmada p pozisyonuna atanmış aktiviteyi D_p , bu aktivite sırasında yaptıkları dolu hareket süresini δ_p^l ve bu aktiviteden sonra yapılan boş hareket süresini ise δ_p^e ile gösterelim. Ayrıca, herhangi bir aktivite için kendinden önceki aktivitenin atandığı son pozisyonu k , ($k < p$), ile gösterelim.

s bir sayaç olsun. Algoritma, başlangıçta sayacı (s), tüm aktivitelerin başlangıç zamanlarını ($\forall p, T_p^s$) ve sistemin çevrim zamanını (CT) sıfıra eşitleyerek başlar. Ardından, sayacı bir artırır

ve ilk pozisyona atanan aktivitenin başlangıç zamanını sıfıra eşitler. Diğer pozisyonlardaki aktiviteler için başlangıç zamanlarını ve ilgili çevrim zamanını her adımda $\forall p \in \{1, 2, \dots, n(m+1)\}$ için aşağıdaki denklemlerden yararlanarak hesaplar:

$$T_p^s = \begin{cases} 0, & \text{eğer } p = 1 \text{ ise,} \\ T_{p-1}^s + 2\varepsilon + \delta_{p-1}^l + \delta_{p-1}^e, & \text{eğer } D_p = 0 \text{ ise,} \\ 2\varepsilon + \max\{T_k^s + \delta_k^l + f_{D_p}, T_{p-1}^s + \delta_{p-1}^l + \delta_{p-1}^e\}, & \text{eğer } D_p \neq 0, D_k = D_p - 1, p > k \text{ ise,} \\ 2\varepsilon + \max\{T_k^{s-1} + \delta_k^l + f_{D_p} - CT^{s-1}, T_{p-1}^s + \delta_{p-1}^l + \delta_{p-1}^e\}, & \text{eğer } D_p \neq 0, D_k = D_p - 1, p < k \text{ ise,} \end{cases}$$

Ayrıca, her adımın sonunda ilgili çevrim zamanı değeri

$$CT^s = T_{n(m+1)}^s + 2\varepsilon + \delta_{n(m+1)}^l + \delta_{n(m+1)}^e \quad (53)$$

ile hesaplanır.

Yukarıdaki denklemler iki farklı bilgidен yararlanarak oluşturulmuştur. Bunlardan ilki; p pozisyonuna atanan bir aktivitenin, $(p-1)$ pozisyonuna atanan aktivitenin başlayıp (T_{p-1}), robotun bu aktiviteyi tamamlayarak ($2\varepsilon + \delta_{p-1}^l$) sıradaki aktivite için ilgili makinenin önüne geldiğinden (δ_{p-1}^e) sonra başlayabileceğidir. Diğerisi ise, ilgili makinedeki işlem sona ermeden ona ait aktivitenin başlayamamasıdır. Bu durumda, bir önceki aktivitenin en son başlayıp (T_k), bir sonraki makineye parçayı yükleme işlemini tamamladığı andan ($2\varepsilon + \delta_k^l$) itibaren makinedeki işlem süresi (f_{D_p}) kadar zaman geçtikten sonra p pozisyonundaki aktivite başlayabilir.

Algoritma, sistem kararlı bir hale ulaştığında, yani farklı adımlar sonucu elde edilen başlangıç zamanları özdeş olduğunda, durur. Yani, $\exists u = 1, 2, \dots, s^* - 1$ öyle ki $\forall k, T_k^{s^*} = T_k^{s^*-u}$ şartını sağlayan ilk s^* adımında algoritma durur. Sistemin çevrim zamanı ise u adım sonucu elde edilen sonuçların aritmetik ortalaması ile hesaplanır.

Komşuluk Arama Algoritması:

Crama ve Van de Klundert [21] bir n -birim döngünün olurlu olması için gerek ve yeter şartları yaptıkları bir çalışmada belirtmiştir. Bu çalışmaya göre, olurlu bir n -birim döngüde $\forall i = 1, 2, \dots, m-1$ için iki A_i aktivitesi arasında tam olarak bir A_{i-1} aktivitesi ve tam olarak bir A_{i+1} aktivitesi atanmış olmalıdır. Ayrıca, A_0 (A_m) aktivitesi için, olurluluk için iki A_0 (A_m) aktivitesi arasında tam olarak bir A_1 (A_{m-1}) aktivitesi atanması gerekir.

Algoritmada bir döngünün olurlu olması için sahip olması gereken bu özelliklerden yararlanılarak, bir çözümün robot aktivite sıralaması ve başlangıç pozisyonu p verildiğinde tüm olurlu komşularının taranması sağlanır. Algoritma, p pozisyonuna atanmış olan aktiviteyi bulunduğu pozisyondan alıp olası diğer pozisyonlara yerleştirerek komşuları üretir. Algoritmanın ilk iki adımında bahsedilen olurluluk şartları kullanılarak p pozisyonundaki aktivitenin olursuzluğa yol açmadan yerleştirilebileceği olası pozisyonlar belirlenir. p pozisyonuna atanan aktivite kullanılarak oluşturulan bütün olurlu komşular NB_p kümesinde tutulur. Komşuluk Arama Algoritmasının adımları aşağıda verilmiştir:

Girdi Robot aktivite sıralaması ve komşuları taranacak aktivitenin bulunduğu pozisyon, p

Çıktı p pozisyonundaki aktivitenin tüm olurlu komşularının kümesi NB_p

Adım 1 Döngüde p pozisyonundan önce A_{D_p-1} veya A_{D_p+1} aktivitelerinden birinin atanmış olduğu son pozisyonu belirle. İlk pozisyona kadar bu aktivitelerden biri atanmamış olursa, bahsedilen işleme son pozisyondan başlayarak devam et. Burada bahsedilen duruma rastlanan pozisyonu p_{min} ile gösterelim.

Adım 2 Döngüde p pozisyonundan sonra A_{D_p-1} veya A_{D_p+1} aktivitelerinden birinin atanmış olduğu ilk pozisyonu belirle. Son pozisyona kadar bu aktivitelerden biri atanmamış olursa, bahsedilen işleme ilk pozisyondan başlayarak devam et. Burada bahsedilen duruma rastlanan pozisyonu p_{max} ile gösterelim.

Adım 3 $k = p_{min} + 1$.

Adım 4 Döngü $k \leq p_{max} - 1$ o zaman

$\Rightarrow A_{D_p}$ aktivitesini p pozisyonundan çıkarıp k pozisyonuna yerleştir.

\Rightarrow Oluşturulan yeni aktivite sıralamasını komşuların kümesi NB_p 'ye ekle.

$\Rightarrow A_{D_p}$ aktivitesini p pozisyonuna geri getir.

$\Rightarrow k++$.

Bitir Döngü

Hız Hesaplama Algoritması:

Şimdi, robot aktivite sıralaması ve istenilen çevrim zamanı verildiğinde robotun enerji tüketim değerini minimize eden en iyi hız değerlerini belirlemek için tasarladığımız yöntemi anlatacağız. İstenilen çevrim zamanından bilinen parametreleri (makinelerin yükleme/boşaltma zamanlarını ve robotun tam bekleme zamanlarını) çıkardığımızda elde ettiğimiz zamanı değiştirilmiş çevrim zamanı olarak adlandıralım ve Ct' ile gösterelim. Eğer \mathcal{F} robotun döngü boyunca parçayı makineye yükledikten sonra makine önünde tüm işlem süresince beklediği (tam bekleme) makinelerin kümesi olarak tanımlanırsa, o halde $Ct' = Ct - 2n(m+1)\varepsilon - \sum_{i \in \mathcal{F}} f_i$ şeklinde tanımlanır. Ayrıca, robotun makine önünde beklemediği veya kısmi olarak beklediği makinelerin kümesi $\mathcal{M} \setminus \mathcal{F}$ şeklinde tanımlanır. Robot bu kümede yer alan makineleri yükledikten sonra farklı aktiviteleri gerçekleştirmek için başka bir makinenin önüne gider ve belirli bir zaman sonra bu makineyi boşaltmak için geri döner. Bu durumda, robot makineyi boşaltmak için makine önüne geri döndüğünde işlem sona ermediyse işlem sona erene kadar makine önünde bekler; işlem tamamlandıysa robot makine önünde beklemeden parçayı boşaltır. Robot hızlarını belirlerken, öncelikle tüm kısmi bekleme zamanları ve hız alt sınırlarının sıfır olduğu varsayılır. Sonuç olarak, optimal hız değerlerini belirlemek için aşağıda formülasyonu verilen problemi çözmeliyiz:

$$\text{PS: En küçük} \quad Cl \sum_{p \in \mathcal{P}} d_p^l (v_p^l)^a + Ce \sum_{p \in \mathcal{P}} d_p^e (v_p^e)^b$$

$$\sum_{p \in \mathcal{P}} \left(\frac{d_p^l}{v_p^l} + \frac{d_p^e}{v_p^e} \right) \leq Ct' \quad (54)$$

$$v_p^t \geq 0 \quad p \in \mathcal{P}, t \in \{l, e\} \quad (55)$$

Notasyonu basitleştirmek için verilen aktivite sıralamasında kullanılan tüm hareketlerin

kümesini $\Gamma = \{(p, t), p \in \mathcal{P}, t \in \{l, e\}\}$ ile gösterelim. PS modelinde Ct' sonlu ise, $v_p^t > 0, \forall (p, t) \in \Gamma$ sağlanmalıdır; aksi halde Kısıt (54) sağlanamaz. Şimdi, PS modelinin optimal çözümünü $v^* = \{v_p^{t*} : v_p^{t*} > 0, (p, t) \in \Gamma\}$ ile gösterdiğimiz varsayalım. Ct' sonluysa ve $Ct' \geq \sum_{p \in \mathcal{P}} \left(\frac{d_p^l}{up} + \frac{d_p^e}{up} \right)$ sağlanıyorsa, Kısıt (54)'in optimal çözümde her zaman sıkı bir kısıt olarak sağlandığı kolaylıkla görülebilir. Amaç fonksiyonu ve kısıtı konveks olan bu doğrusal olmayan matematiksel modeli çözmek için KKT şartlarını kullanabiliriz.

Anlatımı kolaylaştırmak için $z(\cdot) = Cl \sum_{p \in \mathcal{P}} d_p^l (v_p^l)^a + Ce \sum_{p \in \mathcal{P}} d_p^e (v_p^e)^b$ ve $f(\cdot) = Ct' - \sum_{p \in \mathcal{P}} \left(\frac{d_p^l}{v_p^l} + \frac{d_p^e}{v_p^e} \right)$ ifadelerini tanımlayalım. Biliyoruz ki, aşağıdaki eşitsizlikleri sağlayacak şekilde en az bir λ çarpanı olmalıdır:

$$\frac{\partial z(v^*)}{\partial v_p^t} + \lambda \frac{\partial f(v^*)}{\partial v_p^t} \geq 0 \quad \forall (p, t) \in \Gamma \quad (56)$$

$$\lambda (f(v^*)) = 0 \quad (57)$$

$$v_p^t \left(\frac{\partial z(v^*)}{\partial v_p^t} + \lambda \frac{\partial f(v^*)}{\partial v_p^t} \right) = 0 \quad \forall (p, t) \in \Gamma \quad (58)$$

$$\lambda \geq 0 \quad (59)$$

Eğer $\lambda = 0$ ise, o zaman (58) numaralı denklemden dolayı $\frac{\partial z(v^*)}{\partial v_p^t} = 0$ eşitliği sağlanmalıdır ve bu durum ancak $v_p^{t*} = 0$ olduğunda sağlanır. Öte yandan, bu durum Kısıt (54) açısından olumsuzluğa yol açar ve $(\lambda = 0)$ olamaz. Dolayısıyla, $(\lambda > 0)$ durumunu ele almalıyız ve $(\lambda > 0)$ durumunda aşağıdaki denklemleri buluruz:

$$\frac{\partial z(v^*)}{\partial v_p^t} + \lambda \frac{\partial f(v^*)}{\partial v_p^t} = 0 \quad \forall (p, t) \in \Gamma \quad (60)$$

$$\sum_{p \in \mathcal{P}} \left(\frac{d_p^f}{v_p^{f*}} + \frac{d_p^e}{v_p^{e*}} \right) = Ct' \quad (61)$$

$$\lambda > 0 \quad (62)$$

Denklem (60), Denklem (58) ve $v_p^t > 0$ eşitsizliği sonucunda; Denklem (61) ise Denklem (57) ve $\lambda > 0$ kullanılarak elde edilir. Denklem (60)'yi kullanarak dolu ve boş robot hareket hızları için aşağıdaki eşitlikleri buluruz:

$$\lambda = Cl \cdot a \cdot (v_p^{l*})^{a+1} \quad \forall p \in \mathcal{P} \quad (63)$$

$$\lambda = Ce \cdot b \cdot (v_p^{e*})^{b+1} \quad \forall p \in \mathcal{P} \quad (64)$$

Sonuç olarak, (63) ve (64) numaralı eşitliklerden $v_{p_1}^{l*} = v_{p_2}^{l*}, v_{p_1}^{e*} = v_{p_2}^{e*}$ ve her $p_1, p_2 \in \mathcal{P}$ için $v_{p_1}^{l*} = \sqrt{(a+1) \frac{Ce \cdot b}{Cl \cdot a}} (v_{p_2}^{e*})^{(b+1)}$ eşitlikleri elde edilir. Bu eşitlikleri ve (61) numaralı denklemi

kullanarak, optimal hız değerleri kolayca hesaplanabilir.

Bu yöntemi kullanarak modeli çözdükten sonra, eğer bulunan hız değerlerinin tamamı verilen alt sınır değerinden daha büyük ise ve makineler önündeki kısmi bekleme zamanlarının tamamı sıfır ise bulunan çözüm optimal olur. Aksi halde, hız değerleri yeniden düzenlenmelidir. Hız Hesaplama Algoritması kullanılarak bu durum ele alınır ve hızlar yeniden düzenlenir. Algoritmada, K verilen döngü için makine önündeki kısmi bekleme zamanının pozitif olduğu makinelerin kümesini, $S_i \subset \Gamma$ ise her $i \in K$ için M_i makinesinin yükleme ve boşaltma zamanı arasında gerçekleştirilen tüm dolu ve boş hareketlerin kümesini gösterebilir.

Aşağıda adımları verilen algoritmada bahsedilen hızların düzenlenmesi sağlanır. Burada, Ω yeniden düzenlenen hızların kümesi olsun. Ayrıca, Ct' ifadesine benzer şekilde, M_i makinesinin değiştirilmiş işlem süresi f'_i ifadesini tanımlayalım; yani f'_i M_i makinesinin verilen işlem süresinden, ilgili makinenin yüklendiği ve boşaltıldığı anlar arasında gerçekleşen tüm yükleme/boşaltma zamanlarının ve tam bekleme zamanlarının çıkarılmasıyla hesaplanır. Hız Hesaplama Algoritması'nın adımları aşağıda verilmiştir:

Girdi Robot aktivite sıralaması, $\forall i \in \mathcal{M}$ için f_i değerleri, $Ct', low, up, Cl, Ce, a, b, \Gamma, \Omega = \emptyset$

Çıktı Yaklaşık optimal hız değerleri

Adım 1 $(p, t) \in \Gamma \setminus \Omega$ kümesindeki hızları kullanarak, (54) numaralı kısıtın sağ tarafının $Ct' - \sum_{(p,t) \in \Omega} \begin{pmatrix} d_p^t \\ v_p^t \end{pmatrix}$ şeklinde değiştirilmesiyle oluşturulmuş yeni **PS** modelini daha önce tanımlanan yöntemi kullanarak çöz. Eğer tüm kısmi bekleme zamanları sıfır ve $v^* \geq low$ ise DUR.

Adım 2 a- Eğer $\exists (p, t)$ öyle ki $v_p^t < low$ ise **o zaman**

\Rightarrow Tüm bu tip (p, t) indisleri için $v_p^t \leftarrow low$ şeklinde düzenle ve bu indisleri Ω kümesine ekle.

\Rightarrow Adım 1'e git.

b- Değilse Eğer en az bir kısmi bekleme zamanı pozitifse **o zaman**

Eğer $\bigcap_{i \in K} S_i = \emptyset$ ise **o zaman**

Döngü $\forall i \in K$ **o zaman**

$(p, t) \in S_i \setminus \Omega$ indislerini ele alarak ve PS modelinde bulunan Kısıt (54)'in

sağ tarafını $f'_i - \sum_{(p,t) \in S_i \cap \Omega} \begin{pmatrix} d_p^t \\ v_p^t \end{pmatrix}$ şeklinde değiştirerek PS modelini tekrar

çöz. Adım 1'i ve Adım 2'nin (a) alt adımını tekrar uygula. Her $(p, t) \in S_i \setminus \Omega$ için hız değerlerini belirle ve bu hızları sabitle.

Bitir Döngü

Adım 1'e git.

Değilse

Döngü Her bir $L \subset K$ için şayet $\bigcap_{i \in L} S_i \neq \emptyset$ ve $\left(\bigcup_{i \in L} S_i \right) \cap \left(\bigcup_{j \in K \setminus L} S_j \right) = \emptyset$ ise **o zaman**

$\Rightarrow L$ kümesinde bulunan hızları kesişimde yer aldıkları bölgeye göre

gruplandır. Örneğin sadece iki küme (S_1 ve S_2) kesişiyorsa $S_1 \cap S_2$, $S_1 \setminus S_2$ ve $S_2 \setminus S_1$ kümelerinde bulunan hızların her biri bir grup oluşturur ve toplamda

3 grup oluşur. $\Rightarrow L$ için denklemleri
$$\sum_{(p,t) \in S_i \setminus \Omega} \begin{pmatrix} d_p^t \\ v_p^t \end{pmatrix} = f_i^t - \sum_{(p,t) \in S_i \cap \Omega} \begin{pmatrix} d_p^t \\ v_p^t \end{pmatrix},$$

$\forall i \in L$ olan bir denklem sistemi oluştur.

\Rightarrow Oluşturulan denklem sistemini çöz ve bu hızları $\exists! (p, t) \in S_i, \forall i \in L$ cinsinden belirle.

\Rightarrow Oluşturulan denklem sisteminde bulunan denklemlerin PS modelinde kısıt olarak bulunduğunu düşündüğümüzde ilgili kısıtların KKT koşullarından, Kısıt (54) ve karşılık gelen çarpanlardan yararlanarak her L bir kümesi için bir denklem oluştur.

Bitir Döngü

\Rightarrow Kısıt (54) ve her L için oluşturulan denklemlerden oluşan denklem sistemini çöz.

\Rightarrow Her $(p, t) \in S_i$ ve $\forall i \in L$ için v_p^t indisli hızların değerlerini, ilgili L kümesinde değeri bilinen hızdan ve diğer hızların bu hız cinsinden yazımlarından yararlanarak belirleyip, bulunan bu hızları sabitle.

\Rightarrow Kalan hızları kullanarak PS modelini çöz ve o hız değerlerini belirle.

Bitir Eğer

Bitir Eğer

META ve içinde kullanılan algoritmaların tamamı C++ dilinde kodlanmıştır. META'nın çözüm kalitesi ve süresi açısından etkinliğinin ölçülmesi için MISOCP ile karşılaştırmalı testler yapılmıştır. Ayrıca, çözüm zamanı açısından performansının ölçülmesi için daha büyük boyutlu problemler META ile çözdürülmüştür. Bu problemlerde döngü sabitlenerek, o döngü için en iyi enerji tüketim değerine sahip çözüm konik model kullanılarak bulunmuştur. İlerleyen bölümlerde öncelikle META'nın çözüm kalitesi açısından değerlendirmesi için çözdürülmüş 3-makineli test problemlerinin sonuçları, ardından ise daha büyük boyutlu problemlerdeki performansının test edilmesi için oluşturulmuş 5 ve 10-makineli test problemlerinin sonuçları verilmiştir.

3.3.1 META - MISOCP Karşılaştırması

META ve MISOCP karşılaştırması iki aşamada yapılmıştır. Bunlardan ilki, daha önce bahsedilen çevrim zamanı üst sınırının belirlenmesi için çözdürülen model açısından yapılan karşılaştırmadır. Ardından bu iki çözüm yöntemi farklı problem parametreleri altında enerji tüketimi amaç fonksiyonu problemi için karşılaştırılmıştır.

META ve MISOCP'yi karşılaştırmak için toplam 60 test problemi oluşturulmuş ve çözdürülmüştür. Performans analizi hem çözüm süresi hem de çözüm kalitesi açısından yapılmıştır. Test problemlerinin tamamında 3-makineli robotik hücreler ele alınmıştır. Oluşturulan test problemleri makinelerin arasındaki uzaklıkların tipine, işlem sürelerinin varyansına ve enerji tüketim fonksiyonundaki dolu ve boş hareket katsayılarına göre farklılaşmaktadır. Öncelikle, test problemlerinde işlem süreleri aynı, küçük varyanslı ve büyük varyanslı olacak şekilde oluşturuldu. Küçük ve büyük varyanslı işlem sürelerine sahip

problemlerde makinelerdeki toplam işlem süresi aynı iken, büyük varyanslı problemlerde ortadaki makinenin işlem süresi diğerlerine göre daha uzun sürmektedir. Burada, parçanın sistemde işlem gördüğü zamanın makinelere dağılımının robotun enerji tüketimine etkisi ölçülmek istenmektedir. Makinelerin arasındaki uzaklıklar ise özdeş veya genel toplanabilir olarak seçilmiştir. Toplanabilir uzaklıkta iki makine arasındaki uzaklık, bu iki makine arasında bulunan ardışık makineler arasındaki uzaklıkların toplanmasıyla hesaplanır. Eğer uzaklıklar eşit ise tüm makineler arasındaki uzaklıklar eşit olurken, genel durumda makineler arasındaki uzaklıkların üçgen eşitsizliğini sağlaması yeterlidir. Özdeş ve genel toplanabilir uzaklıklara göre oluşturulan problemlerde girdi ve çıktı noktası arasındaki uzaklık değişmezken aradaki makinelerin konumları farklılaşmaktadır. Böylece, makinelerin diziliminin robotun enerji tüketimine etkisi ölçülmek istenmiştir. Son olarak, enerji tüketim fonksiyonundaki boş ve dolu hareket hızlarına karşılık gelen katsayılar değiştirilmiştir. Bu katsayıların birbirine eşit olduğu ve dolu hareket hızlarına karşılık gelen katsayının boş hareket hızları için kullanılan katsayıdan daha büyük olduğu durum ele alınmıştır. Buradan, 12 veri tipi oluşturulmuştur.

Her bir veri tipi için 5 farklı problem oluşturulmuştur. Bir veri tipinde oluşturulan farklı problemler, makineler arasındaki mesafe matrisine göre farklılaşır. Mesafe matrisleri rastgele olarak oluşturulmuştur. Her veri tipi için oluşturulan problemleri sıraladığımızı düşünürsek, ilk problemin ilk pareto-etkin çözümünde girdi noktasından parçayı aldıktan sonra son makineden başlayarak ilk makineye doğru azalan sırada sırasıyla makineleri boşalttığı döngü (uphill döngüsü) optimal çözüm olacak şekilde oluşturulmuştur. Ara problemlerin ilk pareto-etkin çözümü için herhangi bir döngü optimal olabilirken, son problemin ilk pareto-etkin çözümüne ait optimal çözüm girdi noktasından başlayarak her parçanın sırasıyla tüm makinelerdeki işlemleri tamamlandıktan sonra çıktı noktasına ulaştığı döngüdür (downhill döngüsü). Ayrıca, oluşturulan tüm problemler için pareto-etkin çözümler daha önce anlatıldığı gibi en iyi çevrim zamanına sahip çözümden başlar ve en düşük enerji tüketimine sahip çözümde sona erer. Pareto-etkin çözüm kümesindeki ardışık çözümlerin çevrim zamanları arasındaki fark eşit ve sabittir. Test problemlerinde her bir problem için pareto-etkin çözüm kümesinin eleman sayısı 5 olarak alınmıştır.

META - MISOCP Çevrim Zamanı Amacı İçin Karşılaştırma:

META ve MISOCP'nin çevrim zamanları açısından karşılaştırılması için oluşturulan test problemleri kullanılmıştır. Bu test problemleri sonucunda elde edilen sonuçlar Tablo 11'de verilmiştir. Tabloda ikinci dereceden konik model için "MISOCP", sezgisel algoritma için ise "META" kısaltması kullanılmıştır. Ayrıca, tabloda problemdeki işlem sürelerinin oluşturulma tipleri eşit olduğu durum "Eşit", küçük varyanslı olduğu durum "KV" ve büyük varyanslı olduğu durum "BV" ile belirtilmiştir. Benzer şekilde, makineler arasındaki mesafeler için "ÖT" ardışık makineler arasındaki mesafelerin özdeş olduğu toplanabilir mesafeli, yani ardışık iki makine arasındaki mesafenin sabit ve herhangi iki makine arasındaki mesafenin ise ardışık mesafelerin toplamı olduğu sistem, robotik hücreleri, "GT" ise makineler arasındaki mesafeler üzerinde üçgen eşitsizliğinin sağlanması dışında herhangi bir kısıtın bulunmadığı ve toplanabilir mesafeli robotik hücreleri temsil etmektedir. Ayrıca, C_l dolu robot hareket hızları ve C_e boş robot hareket hızları için robotun enerji tüketim fonksiyonunda kullanılan katsayıları gösterir. Bu katsayıların

eşit olduğu durum $C_l = C_e$ ile, dolu hareket sırasında daha fazla enerji tüketildiği durum ise $C_l > C_e$ ile gösterilmektedir. Ayrıca, tüm test problemlerinde enerji tüketim fonksiyonunda dolu ve boş hızlara karşılık gelen üstel kuvvetler birbirine eşit ve 2 olarak alınmıştır.

Tabloda problemlerin çözümüne ait robot aktivite döngüleri de verilmiştir. Robot aktivite döngüleri için aşağıda verilen numaralandırmalar kısaltma olarak kullanılmıştır. Bu numaralandırmalarda ilk sayı robotik hücrede bulunan makine sayısını gösterirken, ikinci sayı (yani, “-” işaretinden sonra gelen sayı) döngü indisini göstermektedir. Karşılaştırma için çözdürülen test problemlerinin çözümlerinde karşılaştığımız döngülere ait numaralandırmalar şöyledir:

3-1: $\{A_0, A_1, A_2, A_3, A_0, A_1, A_2, A_3, A_0, A_1, A_2, A_3\}$

3-2: $\{A_0, A_1, A_2, A_3, A_0, A_1, A_2, A_0, A_1, A_3, A_2, A_3\}$

3-3: $\{A_0, A_1, A_3, A_2, A_0, A_1, A_3, A_2, A_0, A_1, A_3, A_2\}$

3-4: $\{A_0, A_1, A_3, A_2, A_0, A_1, A_3, A_2, A_3, A_0, A_1, A_2\}$

3-5: $\{A_0, A_1, A_3, A_2, A_3, A_0, A_1, A_0, A_2, A_1, A_3, A_2\}$

3-6: $\{A_0, A_2, A_1, A_3, A_0, A_2, A_1, A_3, A_0, A_2, A_1, A_3\}$

3-7: $\{A_0, A_2, A_1, A_3, A_0, A_2, A_1, A_0, A_3, A_2, A_1, A_3\}$

3-8: $\{A_0, A_2, A_1, A_3, A_0, A_2, A_1, A_3, A_2, A_0, A_3, A_1\}$

3-9: $\{A_0, A_2, A_1, A_3, A_2, A_0, A_1, A_3, A_2, A_0, A_1, A_3\}$

3-10: $\{A_0, A_2, A_1, A_3, A_2, A_3, A_0, A_1, A_2, A_3, A_0, A_1\}$

3-11: $\{A_0, A_2, A_3, A_1, A_0, A_2, A_1, A_3, A_0, A_2, A_3, A_1\}$

3-12: $\{A_0, A_2, A_3, A_1, A_0, A_2, A_1, A_3, A_2, A_0, A_1, A_3\}$

3-13: $\{A_0, A_2, A_3, A_1, A_0, A_2, A_1, A_3, A_2, A_3, A_0, A_1\}$

3-14: $\{A_0, A_2, A_3, A_1, A_0, A_2, A_3, A_1, A_0, A_2, A_3, A_1\}$

3-15: $\{A_0, A_2, A_3, A_1, A_2, A_0, A_1, A_3, A_2, A_3, A_0, A_1\}$

3-16: $\{A_0, A_2, A_3, A_1, A_2, A_3, A_0, A_1, A_2, A_3, A_0, A_1\}$

3-17: $\{A_0, A_3, A_2, A_1, A_0, A_3, A_2, A_1, A_0, A_3, A_2, A_1\}$

3-18: $\{A_0, A_3, A_2, A_1, A_3, A_2, A_0, A_1, A_3, A_2, A_0, A_1\}$

3-19: $\{A_0, A_3, A_2, A_1, A_3, A_0, A_2, A_1, A_3, A_2, A_0, A_1\}$

3-20: $\{A_0, A_3, A_2, A_3, A_1, A_2, A_0, A_1, A_0, A_3, A_2, A_1\}$

3-21: $\{A_0, A_3, A_2, A_3, A_1, A_0, A_2, A_1, A_0, A_3, A_2, A_1\}$

3–22: $\{A_0, A_3, A_2, A_3, A_1, A_0, A_2, A_3, A_1, A_0, A_2, A_1\}$

3–23: $\{A_0, A_3, A_2, A_3, A_1, A_2, A_3, A_0, A_1, A_2, A_0, A_1\}$

Yukarıda verilen numaralandırma tablolarında bulunan çözümlerin döngüsünü belirtmek için kullanılacaktır. Şimdi, ilk olarak META ve MISOCP'nin karşılaştırması için yapılan test sonuçları Tablo 11'de verilmiştir:

Tablo 11'de verilen sonuçlara göre tüm problemler için MISOCP optimal çözüm bulmuştur. Ayrıca, bu problemler için sezgisel algoritmanın altmış problemin altısı dışında en iyi çevrim zamanını, altısında ise en iyi çevrim zamanına %2'den daha yakın bir çözüm bulmuştur. Verilen test problemlerine baktığımızda, MISOCP'nin en yüksek çözüm zamanınının 38. test probleminde yaklaşık 17 dakika olduğunu görebiliriz. Oysa tüm bu problemler için META'nın çözüm zamanı 1 saniyenin altındadır. Ek olarak, 3-makineli problemlere benzer mantıkla oluşturulan 5-makine 5-birim döngü test problemleri 30 dakika zaman limiti altında MISOCP ile çözüldüğünde herhangi bir çözüm bulunamamıştır; bu sebeple MISOCP ve META'nın karşılaştırması elde edilen çözümlerin kalitesi açısından 3-makineli test problemleri ile yapılmıştır. Dolayısıyla, META'nın çözüm zamanını daha iyi değerlendirebilmek amacıyla test problemleri sadece META ile çözülmüş ve çözüm süreleri kaydedilmiştir. Böylelikle, daha büyük boyutlu problemlerde META'nın çözüm zamanındaki değişim gözlemlenmiş olacaktır.

Çevrim zamanı açısından, robot hareket hızlarının sabit olduğu durumda 3-makineli tek tip parça üreten robotik hücrelerde 1-birim döngülerin optimal olduğu daha önce ispatlanmıştır [92]. Tablo 11'de verilen çözümlerde ise bazı çözümlerde optimal olan döngünün 1-birim döngülerin tekrarı şeklinde olmadığı görülmektedir. Ancak, bu problemlerin tamamında bu tip bir çözümün alternatif optimal olduğu görülmüştür. Zaten, bahsedilen çalışmada verilen ispat bu durumu da kapsamaktadır; çünkü hızların değiştiği durum makineler arasındaki mesafelerin genel uzaklık olduğu duruma benzetilebilir. Makineler arası mesafelerin özdeş toplanabilir olduğu durumda alternatif optimal olan farklı döngülerin sayısı daha fazladır. Bu noktada, alternatif olan döngülerde herhangi bir makine önünde bekleme olmuyorsa robotun döngü boyunca kat ettiği mesafe döngünün çevrim zamanını etkiler. Örneğin; ardışık iki makine arasındaki mesafeyi Δ ile gösterirsek; 3–6:, 3–7:, 3–8: ve 3–17: numaralı döngülerin tamamında robot döngü boyunca (24Δ) 'lık mesafe kateder. Örnek olarak Tablo 11'te görüleceği gibi 3 ve 4 numaralı problemlerde döngüler 3–7: ve 3–17:'nin alternatif optimal olduğu görülebilir. Benzer şekilde, 7 ve 8 numaralı problemlerdeyse döngüler 3–5: ve 3–14: alternatif optimaldir.

META - MISOCP Enerji Tüketimi Amacı İçin Karşılaştırma:

META ile MISOCP enerji tüketimi amaç fonksiyonuna göre karşılaştırmak için de daha önce bahsedilen 3-makineli test problemleri kullanılmıştır. Oluşturulan test problemlerine ait detaylar ve elde edilen sonuçlar sırasıyla Tablo 12'te verilmiştir. Problem numaraları Tablo 11'de verilen numaralandırmaya uygun olarak kullanılmıştır. Bu yüzden tabloda problemlere ait detaylara tabloda yer verilmemiş, sadece problem numarası kullanılmıştır.

Tablo 11. META - MISOCP Çevrim Zamanı Karşılaştırması

No.	<i>m</i>	<i>n</i>	<i>P</i>	<i>d</i>	$C_l - C_e$	MISOCP Döngü	MISOCP Sapma (%)	MISOCP Çözüm Süresi (sn)	META Döngü	META Sapma (%)	META Çözüm Süresi (sn)
1	3	3	Eşit	ÖT	$C_l = C_e$	3-17:	0.00	40.453	3-17:	0.00	0.000
2	3	3	Eşit	ÖT	$C_l > C_e$	3-17:	0.00	43.375	3-17:	0.00	0.001
3	3	3	Eşit	ÖT	$C_l = C_e$	3-7:	0.00	49.188	3-17:	0.00	0.001
4	3	3	Eşit	ÖT	$C_l > C_e$	3-7:	0.00	49.687	3-17:	0.00	0.001
5	3	3	Eşit	ÖT	$C_l = C_e$	3-3:	0.00	67.906	3-3:	0.00	0.001
6	3	3	Eşit	ÖT	$C_l > C_e$	3-3:	0.00	31.500	3-3:	0.00	0.001
7	3	3	Eşit	ÖT	$C_l = C_e$	3-13:	0.00	26.485	3-4:	0.00	0.000
8	3	3	Eşit	ÖT	$C_l > C_e$	3-13:	0.00	556.323	3-4:	0.00	0.000
9	3	3	Eşit	ÖT	$C_l = C_e$	3-1:	0.00	49.672	3-1:	0.00	0.000
10	3	3	Eşit	ÖT	$C_l > C_e$	3-1:	0.00	22.015	3-1:	0.00	0.000
11	3	3	KV	ÖT	$C_l = C_e$	3-17:	0.00	50.359	3-17:	0.00	0.000
12	3	3	KV	ÖT	$C_l > C_e$	3-17:	0.00	47.140	3-17:	0.00	0.001
13	3	3	KV	ÖT	$C_l = C_e$	3-7:	0.00	45.578	3-8:	0.00	0.001
14	3	3	KV	ÖT	$C_l > C_e$	3-6:	0.00	30.797	3-8:	0.00	0.001
15	3	3	KV	ÖT	$C_l = C_e$	3-3:	0.00	89.422	3-3:	0.00	0.001
16	3	3	KV	ÖT	$C_l > C_e$	3-3:	0.00	115.016	3-3:	0.00	0.001
17	3	3	KV	ÖT	$C_l = C_e$	3-3:	0.00	79.281	3-2:	1.98	0.001
18	3	3	KV	ÖT	$C_l > C_e$	3-3:	0.00	777.718	3-2:	1.98	0.001
19	3	3	KV	ÖT	$C_l = C_e$	3-1:	0.00	112.234	3-1:	0.00	0.000
20	3	3	KV	ÖT	$C_l > C_e$	3-1:	0.00	599.797	3-1:	0.00	0.000
21	3	3	BV	ÖT	$C_l = C_e$	3-17:	0.00	35.782	3-17:	0.00	0.000
22	3	3	BV	ÖT	$C_l > C_e$	3-17:	0.00	35.781	3-17:	0.00	0.001
23	3	3	BV	ÖT	$C_l = C_e$	3-17:	0.00	81.844	3-8:	0.00	0.000
24	3	3	BV	ÖT	$C_l > C_e$	3-6:	0.00	82.984	3-8:	0.00	0.000
25	3	3	BV	ÖT	$C_l = C_e$	3-3:	0.00	94.532	3-3:	0.00	0.000
26	3	3	BV	ÖT	$C_l > C_e$	3-5:	0.00	100.922	3-3:	0.00	0.001
27	3	3	BV	ÖT	$C_l = C_e$	3-5:	0.00	81.078	3-13:	0.66	0.000
28	3	3	BV	ÖT	$C_l > C_e$	3-5:	0.00	102.188	3-13:	0.66	0.000
29	3	3	BV	ÖT	$C_l = C_e$	3-1:	0.00	69.563	3-1:	0.00	0.001
30	3	3	BV	ÖT	$C_l > C_e$	3-1:	0.00	54.110	3-1:	0.00	0.000
31	3	3	Eşit	GT	$C_l = C_e$	3-17:	0.00	51.500	3-17:	0.00	0.000
32	3	3	Eşit	GT	$C_l > C_e$	3-17:	0.00	44.016	3-17:	0.00	0.000
33	3	3	Eşit	GT	$C_l = C_e$	3-3:	0.00	129.921	3-3:	0.00	0.001
34	3	3	Eşit	GT	$C_l > C_e$	3-3:	0.00	104.078	3-3:	0.00	0.001
35	3	3	Eşit	GT	$C_l = C_e$	3-3:	0.00	87.032	3-3:	0.00	0.001
36	3	3	Eşit	GT	$C_l > C_e$	3-3:	0.00	78.765	3-3:	0.00	0.001
37	3	3	Eşit	GT	$C_l = C_e$	3-3:	0.00	74.313	3-3:	0.00	0.001
38	3	3	Eşit	GT	$C_l > C_e$	3-3:	0.00	1016.97	3-3:	0.00	0.001
39	3	3	Eşit	GT	$C_l = C_e$	3-1:	0.00	630.265	3-1:	0.00	0.001
40	3	3	Eşit	GT	$C_l > C_e$	3-1:	0.00	118.094	3-1:	0.00	0.001
41	3	3	KV	GT	$C_l = C_e$	3-17:	0.00	44.640	3-17:	0.00	0.001
42	3	3	KV	GT	$C_l > C_e$	3-17:	0.00	47.266	3-17:	0.00	0.001
43	3	3	KV	GT	$C_l = C_e$	3-3:	0.00	36.750	3-3:	0.00	0.001
44	3	3	KV	GT	$C_l > C_e$	3-3:	0.00	114.562	3-3:	0.00	0.001
45	3	3	KV	GT	$C_l = C_e$	3-3:	0.00	61.469	3-3:	0.00	0.001
46	3	3	KV	GT	$C_l > C_e$	3-3:	0.00	87.875	3-3:	0.00	0.001
47	3	3	KV	GT	$C_l = C_e$	3-3:	0.00	69.187	3-3:	0.00	0.000
48	3	3	KV	GT	$C_l > C_e$	3-3:	0.00	30.766	3-3:	0.00	0.000
49	3	3	KV	GT	$C_l = C_e$	3-1:	0.00	769.468	3-1:	0.00	0.000
50	3	3	KV	GT	$C_l > C_e$	3-1:	0.00	961.937	3-1:	0.00	0.000
51	3	3	BV	GT	$C_l = C_e$	3-17:	0.00	36.812	3-17:	0.00	0.001
52	3	3	BV	GT	$C_l > C_e$	3-17:	0.00	37.094	3-17:	0.00	0.001
53	3	3	BV	GT	$C_l = C_e$	3-3:	0.00	53.281	3-3:	0.00	0.001
54	3	3	BV	GT	$C_l > C_e$	3-3:	0.00	146.266	3-3:	0.00	0.001
55	3	3	BV	GT	$C_l = C_e$	3-3:	0.00	108.922	3-3:	0.00	0.001
56	3	3	BV	GT	$C_l > C_e$	3-3:	0.00	86.907	3-3:	0.00	0.001
57	3	3	BV	GT	$C_l = C_e$	3-3:	0.00	67.313	3-3:	0.00	0.000
58	3	3	BV	GT	$C_l > C_e$	3-3:	0.00	90.281	3-3:	0.00	0.003
59	3	3	BV	GT	$C_l = C_e$	3-1:	0.00	74.313	3-1:	0.00	0.000
60	3	3	BV	GT	$C_l > C_e$	3-1:	0.00	958.032	3-1:	0.00	0.001

Tablo 12. META - MISOCP Enerji Tüketimi Amacı Açısından Karşılaştırması

No.	Alt Prob. Numarası	MISOCP Döngü	MISOCP Sapma (%)	MISOCP Dolu Hareket ET (%)	MISOCP Boş Hareket ET (%)	MISOCP Çözüm Süresi (sn)	META Döngü	META Sapma (%)	META Çözüm Süresi (sn)
1	1	3-17:	0.00	39.13	60.87	40.047	3-17:	0.000	1.019
	2	3-17:	0.00	33.33	66.67	40.875	3-17:	0.000	3.008
	3	3-11:	0.00	37.50	62.50	39.937	3-20:	0.000	2.297
	4	3-5:	0.00	42.86	57.14	42.016	3-15:	0.000	3.92
	5	3-1:	0.00	50.00	50.00	45.062	3-1:	0.000	2.872
2	1	3-17:	0.00	56.25	43.75	41.625	3-17:	0.000	1.437
	2	3-17:	0.00	50.00	50.00	48.328	3-17:	0.000	4.241
	3	3-12:	0.00	54.55	45.45	42.422	3-21:	4.545	3.418
	4	3-4:	0.00	60.00	40.00	39.125	3-23:	0.000	3.195
	5	3-1:	0.00	66.67	33.33	32.718	3-1:	0.000	2.279
3	1	3-6:	0.00	33.33	66.67	87.891	3-17:	0.000	1.716
	2	3-13:	0.00	40.00	60.00	49.219	3-14:	0.000	3.137
	3	3-13:	0.00	40.00	60.00	48.546	3-14:	0.000	2.653
	4	3-1:	0.00	50.00	50.00	38.313	3-1:	0.000	2.664
	5	3-1:	0.00	50.00	50.00	16.593	3-1:	0.000	1.922
4	1	3-17:	0.00	50.00	50.00	96.625	3-17:	0.000	1.587
	2	3-5:	0.00	45.65	54.35	44.360	3-14:	0.000	2.532
	3	3-5:	0.00	45.65	54.35	45.687	3-14:	0.000	3.164
	4	3-13:	0.00	51.92	48.08	42.719	3-2:	0.000	2.027
	5	3-1:	0.00	66.67	33.33	31.375	3-1:	0.000	2.407
5	1	3-3:	0.00	40.00	60.00	76.672	3-3:	0.000	2.028
	2	3-1:	0.00	50.00	50.00	44.672	3-1:	0.000	2.34
	3	3-1:	0.00	50.00	50.00	44.485	3-1:	0.000	2.224
	4	3-1:	0.00	50.00	50.00	36.063	3-1:	0.000	2.962
	5	3-1:	0.00	50.00	50.00	32.641	3-1:	0.000	1.953
6	1	3-14:	0.00	57.14	42.86	104.203	3-3:	0.000	2.045
	2	3-1:	0.00	55.75	44.25	25.937	3-1:	0.000	3.183
	3	3-1:	0.00	55.75	44.25	36.468	3-1:	0.000	3.019
	4	3-1:	0.00	55.75	44.25	36.969	3-1:	0.000	2.959
	5	3-1:	0.00	66.67	33.33	37.344	3-1:	0.000	3.212
7	1	3-4:	0.00	42.86	57.14	681.11	3-4:	0.002	1.401
	2	3-1:	0.00	50.00	50.00	41.156	3-1:	0.000	1.968
	3	3-1:	0.00	50.00	50.00	32.172	3-1:	0.000	1.838
	4	3-1:	0.00	50.00	50.00	41.984	3-1:	0.000	1.912
	5	3-1:	0.00	50.00	50.00	31.407	3-1:	0.000	2.089
8	1	3-13:	0.00	55.53	44.47	90.984	3-13:	0.000	1.624
	2	3-1:	0.00	55.75	44.25	49.359	3-1:	0.000	1.923
	3	3-1:	0.00	55.75	44.25	40.703	3-1:	0.001	1.581
	4	3-1:	0.00	55.75	44.25	39.125	3-1:	0.001	1.972
	5	3-1:	0.00	66.67	33.33	33.704	3-1:	0.000	1.588
9	1	3-1:	0.00	50.00	50.00	85.704	3-1:	0.000	1.324
	2	3-1:	0.00	50.00	50.00	26.453	3-1:	0.001	3.025
	3	3-1:	0.00	50.00	50.00	38.813	3-1:	0.000	2.88
	4	3-1:	0.00	50.00	50.00	37.797	3-1:	0.001	3.135
	5	3-1:	0.00	50.00	50.00	35.625	3-1:	0.000	2.983
10	1	3-1:	0.00	66.67	33.33	73.844	3-1:	0.000	0.954
	2	3-1:	0.00	55.75	44.25	40.829	3-1:	0.001	1.627
	3	3-1:	0.00	55.75	44.25	45.719	3-1:	0.001	2.428
	4	3-1:	0.00	55.75	44.25	36.609	3-1:	0.001	2.005
	5	3-1:	0.00	66.67	33.33	33.921	3-1:	0.000	1.974
11	1	3-17:	0.00	40.93	59.07	82.781	3-17:	0.000	1.173
	2	3-17:	0.00	33.34	66.66	91.282	3-17:	0.000	2.040
	3	3-12:	0.00	37.50	62.50	86.016	3-21:	6.250	3.090
	4	3-4:	0.00	42.86	57.14	74.157	3-4:	0.001	2.960
	5	3-1:	0.00	50.00	50.00	72.765	3-1:	0.000	2.556
12	1	3-17:	0.00	55.01	44.99	78.282	3-17:	0.000	1.482
	2	3-17:	0.00	45.10	54.90	78.344	3-17:	0.000	2.723
	3	3-9:	0.00	54.55	45.45	78.375	3-22:	0.000	1.462
	4	3-4:	0.00	60.00	40.00	80.453	3-4:	0.000	2.158
	5	3-1:	0.00	66.67	33.33	75.828	3-1:	0.000	1.479
13	1	3-6:	0.00	33.33	66.67	101.235	3-8:	0.000	1.181
	2	3-3:	0.00	40.00	60.00	45.453	3-13:	4.300	2.329
	3	3-3:	0.00	40.00	60.00	46.484	3-5:	2.304	1.382
	4	3-1:	0.00	50.00	50.00	38.844	3-1:	0.001	1.683
	5	3-1:	0.00	50.00	50.00	35.906	3-1:	0.000	1.324
14	1	3-6:	0.00	50.00	50.00	192.375	3-8:	0.000	1.267
	2	3-3:	0.00	45.65	54.35	59.5	3-13:	4.301	1.608
	3	3-3:	0.00	45.65	54.35	62.609	3-5:	1.569	1.926
	4	3-1:	0.00	55.75	44.25	52.593	3-1:	0.001	1.442
	5	3-1:	0.00	66.67	33.33	47.812	3-1:	0.000	1.357
15	1	3-3:	0.00	40.00	60.00	138.016	3-3:	0.000	1.164
	2	3-1:	0.00	50.00	50.00	69.625	3-1:	0.000	1.43
	3	3-1:	0.00	50.00	50.00	72.281	3-1:	0.000	1.72
	4	3-1:	0.00	50.00	50.00	31	3-1:	0.000	1.574
	5	3-1:	0.00	50.00	50.00	49.515	3-1:	0.000	1.509

No.	Alt Prob. Numarası	MISOCP Döngü	MISOCP Sapma (%)	MISOCP Dolu Hareket ET (%)	MISOCP Boş Hareket ET (%)	MISOCP Çözüm Süresi (sn)	META Döngü	META Sapma (%)	META Çözüm Süresi (sn)
16	1	3-3:	0.00	57.14	42.86	171.891	3-3:	0.000	0.88
	2	3-1:	0.00	55.75	44.25	59.422	3-1:	0.000	1.796
	3	3-1:	0.00	55.75	44.25	67.828	3-1:	0.000	1.505
	4	3-1:	0.00	55.75	44.25	60.11	3-1:	0.000	1.753
	5	3-1:	0.00	66.67	33.33	46.328	3-1:	0.000	1.469
17	1	3-2:	0.00	46.15	53.85	126	3-2:	0.000	1.356
	2	3-1:	0.00	50.00	50.00	58.391	3-1:	0.000	1.269
	3	3-1:	0.00	50.00	50.00	57.297	3-1:	0.000	1.498
	4	3-1:	0.00	50.00	50.00	55.89	3-1:	0.000	1.202
	5	3-1:	0.00	50.00	50.00	53.547	3-1:	0.000	1.378
18	1	3-2:	0.00	62.28	37.72	131.156	3-2:	0.000	0.788
	2	3-1:	0.00	55.75	44.25	63.266	3-1:	0.001	1.270
	3	3-1:	0.00	55.75	44.25	41.25	3-1:	0.001	1.637
	4	3-1:	0.00	55.75	44.25	58.97	3-1:	0.000	1.321
	5	3-1:	0.00	66.67	33.33	48.281	3-1:	0.000	1.473
19	1	3-1:	0.00	50.00	50.00	137.532	3-1:	0.000	0.662
	2	3-1:	0.00	50.00	50.00	55.687	3-1:	0.000	1.569
	3	3-1:	0.00	50.00	50.00	62.11	3-1:	0.000	1.263
	4	3-1:	0.00	50.00	50.00	55.016	3-1:	0.001	1.71
	5	3-1:	0.00	50.00	50.00	46.797	3-1:	0.000	1.359
20	1	3-1:	0.00	66.67	33.33	145.093	3-1:	0.000	0.745
	2	3-1:	0.00	55.75	44.25	54.781	3-1:	0.001	1.49
	3	3-1:	0.00	55.75	44.25	66.344	3-1:	0.001	1.463
	4	3-1:	0.00	55.75	44.25	61.765	3-1:	0.001	1.483
	5	3-1:	0.00	66.67	33.33	51.359	3-1:	0.000	1.525
21	1	3-17:	0.00	42.92	57.08	83.625	3-17:	0.000	1.134
	2	3-6:	0.00	33.33	66.67	84.766	3-17:	0.000	1.822
	3	3-9:	0.00	37.50	62.50	73.672	3-20:	0.000	1.938
	4	3-4:	0.00	42.86	57.14	82	3-23:	0.000	2.109
	5	3-1:	0.00	50.00	50.00	71.828	3-1:	0.000	2.371
22	1	3-17:	0.00	57.71	42.29	72.469	3-17:	0.000	1.348
	2	3-6:	0.00	50.00	50.00	68.578	3-17:	0.000	1.64
	3	3-9:	0.00	54.55	45.45	69.297	3-20:	0.000	1.969
	4	3-5:	0.00	60.00	40.00	85.86	3-23:	0.000	1.826
	5	3-1:	0.00	66.67	33.33	35.765	3-1:	0.000	1.45
23	1	3-6:	0.00	33.33	66.67	185.688	3-8:	0.000	1.308
	2	3-5:	0.00	40.00	60.00	68.172	3-13:	0.000	1.839
	3	3-5:	0.00	40.00	60.00	67.75	3-13:	0.000	1.312
	4	3-1:	0.00	50.00	50.00	58.547	3-1:	0.001	1.328
	5	3-1:	0.00	50.00	50.00	49.032	3-1:	0.000	1.532
24	1	3-6:	0.00	50.00	50.00	146.641	3-8:	0.000	1.396
	2	3-13:	0.00	45.65	54.35	57.484	3-13:	0.001	2.033
	3	3-5:	0.00	45.65	54.35	61.266	3-13:	0.000	1.606
	4	3-1:	0.00	55.75	44.25	58.156	3-16:	1.261	1.591
	5	3-1:	0.00	66.67	33.33	45.781	3-1:	0.000	1.851
25	1	3-14:	0.00	40.00	60.00	158.703	3-3:	0.000	1.331
	2	3-1:	0.00	50.00	50.00	69.969	3-1:	0.000	1.357
	3	3-1:	0.00	50.00	50.00	60.188	3-1:	0.001	1.854
	4	3-1:	0.00	50.00	50.00	58.469	3-1:	0.002	1.804
	5	3-1:	0.00	50.00	50.00	44.969	3-1:	0.000	1.624
26	1	3-14:	0.00	57.14	42.86	120.39	3-3:	0.000	1.319
	2	3-1:	0.00	55.75	44.25	61.406	3-1:	0.000	1.452
	3	3-1:	0.00	55.75	44.25	54.953	3-1:	0.000	1.306
	4	3-1:	0.00	55.75	44.25	63.563	3-1:	0.003	1.642
	5	3-1:	0.00	66.67	33.33	52.328	3-1:	0.000	1.302
27	1	3-13:	0.00	40.00	60.00	168.61	3-13:	0.001	1.263
	2	3-1:	0.00	50.00	50.00	65.25	3-1:	0.001	1.405
	3	3-1:	0.00	50.00	50.00	63.062	3-1:	0.001	1.519
	4	3-1:	0.00	50.00	50.00	52.953	3-1:	0.002	1.49
	5	3-1:	0.00	50.00	50.00	55.844	3-1:	0.000	1.617
28	1	3-13:	0.00	56.22	43.78	113.031	3-13:	0.000	1.332
	2	3-1:	0.00	55.75	44.25	63.609	3-1:	0.001	1.409
	3	3-1:	0.00	55.75	44.25	61.953	3-1:	0.002	1.329
	4	3-1:	0.00	55.75	44.25	54.235	3-1:	0.001	1.252
	5	3-1:	0.00	66.67	33.33	50.406	3-1:	0.000	1.474
29	1	3-1:	0.00	50.00	50.00	107.234	3-1:	0.000	0.737
	2	3-1:	0.00	50.00	50.00	56.11	3-1:	0.000	1.941
	3	3-1:	0.00	50.00	50.00	67.532	3-1:	0.000	1.479
	4	3-1:	0.00	50.00	50.00	59.016	3-1:	0.001	1.305
	5	3-1:	0.00	50.00	50.00	48.188	3-1:	0.000	1.298
30	1	3-1:	0.00	66.67	33.33	145.172	3-1:	0.000	0.749
	2	3-1:	0.00	55.75	44.25	69.329	3-1:	0.001	1.502
	3	3-1:	0.00	55.75	44.25	61.75	3-1:	0.001	1.374
	4	3-1:	0.00	55.75	44.25	54.266	3-1:	0.001	1.684
	5	3-1:	0.00	66.67	33.33	47.359	3-1:	0.000	1.607

No.	Alt Prob. Numarası	MISOCP Döngü	MISOCP Sapma (%)	MISOCP Dolu Hareket ET (%)	MISOCP Boş Hareket ET (%)	MISOCP Çözüm Süresi (sn)	META Döngü	META Sapma (%)	META Çözüm Süresi (sn)
31	1	3-17:	0.00	38.81	61.19	109.672	3-17:	0.000	0.865
	2	3-17:	0.00	28.58	71.42	107.406	3-17:	0.000	1.178
	3	3-9:	0.00	35.29	64.71	105.218	3-18:	0.000	1.352
	4	3-4:	0.00	42.86	57.14	116.438	3-4:	0.000	0.648
	5	3-1:	0.00	50.00	50.00	138.891	3-1:	0.000	0.857
32	1	3-17:	0.00	54.21	45.79	95.703	3-17:	0.000	1.009
	2	3-17:	0.00	36.63	63.37	105.641	3-17:	0.004	1.934
	3	3-18:	0.00	52.17	47.83	105.375	3-18:	0.000	1.169
	4	3-4:	0.00	60.00	40.00	98.859	3-10:	5.000	0.883
	5	3-1:	0.00	66.67	33.33	86.609	3-1:	0.563	0.563
33	1	3-3:	0.00	40.00	60.00	67.578	3-3:	0.000	0.466
	2	3-3:	0.00	40.00	60.00	73.328	3-3:	0.000	2.171
	3	3-3:	0.00	40.00	60.00	77.406	3-3:	0.001	1.161
	4	3-1:	0.00	50.00	50.00	67.141	3-1:	0.000	0.579
	5	3-1:	0.00	50.00	50.00	55.985	3-1:	0.000	0.734
34	1	3-3:	0.00	57.14	42.86	40.047	3-3:	0.000	0.723
	2	3-3:	0.00	45.65	54.35	40.875	3-3:	0.000	2.102
	3	3-3:	0.00	45.65	54.35	39.937	3-3:	0.000	0.869
	4	3-1:	0.00	51.92	48.08	42.016	3-2:	0.000	0.505
	5	3-1:	0.00	66.67	33.33	45.062	3-1:	0.000	1.245
35	1	3-3:	0.00	40.00	60.00	88.828	3-3:	0.000	0.382
	2	3-1:	0.00	50.00	50.00	69.187	3-1:	0.000	0.978
	3	3-1:	0.00	50.00	50.00	66.25	3-1:	0.000	0.593
	4	3-1:	0.00	50.00	50.00	68.672	3-1:	0.000	0.728
	5	3-1:	0.00	50.00	50.00	59.328	3-1:	0.000	0.547
36	1	3-3:	0.00	57.14	42.86	98.906	3-3:	0.000	0.310
	2	3-1:	0.00	55.75	44.25	74.125	3-1:	0.000	0.594
	3	3-1:	0.00	55.75	44.25	30.5	3-1:	0.000	0.645
	4	3-1:	0.00	55.75	44.25	28.546	3-1:	0.000	0.707
	5	3-1:	0.00	66.67	33.33	56.703	3-1:	0.000	0.625
37	1	3-3:	0.00	39.94	60.06	99.078	3-3:	0.000	0.323
	2	3-1:	0.00	50.00	50.00	33.094	3-1:	0.001	0.547
	3	3-1:	0.00	50.00	50.00	65.937	3-1:	0.001	0.631
	4	3-1:	0.00	50.00	50.00	58.235	3-1:	0.001	0.619
	5	3-1:	0.00	50.00	50.00	53.734	3-1:	0.000	0.739
38	1	3-3:	0.00	57.08	42.92	122	3-3:	0.001	0.323
	2	3-1:	0.00	55.75	44.25	73.86	3-1:	0.001	0.684
	3	3-1:	0.00	55.75	44.25	75.968	3-1:	0.001	0.620
	4	3-1:	0.00	55.75	44.25	65.532	3-1:	0.001	0.531
	5	3-1:	0.00	66.67	33.33	60.844	3-1:	0.000	0.492
39	1	3-1:	0.00	50.00	50.00	120.297	3-1:	0.000	0.531
	2	3-1:	0.00	50.00	50.00	63.266	3-1:	0.000	0.546
	3	3-1:	0.00	50.00	50.00	68.953	3-1:	0.000	0.533
	4	3-1:	0.00	50.00	50.00	64.203	3-1:	0.001	0.518
	5	3-1:	0.00	50.00	50.00	57.828	3-1:	0.000	0.639
40	1	3-1:	0.00	66.67	33.33	135.188	3-1:	0.000	0.531
	2	3-1:	0.00	55.75	44.25	67.078	3-1:	0.000	0.655
	3	3-1:	0.00	55.75	44.25	65.89	3-1:	0.001	0.508
	4	3-1:	0.00	55.75	44.25	59.891	3-1:	0.001	0.525
	5	3-1:	0.00	66.67	33.33	59.75	3-1:	0.000	0.531
41	1	3-17:	0.00	39.69	60.31	97.766	3-17:	0.000	0.779
	2	3-7:	0.00	28.57	71.43	112.469	3-17:	0.000	1.15
	3	3-9:	0.00	35.42	64.60	111.062	3-18:	0.007	1.071
	4	3-4:	0.00	42.86	57.14	98.843	3-4:	0.000	0.813
	5	3-1:	0.00	50.00	50.00	89.188	3-1:	0.000	0.811
42	1	3-17:	0.00	55.47	44.53	40.047	3-17:	0.000	0.931
	2	3-7:	0.00	33.51	66.49	40.875	3-7:	0.002	1.462
	3	3-9:	0.00	51.01	48.99	39.937	3-9:	0.002	1.298
	4	3-4:	0.00	60.00	40.00	42.016	3-4:	0.000	1.45
	5	3-1:	0.00	66.67	33.33	45.062	3-1:	0.000	0.532
43	1	3-3:	0.00	40.00	60.00	86.391	3-3:	0.000	0.531
	2	3-3:	0.00	40.00	60.00	72.735	3-3:	0.000	1.862
	3	3-3:	0.00	40.00	60.00	68.328	3-3:	0.000	0.56
	4	3-1:	0.00	50.00	50.00	61.485	3-1:	0.001	0.492
	5	3-1:	0.00	50.00	50.00	57.578	3-1:	0.000	0.54
44	1	3-3:	0.00	57.14	42.86	133.687	3-3:	0.000	0.421
	2	3-3:	0.00	45.65	54.35	70.859	3-3:	0.000	2.18
	3	3-3:	0.00	45.65	54.35	31.688	3-3:	0.000	1.334
	4	3-1:	0.00	55.75	44.25	71.422	3-1:	0.001	0.798
	5	3-1:	0.00	66.67	33.33	21.985	3-1:	0.000	0.53
45	1	3-3:	0.00	40.00	60.00	105	3-3:	0.000	0.375
	2	3-1:	0.00	50.00	50.00	65.765	3-1:	0.000	0.503
	3	3-1:	0.00	50.00	50.00	66.532	3-1:	0.001	0.499
	4	3-1:	0.00	50.00	50.00	66.672	3-1:	0.002	0.737
	5	3-1:	0.00	50.00	50.00	55.297	3-1:	0.000	0.902

No.	Alt Prob. Numarası	MISOCP Döngü	MISOCP Sapma (%)	MISOCP Dolu Hareket ET (%)	MISOCP Boş Hareket ET (%)	MISOCP Çözüm Süresi (sn)	META Döngü	META Sapma (%)	META Çözüm Süresi (sn)
46	1	3-3:	0.00	57.14	42.86	113.782	3-3:	0.000	0.328
	2	3-1:	0.00	55.75	44.25	70.281	3-1:	0.000	0.498
	3	3-1:	0.00	55.75	44.25	67.141	3-1:	0.000	0.5
	4	3-1:	0.00	55.75	44.25	61.453	3-1:	0.003	0.548
	5	3-1:	0.00	66.67	33.33	62.985	3-1:	0.000	0.954
47	1	3-3:	0.00	39.94	60.06	109.719	3-1:	0.000	0.344
	2	3-1:	0.00	50.00	50.00	39	3-1:	0.001	0.528
	3	3-1:	0.00	50.00	50.00	73.609	3-1:	0.001	0.538
	4	3-1:	0.00	50.00	50.00	65.203	3-1:	0.000	0.544
	5	3-1:	0.00	50.00	50.00	52.657	3-1:	0.000	0.531
48	1	3-3:	0.00	57.08	42.92	1048.954	3-1:	0.001	0.358
	2	3-1:	0.00	55.75	44.25	65.438	3-1:	0.001	0.578
	3	3-1:	0.00	55.75	44.25	67.078	3-1:	0.001	0.501
	4	3-1:	0.00	55.75	44.25	58.016	3-1:	0.001	0.535
	5	3-1:	0.00	66.67	33.33	57.093	3-1:	0.000	0.708
49	1	3-1:	0.00	50.00	50.00	108.828	3-1:	0.000	0.776
	2	3-1:	0.00	50.00	50.00	60.859	3-1:	0.000	0.512
	3	3-1:	0.00	50.00	50.00	29.688	3-1:	0.000	0.844
	4	3-1:	0.00	50.00	50.00	57.735	3-1:	0.001	0.526
	5	3-1:	0.00	50.00	50.00	56.719	3-1:	0.000	0.508
50	1	3-1:	0.00	66.67	33.33	124.14	3-1:	0.000	0.431
	2	3-1:	0.00	55.75	44.25	44.61	3-1:	0.001	0.505
	3	3-1:	0.00	55.75	44.25	69.782	3-1:	0.001	0.530
	4	3-1:	0.00	55.75	44.25	57.094	3-1:	0.001	0.534
	5	3-1:	0.00	66.67	33.33	56.672	3-1:	0.000	0.546
51	1	3-17:	0.00	42.18	57.82	102.047	3-17:	0.000	1.031
	2	3-6:	0.00	28.58	71.42	101.828	3-17:	0.000	1.545
	3	3-18:	0.00	35.53	64.48	77.235	3-18:	0.007	1.329
	4	3-4:	0.00	42.86	57.14	74.703	3-4:	0.000	0.732
	5	3-1:	0.00	50.00	50.00	95.766	3-1:	0.000	0.785
52	1	3-17:	0.00	58.34	41.66	87.546	3-17:	0.000	0.607
	2	3-6:	0.00	36.63	63.37	68.25	3-19:	7.632	3.369
	3	3-9:	0.00	49.80	50.20	93.531	3-19:	3.756	1.928
	4	3-4:	0.00	60.00	40.00	78.578	3-4:	0.000	1.241
	5	3-1:	0.00	66.67	33.33	96.047	3-1:	0.000	0.552
53	1	3-3:	0.00	41.00	59.00	116.547	3-3:	0.000	0.468
	2	3-3:	0.00	40.00	60.00	68.297	3-3:	0.000	1.489
	3	3-3:	0.00	40.00	60.00	70.39	3-3:	0.000	0.640
	4	3-1:	0.00	50.00	50.00	62.657	3-1:	0.000	0.520
	5	3-1:	0.00	50.00	50.00	52.844	3-1:	0.000	0.572
54	1	3-3:	0.00	56.49	43.51	40.047	3-3:	0.000	0.640
	2	3-3:	0.00	45.65	54.35	40.875	3-3:	0.00	2.003
	3	3-3:	0.00	45.65	54.35	39.937	3-3:	0.000	1.350
	4	3-1:	0.00	55.75	44.25	42.016	3-1:	0.000	0.677
	5	3-1:	0.00	66.67	33.33	45.062	3-1:	0.000	0.593
55	1	3-3:	0.00	40.00	60.00	106.406	3-3:	0.000	0.390
	2	3-1:	0.00	50.00	50.00	62.625	3-1:	0.000	0.503
	3	3-1:	0.00	50.00	50.00	58.687	3-1:	0.001	0.555
	4	3-1:	0.00	50.00	50.00	58.685	3-1:	0.002	0.544
	5	3-1:	0.00	50.00	50.00	56.609	3-1:	0.000	1.060
56	1	3-3:	0.00	57.14	42.86	115.109	3-3:	0.000	0.344
	2	3-1:	0.00	55.75	44.25	66.062	3-1:	0.000	0.543
	3	3-1:	0.00	55.75	44.25	66.25	3-1:	0.000	0.594
	4	3-1:	0.00	55.75	44.25	60.656	3-1:	0.003	0.530
	5	3-1:	0.00	66.67	33.33	54.422	3-1:	0.000	0.539
57	1	3-3:	0.00	39.94	60.06	114.672	3-3:	0.000	0.322
	2	3-1:	0.00	50.00	50.00	63.485	3-1:	0.001	0.546
	3	3-1:	0.00	50.00	50.00	68.359	3-1:	0.002	0.592
	4	3-1:	0.00	50.00	50.00	60.407	3-1:	0.001	0.692
	5	3-1:	0.00	50.00	50.00	55.64	3-1:	0.000	0.739
58	1	3-3:	0.00	57.08	42.92	102.672	3-3:	0.001	0.313
	2	3-1:	0.00	55.75	44.25	69.703	3-1:	0.001	0.509
	3	3-1:	0.00	55.75	44.25	61.156	3-1:	0.001	0.534
	4	3-1:	0.00	55.75	44.25	63.625	3-1:	0.001	0.528
	5	3-1:	0.00	66.67	33.33	58.016	3-1:	0.000	0.520
59	1	3-1:	0.00	50.00	50.00	107.531	3-1:	0.000	0.719
	2	3-1:	0.00	50.00	50.00	60.188	3-1:	0.000	0.564
	3	3-1:	0.00	50.00	50.00	66.328	3-1:	0.000	0.516
	4	3-1:	0.00	50.00	50.00	61.875	3-1:	0.001	0.692
	5	3-1:	0.00	50.00	50.00	60.672	3-1:	0.000	0.592
60	1	3-1:	0.00	66.67	33.33	100.172	3-1:	0.007	0.391
	2	3-1:	0.00	55.75	44.25	66.984	3-1:	0.000	0.658
	3	3-1:	0.00	55.75	44.25	64.687	3-1:	0.001	0.671
	4	3-1:	0.00	55.75	44.25	58.078	3-1:	0.001	0.539
	5	3-1:	0.00	66.67	33.33	57.938	3-1:	0.000	0.529

Tablo 12’te verilen sonuçlarda META’nın ilgili problemde optimal döngüyü bulduğunda iyi sonuçlar verdiği görülebilir. Ancak, ilgili döngü bulunup bulunmaması algoritmada kullanılan bazı parametrelere bağlı olarak değişmektedir. META’da döngülerin bulunması komşuluk arama algoritmasındaki adım sayısına göre değişmektedir. Enerji tüketimi için yapılan komşuluk aramasında belirli bir adım sayısı boyunca eldeki çözüm iyileşmezse komşuluk arama durur. Ayrıca, bu komşuluk arama sırasında tavlama benzetimi metasezgiseli de kullanılmaktadır. Dolayısıyla, komşuluk arama sırasında kullanılan adım sayısı parametresi, tavlama benzetiminde kullanılan sıcaklık ve soğuma eğrisine göre optimal döngünün bulunma ihtimali değişmektedir. Adım sayısı büyük seçildiğinde istenen çözüme daha yakın bir çözüm bulunurken, çözüm süresinden feragat edilmesi gerekmektedir. Ayrıca, seçilen sıcaklık değeri ve soğuma eğrisine göre de bulunan çözümün kalitesi değişmektedir. Bahsedilen parametrelerin değerlerine içerisinde bulunan sisteme göre karar verilmelidir. Şayet, elde edilecek bir çözümde en iyi çözümden küçük bir sapmaya dahi duyarlı ise çözüm süresi çok önem arz etmeyebilir. Benzer şekilde, eğer çözümdeki sapmalar belirli bir boyuta kadar kabul edilebilir ise ve karar verici çözüm süresi açısından çok fazla esnekliğe sahip değilse bu parametrelerin değerleri ona uygun seçebilir.

Bununla beraber, Tablo 12’teki gibi MISOCP ile 20 dakikadan daha kısa sürede çözülen problemler için META ile çözüm aramak çok makul olmayabilir. Ancak, MISOCP ile daha büyük boyutlu problemler için makul sürede çözüm alınamamaktadır. Bununla birlikte, daha büyük boyutlu problemler için Tablo 13’te verilen ve sezgisel algoritmanın çözüm süresi açısından performansı hakkında fikir sahibi olmamızı sağlayan 5-makineli problemlerde, META’nın çözümü algoritmada bulunan döngü için tespit edilen optimal enerji tüketim değeri ile karşılaştırılarak META’nın çözüm kalitesi hakkında fikir edinilmiştir. Bununla beraber, algoritmada bulunan döngünün optimal olma garantisi yoktur. Son olarak, yine algoritmanın çözüm süresinin problem boyutuna göre değişiminin anlaşılması için Tablo 14’te 10-makineli robotik hücreler için oluşturulan test problemlerine ait çözüm süreleri verilmiştir.

Sonuç olarak, 3-makineli sistemler için çevrim zamanı amacı açısından ortaya atılan teorinin enerji tüketimi amacı ele alındığında ve robot hareket hızlarının karar değişkeni olduğu sistemlerde geçerli olup olmadığı incelenmiştir. Ardından, enerji tüketimi amaç fonksiyonu açısından ele alınan problemin optimal çözümüne ait robot aktivite sıralamasının 1-birim döngülerin tekrarı şeklinde olması gerektiği anlaşılmıştır. Burada aksine bulunan örnek olarak 1 numaralı problemin¹ 4. alt problemi verilebilir. Açıkça görülebilir ki, 3–5: 1-birim döngülerin tekrarı şeklinde değildir ve bu problemde optimal enerji tüketimi 56 birimdir. 1-birim döngülerin tekrarı şeklinde olan döngüler arasında en iyi enerji tüketim değerine sahip olan döngü is 60 birimlik bir enerji tüketim değerine sahiptir. Dolayısıyla, optimal n -birim döngü ($n = 3$) en iyi 1-birim döngüden yaklaşık % 7,15 daha iyi bir enerji tüketim değerine sahiptir. Buradan, enerji tüketimi açısından optimal çözümün 1-birim döngülerin tekrarı şeklinde olması gerektiği görülebilir.

¹Probleme ait parametreler şöyledir: makinelerin işlem süreleri sırasıyla $f_1 : 14$, $f_2 : 14$ ve $f_3 : 14$, makineler arası mesafeler özdeş ve 2 birim, $C_f = C_e = 1$, $\varepsilon = 1$, $low = 1$, $up = 3$, $\overline{Ct} = 62$, $\lambda = 34$

5-7:

$\{A_0, A_4, A_5, A_3, A_4, A_2, A_1, A_0, A_3, A_5, A_2, A_4, A_5, A_1, A_0, A_3, A_4, A_5, A_2, A_3, A_1, A_0, A_2, A_4, A_5, A_3, A_1, A_2, A_0, A_1\}$

5-8:

$\{A_0, A_4, A_5, A_3, A_4, A_5, A_2, A_1, A_0, A_3, A_4, A_5, A_2, A_3, A_4, A_5, A_1, A_2, A_0, A_1, A_3, A_4, A_5, A_0, A_2, A_3, A_1, A_2, A_0, A_1\}$

5-9:

$\{A_0, A_3, A_5, A_4, A_2, A_1, A_0, A_3, A_5, A_4, A_2, A_5, A_3, A_1, A_0, A_2, A_4, A_3, A_5, A_1, A_0, A_2, A_1, A_4, A_3, A_5, A_4, A_0, A_2, A_1\}$

5-10:

$\{A_0, A_3, A_5, A_4, A_2, A_1, A_0, A_3, A_5, A_2, A_4, A_3, A_5, A_1, A_0, A_2, A_4, A_3, A_5, A_4, A_1, A_0, A_2, A_1, A_3, A_5, A_4, A_0, A_2, A_1\}$

5-11:

$\{A_0, A_3, A_5, A_4, A_5, A_2, A_3, A_1, A_0, A_2, A_4, A_5, A_3, A_1, A_0, A_2, A_4, A_3, A_5, A_4, A_1, A_0, A_2, A_1, A_3, A_5, A_4, A_2, A_0, A_1\}$

5-12:

$\{A_0, A_2, A_5, A_1, A_0, A_3, A_2, A_4, A_3, A_5, A_4, A_1, A_0, A_2, A_1, A_0, A_3, A_2, A_5, A_4, A_3, A_1, A_0, A_2, A_1, A_5, A_4, A_3, A_5, A_4\}$

5-13:

$\{A_0, A_2, A_1, A_0, A_3, A_2, A_4, A_3, A_5, A_4, A_1, A_0, A_2, A_1, A_0, A_3, A_2, A_5, A_4, A_3, A_1, A_0, A_2, A_1, A_5, A_4, A_3, A_5, A_4, A_5\}$

5-14:

$\{A_0, A_2, A_4, A_3, A_5, A_4, A_5, A_1, A_0, A_2, A_3, A_1, A_0, A_2, A_4, A_5, A_3, A_4, A_5, A_1, A_0, A_2, A_3, A_1, A_0, A_2, A_4, A_5, A_3, A_1\}$

5-15:

$\{A_0, A_1, A_0, A_2, A_3, A_4, A_5, A_1, A_2, A_3, A_0, A_1, A_2, A_4, A_5, A_3, A_4, A_5, A_0, A_1, A_2, A_3, A_0, A_1, A_2, A_4, A_5, A_3, A_4, A_5\}$

5-16:

$\{A_0, A_3, A_5, A_4, A_2, A_1, A_3, A_5, A_4, A_5, A_2, A_0, A_1, A_3, A_4, A_5, A_2, A_0, A_1, A_0, A_3, A_4, A_2, A_1, A_0, A_3, A_5, A_4, A_2, A_1\}$

5-17:

$\{A_0, A_3, A_5, A_4, A_5, A_2, A_1, A_3, A_4, A_5, A_2, A_0, A_1, A_3, A_2, A_4, A_5, A_3, A_0, A_1, A_0, A_2, A_4, A_3, A_5, A_4, A_1, A_0, A_2, A_1\}$

5-18:

$\{A_0, A_3, A_5, A_4, A_5, A_2, A_1, A_3, A_4, A_5, A_2, A_3, A_0, A_1, A_2, A_4, A_5, A_3, A_4, A_5, A_0, A_1, A_0, A_2, A_3, A_4, A_1, A_2, A_0, A_1\}$

5-19:

$\{A_0, A_4, A_3, A_5, A_4, A_2, A_1, A_0, A_3, A_5, A_4, A_5, A_2, A_1, A_3, A_0, A_2, A_1, A_0, A_4, A_3, A_5, A_2, A_4, A_3, A_5, A_1, A_0, A_2, A_1\}$

5-20:

$\{A_0, A_3, A_5, A_4, A_2, A_1, A_0, A_3, A_5, A_4, A_5, A_2, A_3, A_1, A_0, A_2, A_1, A_4, A_5, A_3, A_0, A_2, A_4, A_3, A_5, A_4, A_1, A_0, A_2, A_1\}$

5-21:

$\{A_0, A_3, A_5, A_4, A_5, A_2, A_1, A_0, A_3, A_4, A_5, A_2, A_3, A_4, A_5, A_1, A_2, A_0, A_1, A_0, A_3, A_4, A_5, A_2, A_3, A_4, A_1, A_2, A_0, A_1\}$

5-22:

$\{A_0, A_1, A_3, A_5, A_4, A_0, A_2, A_1, A_3, A_5, A_4, A_0, A_2, A_1, A_3, A_5, A_4, A_0, A_2, A_1, A_3, A_5, A_4, A_0, A_2, A_1, A_3, A_5, A_4, A_2\}$

5-23:

$\{A_0, A_1, A_3, A_5, A_4, A_5, A_2, A_0, A_1, A_3, A_4, A_5, A_2, A_3, A_0, A_1, A_0, A_2, A_1, A_4, A_5, A_3, A_4, A_2, A_0, A_1, A_3, A_5, A_4, A_2\}$

5-24:

$\{A_0, A_1, A_3, A_5, A_4, A_2, A_0, A_1, A_3, A_5, A_4, A_2, A_0, A_1, A_3, A_5, A_4, A_2, A_0, A_1, A_3, A_5, A_4, A_2, A_0, A_1, A_3, A_5, A_4, A_2, A_0, A_1, A_3, A_5, A_4, A_2\}$

5-25:

$\{A_0, A_1, A_3, A_5, A_4, A_5, A_2, A_0, A_1, A_3, A_4, A_5, A_2, A_3, A_4, A_5, A_0, A_1, A_2, A_0, A_1, A_3, A_4, A_5, A_2, A_0, A_1, A_3, A_4, A_2\}$

5-26:

$\{A_0, A_1, A_3, A_5, A_4, A_5, A_2, A_0, A_1, A_3, A_4, A_5, A_2, A_0, A_1, A_3, A_4, A_5, A_2, A_0, A_1, A_3, A_4, A_5, A_2, A_0, A_1, A_3, A_4, A_5, A_2, A_0, A_1, A_3, A_4, A_2\}$

5-27:

$\{A_0, A_1, A_2, A_3, A_5, A_4, A_0, A_1, A_0, A_2, A_1, A_3, A_5, A_4, A_2, A_3, A_5, A_4, A_0, A_1, A_0, A_2, A_1, A_3, A_5, A_4, A_2, A_3, A_5, A_4\}$

5-28:

$\{A_0, A_1, A_2, A_3, A_4, A_5, A_0, A_1, A_2, A_0, A_1, A_3, A_4, A_5, A_2, A_3, A_4, A_5, A_0, A_1, A_2, A_0, A_1, A_3, A_4, A_5, A_2, A_3, A_4, A_5\}$

5-29:

$\{A_0, A_1, A_0, A_4, A_5, A_3, A_4, A_5, A_2, A_1, A_3, A_4, A_5, A_2, A_3, A_4, A_5, A_0, A_1, A_2, A_3, A_4, A_5, A_0, A_1, A_2, A_3, A_0, A_1, A_2\}$

5-30:

$\{A_0, A_2, A_1, A_3, A_5, A_0, A_2, A_4, A_1, A_0, A_3, A_2, A_5, A_4, A_1, A_0, A_3, A_5, A_4, A_2, A_5, A_1, A_0, A_3, A_2, A_1, A_4, A_3, A_5, A_4\}$

5-31:

$\{A_0, A_2, A_1, A_3, A_5, A_4, A_2, A_0, A_1, A_3, A_5, A_4, A_2, A_0, A_1, A_0, A_3, A_5, A_4, A_5, A_2, A_1, A_0, A_3, A_2, A_1, A_4, A_3, A_5, A_4\}$

5-32:

$\{A_0, A_3, A_4, A_5, A_2, A_3, A_4, A_5, A_1, A_2, A_3, A_0, A_1, A_0, A_2, A_4, A_5, A_3, A_1, A_2, A_4, A_5, A_3, A_0, A_1, A_0, A_2, A_4, A_5, A_1\}$

5-33:

$\{A_0, A_4, A_3, A_5, A_2, A_4, A_3, A_5, A_4, A_1, A_0, A_2, A_1, A_0, A_3, A_5, A_4, A_2, A_1, A_3, A_5, A_4, A_2, A_5, A_3, A_0, A_1, A_0, A_2, A_1\}$

5-34:

$\{A_0, A_4, A_5, A_3, A_2, A_4, A_3, A_5, A_4, A_1, A_0, A_2, A_1, A_0, A_3, A_5, A_4, A_5, A_2, A_1, A_3, A_2, A_4, A_5, A_3, A_0, A_1, A_0, A_2, A_1\}$

No.	m	n	P	d	$C_l - C_e$	Alt Prob. Numarası	META Döngü	META Sapma (%)	META Çözüm Süresi (sn)	Enerji Tasarrufu (%)
62	5	5	Eşit	ÖT	$C_l > C_e$	1	5-1:	0.009	3.068	23.55
						2	5-6:	0.039	8.140	
						3	5-7:	0.000	8.080	
						4	5-8:	0.000	6.282	
						5	5-5:	0.000	10.858	
63	5	5	Eşit	ÖT	$C_l = C_e$	1	5-9:	0.001	2.257	12.66
						2	5-10:	0.003	2.671	
						3	5-11:	0.001	3.544	
						4	5-11:	0.010	2.835	
						5	5-5:	0.000	7.530	
64	5	5	Eşit	ÖT	$C_l > C_e$	1	5-12:	0.004	1.690	8.12
						2	5-13:	0.003	3.195	
						3	5-14:	0.004	3.287	
						4	5-14:	0.002	2.122	
						5	5-15:	0.000	1.656	
65	5	5	Eşit	ÖT	$C_l = C_e$	1	5-16:	0.004	2.579	6.87
						2	5-16:	0.001	1.965	
						3	5-17:	0.006	3.436	
						4	5-18:	0.042	2.433	
						5	5-18:	0.116	2.312	
66	5	5	Eşit	ÖT	$C_l > C_e$	1	5-19:	0.000	2.699	13.05
						2	5-20:	0.000	4.129	
						3	5-20:	0.000	2.132	
						4	5-21:	0.012	2.869	
						5	5-21:	0.051	2.775	
67	5	5	Eşit	ÖT	$C_l = C_e$	1	5-22:	0.522	2.358	3.10
						2	5-23:	0.027	3.285	
						3	5-25:	0.290	3.579	
						4	5-25:	0.103	3.177	
						5	5-25:	0.000	2.763	
68	5	5	Eşit	ÖT	$C_l > C_e$	1	5-22:	0.002	2.552	4.52
						2	5-24:	0.008	3.996	
						3	5-26:	0.040	3.180	
						4	5-26:	0.006	2.702	
						5	5-26:	0.022	2.746	
69	5	5	Eşit	ÖT	$C_l = C_e$	1	5-27:	0.002	3.262	9.53
						2	5-28:	0.014	3.641	
						3	5-28:	0.052	2.229	
						4	5-5:	0.028	4.377	
						5	5-5:	0.000	2.328	
70	5	5	Eşit	ÖT	$C_l > C_e$	1	5-27:	0.014	2.880	4.52
						2	5-28:	0.563	3.538	
						3	5-28:	0.658	2.945	
						4	5-5:	0.560	3.288	
						5	5-5:	0.256	3.663	
71	5	5	KV	ÖT	$C_l = C_e$	1	5-1:	0.036	2.751	44.46
						2	5-2:	0.000	8.168	
						3	5-36:	0.418	5.692	
						4	5-32:	0.000	4.137	
						5	5-29:	0.000	2.665	
72	5	5	KV	ÖT	$C_l > C_e$	1	5-33:	0.052	3.336	43.92
						2	5-34:	0.000	5.072	
						3	5-34:	0.000	7.396	
						4	5-34:	0.000	4.029	
						5	5-35:	0.000	5.913	
73	5	5	KV	ÖT	$C_l = C_e$	1	5-30:	0.000	2.057	13.20
						2	5-31:	0.022	3.681	
						3	5-37:	0.209	2.728	
						4	5-37:	0.010	3.114	
						5	5-5:	0.152	4.504	

No.	m	n	P	d	$C_l - C_e$	Alt Prob. Numarası	META Döngü	META Sapma (%)	META Çözüm Süresi (sn)	Enerji Tasarrufu (%)
74	5	5	KV	ÖT	$C_l > C_e$	1	5-22:	0.007	2.578	3.55
						2	5-24:	0.016	3.589	
						3	5-26:	0.418	2.841	
						4	5-26:	0.442	2.531	
						5	5-5:	0.684	4.207	
75	5	5	KV	ÖT	$C_l = C_e$	1	5-22:	0.023	2.578	3.17
						2	5-24:	0.070	3.589	
						3	5-26:	0.188	2.841	
						4	5-26:	0.591	2.531	
						5	5-5:	0.000	4.207	
76	5	5	KV	ÖT	$C_l > C_e$	1	5-22:	0.002	3.269	4.62
						2	5-24:	0.045	4.704	
						3	5-26:	0.146	4.543	
						4	5-26:	0.148	3.208	
						5	5-5:	0.001	4.297	
77	5	5	KV	ÖT	$C_l = C_e$	1	5-27:	0.124	1.339	9.65
						2	5-5:	0.002	2.576	
						3	5-5:	0.000	2.363	
						4	5-5:	0.001	2.539	
						5	5-5:	0.000	1.859	
78	5	5	KV	ÖT	$C_l > C_e$	1	5-27:	0.014	2.964	12.96
						2	5-5:	0.001	2.859	
						3	5-5:	0.000	2.623	
						4	5-5:	0.000	2.831	
						5	5-5:	0.002	2.422	
79	5	5	KV	ÖT	$C_l = C_e$	1	5-5:	0.001	2.595	17.55
						2	5-5:	0.000	1.668	
						3	5-4:	0.000	1.411	
						4	5-5:	0.004	1.172	
						5	5-5:	0.000	1.412	
80	5	5	KV	ÖT	$C_l > C_e$	1	5-5:	0.010	2.804	20.85
						2	5-5:	0.002	2.047	
						3	5-4:	0.000	1.776	
						4	5-5:	0.003	1.786	
						5	5-5:	0.000	1.992	
81	5	5	BV	ÖT	$C_l = C_e$	1	5-1:	0.002	1.758	61.11
						2	5-2:	0.157	3.021	
						3	5-36:	0.015	3.157	
						4	5-32:	0.000	4.186	
						5	5-1:	0.001	6.240	
82	5	5	BV	ÖT	$C_l > C_e$	1	5-1:	0.004	2.044	57.67
						2	5-34:	0.027	3.816	
						3	5-34:	0.036	3.585	
						4	5-34:	0.012	2.684	
						5	5-5:	0.010	3.599	
83	5	5	BV	ÖT	$C_l = C_e$	1	5-6:	0.112	1.810	13.76
						2	5-17:	0.143	2.996	
						3	5-17:	0.101	3.306	
						4	5-11:	0.056	4.356	
						5	5-5:	0.000	2.236	
84	5	5	BV	ÖT	$C_l > C_e$	1	5-6:	0.132	1.711	15.47
						2	5-17:	0.034	2.494	
						3	5-17:	0.040	3.153	
						4	5-11:	0.007	2.559	
						5	5-5:	0.002	1.992	
85	5	5	BV	ÖT	$C_l = C_e$	1	5-22:	0.017	1.649	4.94
						2	5-24:	0.020	3.409	
						3	5-26:	0.116	2.657	
						4	5-26:	0.070	2.386	
						5	5-5:	0.003	4.693	

No.	m	n	P	d	$C_l - C_e$	Alt Prob. Numarası	META Döngü	META Sapma (%)	META Çözüm Süresi (sn)	Enerji Tasarrufu (%)
86	5	5	BV	ÖT	$C_l > C_e$	1	5-22:	0.002	2.349	7.17
						2	5-24:	0.012	4.631	
						3	5-26:	0.001	4.011	
						4	5-26:	0.007	3.557	
						5	5-5:	0.000	3.442	
87	5	5	BV	ÖT	$C_l = C_e$	1	5-24:	0.047	2.497	14.98
						2	5-26:	0.022	3.881	
						3	5-26:	0.006	3.583	
						4	5-26:	0.002	4.209	
						5	5-5:	0.001	4.040	
88	5	5	BV	ÖT	$C_l > C_e$	1	5-24:	0.069	1.626	19.16
						2	5-26:	0.103	4.117	
						3	5-26:	0.028	4.042	
						4	5-26:	0.009	3.834	
						5	5-5:	0.001	3.960	
89	5	5	BV	ÖT	$C_l = C_e$	1	5-37:	0.412	1.726	14.91
						2	5-37:	0.256	2.690	
						3	5-37:	0.046	3.309	
						4	5-37:	0.003	2.583	
						5	5-5:	0.004	2.846	
90	5	5	BV	ÖT	$C_l > C_e$	1	5-37:	0.264	2.085	18.21
						2	5-37:	0.147	2.256	
						3	5-37:	0.037	2.417	
						4	5-37:	0.006	2.210	
						5	5-5:	0.002	2.081	
91	5	5	Eşit	GT	$C_l = C_e$	1	5-1:	0.509	3.948	56.33
						2	5-2:	0.402	2.546	
						3	5-3:	0.267	3.473	
						4	5-36:	0.058	3.596	
						5	5-5:	0.011	3.014	
92	5	5	Eşit	GT	$C_l > C_e$	1	5-1:	0.127	1.213	57.38
						2	5-1:	0.114	1.785	
						3	5-2:	0.254	3.205	
						4	5-3:	0.098	2.362	
						5	5-5:	0.006	3.183	
93	5	5	Eşit	GT	$C_l = C_e$	1	5-38:	0.054	4.444	3.45
						2	5-33:	0.146	3.699	
						3	5-33:	0.114	2.655	
						4	5-33:	0.068	6.510	
						5	5-5:	0.006	2.774	
94	5	5	Eşit	GT	$C_l > C_e$	1	5-39:	0.007	1.572	3.85
						2	5-19:	0.046	2.510	
						3	5-19:	0.124	3.625	
						4	5-6:	0.002	3.097	
						5	5-5:	0.001	3.892	
95	5	5	Eşit	GT	$C_l = C_e$	1	5-40:	0.005	1.781	4.40
						2	5-40:	0.058	3.264	
						3	5-41:	0.063	3.242	
						4	5-41:	0.004	4.011	
						5	5-5:	0.000	4.671	
96	5	5	Eşit	GT	$C_l > C_e$	1	5-42:	0.002	2.044	6.40
						2	5-42:	0.017	3.090	
						3	5-42:	0.000	3.014	
						4	5-11:	0.007	3.462	
						5	5-5:	0.000	3.211	
97	5	5	Eşit	GT	$C_l = C_e$	1	5-24:	0.000	2.283	21.78
						2	5-24:	0.042	2.718	
						3	5-24:	0.010	3.139	
						4	5-28:	0.027	3.060	
						5	5-28:	0.014	3.413	

No.	m	n	P	d	$C_l - C_e$	Alt Prob. Numarası	SA Döngü	SA Sapma (%)	SA Çözüm Süresi (sn)	Enerji Tasarrufu (%)
98	5	5	Eşit	GT	$C_l > C_e$	1	5-24:	0.004	1.864	25.59
						2	5-26:	0.103	4.539	
						3	5-26:	0.054	4.369	
						4	5-26:	0.022	3.744	
						5	5-5:	0.001	2.751	
99	5	5	Eşit	GT	$C_l = C_e$	1	5-24:	0.043	2.215	18.00
						2	5-24:	0.011	3.761	
						3	5-5:	0.001	4.335	
						4	5-5:	0.000	3.914	
						5	5-5:	0.002	3.376	
100	5	5	Eşit	GT	$C_l > C_e$	1	5-24:	0.059	2.398	21.20
						2	5-24:	0.020	4.276	
						3	5-5:	0.000	3.913	
						4	5-5:	0.002	4.217	
						5	5-5:	0.001	3.992	
101	5	5	KV	GT	$C_l = C_e$	1	5-1:	0.009	0.829	35.17
						2	5-1:	0.004	1.491	
						3	5-39:	0.016	3.503	
						4	5-4:	0.061	6.754	
						5	5-5:	0.000	4.809	
102	5	5	KV	GT	$C_l > C_e$	1	5-1:	0.001	1.230	34.04
						2	5-1:	0.007	1.731	
						3	5-39:	0.035	2.132	
						4	5-4:	0.107	3.489	
						5	5-5:	0.042	3.341	
103	5	5	KV	GT	$C_l = C_e$	1	5-1:	0.008	1.102	1.53
						2	5-16:	0.063	3.735	
						3	5-16:	0.046	3.201	
						4	5-21:	0.004	3.824	
						5	5-5:	0.002	3.603	
104	5	5	KV	GT	$C_l > C_e$	1	5-1:	0.038	1.740	3.28
						2	5-16:	0.288	3.009	
						3	5-16:	0.161	3.746	
						4	5-21:	0.078	3.044	
						5	5-5:	0.006	2.990	
105	5	5	KV	GT	$C_l = C_e$	1	5-24:	0.023	2.189	7.85
						2	5-24:	0.017	4.485	
						3	5-4:	0.014	2.621	
						4	5-4:	0.002	3.425	
						5	5-5:	0.001	3.428	
106	5	5	KV	GT	$C_l > C_e$	1	5-24:	0.049	1.886	11.07
						2	5-24:	0.006	4.347	
						3	5-4:	0.048	3.847	
						4	5-4:	0.002	2.803	
						5	5-5:	0.001	3.638	
107	5	5	KV	GT	$C_l = C_e$	1	5-24:	0.015	2.009	6.44
						2	5-25:	0.051	4.541	
						3	5-25:	0.012	4.232	
						4	5-4:	0.019	3.276	
						5	5-5:	0.002	3.778	
108	5	5	KV	GT	$C_l > C_e$	1	5-24:	0.021	2.188	8.67
						2	5-4:	0.016	3.204	
						3	5-4:	0.007	2.917	
						4	5-4:	0.021	2.620	
						5	5-5:	0.001	3.152	
109	5	5	KV	GT	$C_l = C_e$	1	5-26:	0.022	1.981	14.68
						2	5-5:	0.007	3.805	
						3	5-5:	0.000	3.297	
						4	5-5:	0.001	4.073	
						5	5-5:	0.000	3.623	

No.	m	n	P	d	$C_l - C_e$	Alt Prob. Numarası	SA Döngü	SA Sapma (%)	SA Çözüm Süresi (sn)	Enerji Tasarrufu (%)
110	5	5	KV	GT	$C_l > C_e$	1	5-26:	0.004	2.273	17.89
						2	5-5:	0.003	3.969	
						3	5-5:	0.004	3.965	
						4	5-5:	0.002	2.787	
						5	5-5:	0.000	3.581	
111	5	5	BV	GT	$C_l = C_e$	1	5-1:	0.022	2.438	63.22
						2	5-6:	0.053	2.099	
						3	5-6:	0.018	3.202	
						4	5-6:	0.015	2.354	
						5	5-32:	0.094	2.615	
112	5	5	BV	GT	$C_l > C_e$	1	5-1:	0.048	1.652	63.08
						2	5-6:	0.112	2.329	
						3	5-6:	0.058	3.475	
						4	5-6:	0.004	4.314	
						5	5-32:	0.027	2.961	
113	5	5	BV	GT	$C_l = C_e$	1	5-1:	0.330	1.841	0.01
						2	5-1:	0.083	1.863	
						3	5-6:	0.036	2.229	
						4	5-6:	0.003	2.170	
						5	5-5:	0.018	7.259	
114	5	5	BV	GT	$C_l > C_e$	1	5-1:	0.071	2.104	0.02
						2	5-6:	0.057	1.595	
						3	5-6:	0.014	4.886	
						4	5-6:	0.008	2.892	
						5	5-5:	0.013	2.749	
115	5	5	BV	GT	$C_l = C_e$	1	5-24:	0.019	1.736	22.66
						2	5-24:	0.032	4.483	
						3	5-28:	0.006	3.817	
						4	5-4:	0.018	2.730	
						5	5-5:	0.003	3.301	
116	5	5	BV	GT	$C_l > C_e$	1	5-24:	0.157	1.702	26.84
						2	5-24:	0.054	4.111	
						3	5-28:	0.096	3.340	
						4	5-4:	0.024	3.947	
						5	5-5:	0.002	3.103	
117	5	5	BV	GT	$C_l = C_e$	1	5-24:	0.004	1.822	17.78
						2	5-26:	0.177	2.993	
						3	5-26:	0.024	3.023	
						4	5-26:	0.021	3.287	
						5	5-5:	0.006	3.556	
118	5	5	BV	GT	$C_l > C_e$	1	5-24:	0.088	2.103	20.99
						2	5-26:	0.131	3.264	
						3	5-26:	0.056	3.300	
						4	5-26:	0.007	3.050	
						5	5-5:	0.001	3.770	
119	5	5	BV	GT	$C_l = C_e$	1	5-24:	0.049	2.738	14.00
						2	5-26:	0.255	3.121	
						3	5-26:	0.163	4.861	
						4	5-5:	0.084	3.649	
						5	5-5:	0.011	2.982	
120	5	5	BV	GT	$C_l > C_e$	1	5-24:	0.143	2.231	17.17
						2	5-26:	0.135	2.975	
						3	5-26:	0.098	3.805	
						4	5-5:	0.032	4.119	
						5	5-5:	0.006	3.541	

5-makinelik robotik hücrelerde 5-birim döngülerin ele alındığı test problemleri için çözüm sürelerini incelediğimizde çözüm süresi bakımından META'nın hala hızlı olduğunu görüyoruz. Ayrıca, daha önce 3-makinelik sistemlerde bulunan bir döngü için hız hesaplama algoritmasının iyi sonuçlar verdiğine dair gözlemimizin 5-makinelik test problemlerinin sonuçları ile de desteklendiği görülebilir. Problemler için tasarruf miktarları ele alınan her bir problemde optimal çevrim zamanı hesaplanamadığından META ile bulunan en küçük çevrim zamanı için hesaplanmıştır. Bu tasarruf miktarları, robot hareket hızlarının kontrol edilmesinin enerji tüketimini azalttığı ve çevrim zamanını deęiřtirmedięi durumlara örnek olarak verilebilir; çünkü ilgili problemlerde o döngü için aynı çevrim zamanı üst sınırı sağlanmaktadır. Dolayısıyla, her ne kadar farklı bir döngünün daha düşük bir çevrim zamanı verme ihtimali bulursa da sezgisel algoritmanın bulduğu ilk noktadaki döngü için robotun hızı üst sınıra eşit olacak şekilde hareket ettięi durum ele alınarak tasarruf değeri hesaplanabilir. 3-makinelik test problemlerine benzer olarak 5-makinelik problemlerde de özellikle her bir farklı problem grubunun ilk iki probleminde tasarruf değerinun daha fazla olduęu görülebilir. Dolayısıyla, bu sonuçlar da makinelerin birbirlerine daha yakın olduęu ve robotun parça taşıma sırasında görece daha az yol katettięi sistemlerde daha fazla enerji tasarrufu yapma olasılıęının olduęu yorumumuzu güçlendirmektedir. Ayrıca, 5-makine 5-birim döngü için hazırlanan bu problemler ikinci derece konik model ile makul süre içerisinde (30 dakika zaman limiti altında) olurlu bir çözüme ulaşamazken, META algoritmamız ile saniyeler içerisinde çözebilmektedir.

Son olarak, Tablo 14'te detayları verilen 10-makinelik robotik hücre problemleri kullanılarak META'nın daha büyük boyutlu problemlerde çözüm süresinin deęişimi gözlemlenmiştir. Verilen 10-makinelik test problemleri için 10-birim döngü ele alınmış ve META ile çözdürülmüştür. Problemlere ait algoritma çözüm süreleri ise aşağıdaki tabloda verilmiştir:

Tablo 14. META'nın Enerji Tüketimi Amacı Açısından 10-Makineli Sistemlerde Performans Analizi

No.	m	n	P	d	$C_l - C_e$	META Çözüm Süresi (sn)				
						Alt Prob. 1	Alt Prob. 2	Alt Prob. 3	Alt Prob. 4	Alt Prob. 5
121	10	10	Eşit	ÖT	$C_l = C_e$	8.555	12.738	15.402	17.748	11.887
122	10	10	Eşit	ÖT	$C_l = C_e$	7.053	18.878	19.037	19.803	18.604
123	10	10	Eşit	ÖT	$C_l = C_e$	17.026	20.739	24.999	43.774	25.374
124	10	10	Eşit	ÖT	$C_l = C_e$	20.019	23.795	22.997	26.739	20.428
125	10	10	Eşit	ÖT	$C_l = C_e$	12.730	20.373	16.556	16.043	15.257
126	10	10	Eşit	ÖT	$C_l = C_e$	18.732	20.016	21.372	22.775	38.226
127	10	10	Eşit	ÖT	$C_l = C_e$	16.870	22.804	27.910	26.517	21.199
128	10	10	Eşit	ÖT	$C_l = C_e$	11.728	21.919	19.372	20.344	23.826
129	10	10	Eşit	ÖT	$C_l = C_e$	12.403	20.945	18.118	18.002	19.401
130	10	10	Eşit	ÖT	$C_l = C_e$	33.134	21.046	20.190	19.919	20.110
131	10	10	KV	ÖT	$C_l = C_e$	12.365	20.812	24.887	42.840	21.603
132	10	10	KV	ÖT	$C_l = C_e$	22.061	17.282	23.067	22.067	17.012
133	10	10	KV	ÖT	$C_l = C_e$	12.647	18.454	19.031	18.869	18.618
134	10	10	KV	ÖT	$C_l = C_e$	13.961	19.191	22.173	38.069	21.555
135	10	10	KV	ÖT	$C_l = C_e$	19.206	29.398	23.309	20.558	24.140
136	10	10	KV	ÖT	$C_l = C_e$	21.175	12.196	11.168	20.415	19.051
137	10	10	KV	ÖT	$C_l = C_e$	14.335	21.845	20.442	23.973	20.674
138	10	10	KV	ÖT	$C_l = C_e$	13.809	20.922	18.27	17.115	18.840
139	10	10	KV	ÖT	$C_l = C_e$	33.744	22.020	19.400	13.722	19.325
140	10	10	KV	ÖT	$C_l = C_e$	18.634	20.932	17.966	19.906	20.198
141	10	10	BV	ÖT	$C_l = C_e$	14.867	32.655	20.904	20.640	18.316
142	10	10	BV	ÖT	$C_l = C_e$	20.240	14.129	20.954	14.537	18.054
143	10	10	BV	ÖT	$C_l = C_e$	15.952	18.977	25.916	22.859	27.290
144	10	10	BV	ÖT	$C_l = C_e$	13.985	27.592	24.995	19.951	22.650
145	10	10	BV	ÖT	$C_l = C_e$	15.529	18.545	19.706	19.176	18.523
146	10	10	BV	ÖT	$C_l = C_e$	16.822	19.054	19.486	17.438	19.136
147	10	10	BV	ÖT	$C_l = C_e$	15.490	19.600	18.640	21.082	18.035

No.	m	n	P	d	$C_l - C_e$	META Çözüm Süresi (sn)				
						Alt Prob. 1	Alt Prob. 2	Alt Prob. 3	Alt Prob. 4	Alt Prob. 5
148	10	10	BV	ÖT	$C_l = C_e$	18.017	25.047	26.485	32.306	22.483
149	10	10	BV	ÖT	$C_l = C_e$	18.843	21.347	26.230	19.540	19.318
150	10	10	BV	ÖT	$C_l = C_e$	15.304	18.640	19.787	18.671	21.042
151	10	10	Eşit	GT	$C_l = C_e$	5.472	11.073	18.397	17.628	18.189
152	10	10	Eşit	GT	$C_l = C_e$	13.146	18.304	17.529	22.201	24.321
153	10	10	Eşit	GT	$C_l = C_e$	21.749	17.907	17.857	18.723	20.230
154	10	10	Eşit	GT	$C_l = C_e$	21.151	17.789	21.306	16.881	21.296
155	10	10	Eşit	GT	$C_l = C_e$	23.186	22.576	22.542	17.159	20.840
156	10	10	Eşit	GT	$C_l = C_e$	32.341	20.704	26.483	24.061	23.478
157	10	10	Eşit	GT	$C_l = C_e$	15.405	30.449	23.285	42.226	22.0007
158	10	10	Eşit	GT	$C_l = C_e$	11.307	24.822	20.532	23.423	19.462
159	10	10	Eşit	GT	$C_l = C_e$	12.296	18.572	17.208	22.460	18.549
160	10	10	Eşit	GT	$C_l = C_e$	10.641	18.819	18.798	18.881	17.523
161	10	10	KV	GT	$C_l = C_e$	12.590	16.006	16.179	16.926	16.814
162	10	10	KV	GT	$C_l = C_e$	21.621	20.206	19.646	17.036	19.501
163	10	10	KV	GT	$C_l = C_e$	12.018	22.694	17.796	19.639	35.269
164	10	10	KV	GT	$C_l = C_e$	13.505	18.213	20.739	26.080	20.933
165	10	10	KV	GT	$C_l = C_e$	9.033	12.914	10.827	10.925	13.076
166	10	10	KV	GT	$C_l = C_e$	13.049	11.368	9.293	9.103	9.579
167	10	10	KV	GT	$C_l = C_e$	7.599	11.564	11.703	11.286	13.421
168	10	10	KV	GT	$C_l = C_e$	8.496	12.651	11.418	12.918	12.369
169	10	10	KV	GT	$C_l = C_e$	7.813	11.237	20.362	11.003	11.144
170	10	10	KV	GT	$C_l = C_e$	8.007	10.414	10.065	10.131	12.488
171	10	10	BV	GT	$C_l = C_e$	6.193	9.019	14.451	14.131	10.056
172	10	10	BV	GT	$C_l = C_e$	10.244	10.772	10.853	12.308	11.161
173	10	10	BV	GT	$C_l = C_e$	11.593	13.144	11.793	10.190	11.231
174	10	10	BV	GT	$C_l = C_e$	23.278	9.564	9.979	9.999	9.765
175	10	10	BV	GT	$C_l = C_e$	9.074	12.106	10.303	12.551	13.792
176	10	10	BV	GT	$C_l = C_e$	9.454	10.239	12.017	14.169	15.123
177	10	10	BV	GT	$C_l = C_e$	7.027	9.792	11.522	11.092	11.773
178	10	10	BV	GT	$C_l = C_e$	14.619	16.319	10.092	9.863	13.316
179	10	10	BV	GT	$C_l = C_e$	15.963	8.182	10.997	11.028	9.957
180	10	10	BV	GT	$C_l = C_e$	7.698	9.944	10.141	13.763	11.578

Tablo 14'te verilen sonuçlara baktığımızda 10-makine 10-birim test problemlerinde tüm pareto-etkin çözümlerin 1 dakikadan daha kısa sürede bulunduğu gözlemlenmiştir. Elde edilen sonuçlar ışığında sezgisel algoritmanın çözüm süresi açısından etkin bir algoritma olduğunu göstermektedir; çünkü ikinci derece konik model 5-makine 5-birim problemleri için bile 30 dakika içerisinde olurlu bir çözüm bulamazken, META algoritması 1 dakikadan kısa süre içerisinde iyi bir çözüm vermektedir. Tüm pareto-etkin çözümler ise bir problem için 5 dakikadan kısa sürede elde edilmektedir.

META algoritmasında kullanılan parametreler 3-makinele test problemlerinden elde edilen sonuçlar doğrultusunda iyi çözümler verecek şekilde belirlenmiştir. Yine de, çözüm süresinin ikinci planda olduğu durumlarda, karar verici META algoritmasında kullanılan adım sayısı parametrelerinin değerini arttırarak çözüm kalitesinin iyileşme ihtimalini arttırabilir.

3.4 Robot Hız Kontrolü Uygulamasının Sağladığı Tasarruf

Enerji tüketimi amacının ele alınmasının, endüstride robotun enerji tüketim değerinin azaltılmasını destekleyeceğini daha önce öngörmüştük. Bu başlıkta, MISOCP kullanılarak 3-makinele sistem içeren test problemleri iki farklı amaç için çözdürülmüştür. Buradan, elde edilen sonuçlara bakarak MISOCP'nin çözüm süresi ve hız belirleme algoritmasının etkinliği hakkında fikir edinilmiştir.

Tablo 15. Enerji Tüketimi En Küçükleme Amacının Sistemin Enerji Tasarrufuna Etkisi

No.	MISOCP ÇZ En	MISOCP ÇZ En	MISOCP ÇZ En	MISOCP ET En	MISOCP ET En	MISOCP ET En	Tasarruf Miktarı (%)
	Küç. Döngü	Küç. Sapma (%)	Küç. Çözüm Süresi (sn)	Küç. Döngü	Küç. Sapma (%)	Küç. Çözüm Süresi (sn)	
1	3-17:	0.00	40.453	3-17:	0.00	38.797	14.81
2	3-17:	0.00	43.375	3-17:	0.00	41.125	11.11
3	3-7:	0.00	49.188	3-6:	0.00	113.719	0.00
4	3-7:	0.00	49.687	3-7:	0.00	98.891	0.00
5	3-3:	0.00	67.906	3-5:	0.00	107.906	0.00
6	3-3:	0.00	31.500	3-5:	0.00	77.296	0.00
7	3-9:	0.00	26.485	3-5:	0.00	96.203	0.00
8	3-9:	0.00	556.323	3-14:	0.00	868.344	0.00
9	3-1:	0.00	49.672	3-1:	0.00	72.938	0.00
10	3-1:	0.00	22.015	3-1:	0.00	854.953	0.00
11	3-17:	0.00	50.359	3-17:	0.00	51.313	47.70
12	3-17:	0.00	47.140	3-17:	0.00	48.828	45.32
13	3-7:	0.00	45.578	3-6:	0.00	97.828	0.00
14	3-6:	0.00	30.797	3-6:	0.00	144.985	0.00
15	3-3:	0.00	89.422	3-3:	0.00	754.312	0.00
16	3-3:	0.00	115.016	3-3:	0.00	137.375	0.00
17	3-3:	0.00	79.281	3-3:	0.00	762.093	0.00
18	3-3:	0.00	777.718	3-3:	0.00	140.937	0.00
19	3-1:	0.00	112.234	3-1:	0.00	256.47	0.00
20	3-1:	0.00	599.797	3-1:	0.00	115.922	0.00
21	3-17:	0.00	35.782	3-17:	0.00	40.016	46.37
22	3-17:	0.00	35.781	3-17:	0.00	39.219	42.73
23	3-17:	0.00	81.844	3-6:	0.00	186.781	0.00
24	3-6:	0.00	82.984	3-7:	0.00	170.422	0.00
25	3-3:	0.00	94.532	3-14:	0.00	170.953	0.00
26	3-5:	0.00	100.922	3-3:	0.00	136.203	0.00
27	3-5:	0.00	81.078	3-5:	0.00	129	0.00
28	3-5:	0.00	102.188	3-5:	0.00	667.25	0.00
29	3-1:	0.00	69.563	3-1:	0.00	113.812	0.00
30	3-1:	0.00	54.110	3-1:	0.00	145.172	0.00
31	3-17:	0.00	51.500	3-17:	0.00	57.141	33.28
32	3-17:	0.00	44.016	3-17:	0.00	45.032	28.29
33	3-3:	0.00	129.921	3-3:	0.00	72.031	0.00
34	3-3:	0.00	104.078	3-3:	0.00	68.39	0.00
35	3-3:	0.00	87.032	3-3:	0.00	99.75	0.00
36	3-3:	0.00	78.765	3-3:	0.00	127.656	0.00
37	3-3:	0.00	74.313	3-3:	0.00	106.078	0.00
38	3-3:	0.00	1016.97	3-3:	0.00	104.891	0.00
39	3-1:	0.00	630.265	3-1:	0.00	103.25	0.00
40	3-1:	0.00	118.094	3-1:	0.00	108.468	0.00
41	3-17:	0.00	44.640	3-17:	0.00	38.343	36.72
42	3-17:	0.00	47.266	3-17:	0.00	37.578	31.55
43	3-3:	0.00	36.750	3-3:	0.00	101.172	0.00
44	3-3:	0.00	114.562	3-3:	0.00	87.797	0.00
45	3-3:	0.00	61.469	3-3:	0.00	103.109	0.00
46	3-3:	0.00	87.875	3-3:	0.00	108.89	0.00
47	3-3:	0.00	69.187	3-3:	0.00	115.672	0.00
48	3-3:	0.00	30.766	3-3:	0.00	129.531	0.00
49	3-1:	0.00	769.468	3-1:	0.00	93.953	0.00

No.	MISOCP ÇZ En Küç. Döngü	MISOCP ÇZ En Küç. Sapma (%)	MISOCP ET En Küç. Çözüm Süresi (sn)	MISOCP ET En Küç. Döngü	MISOCP ET En Küç. Sapma (%)	MISOCP ET En Küç. Çözüm Süresi (sn)	Tasarruf Miktarı (%)
50	3-1:	0.00	961.937	3-1:	0.00	96.375	0.00
51	3-17:	0.00	36.812	3-17:	0.00	57.391	40.43
52	3-17:	0.00	37.094	3-17:	0.00	40.203	33.36
53	3-3:	0.00	53.281	3-3:	0.00	99.797	4.13
54	3-3:	0.00	146.266	3-3:	0.00	118.156	4.02
55	3-3:	0.00	108.922	3-3:	0.00	78.25	0.00
56	3-3:	0.00	86.907	3-3:	0.00	119.016	0.00
57	3-3:	0.00	67.313	3-3:	0.00	99.063	0.00
58	3-3:	0.00	90.281	3-3:	0.00	106.156	0.00
59	3-1:	0.00	74.313	3-1:	0.00	99.375	0.00
60	3-1:	0.00	958.032	3-1:	0.00	110.578	0.00

Problemlerde tasarruf miktarlarına baktığımızda girdi ve çıktı noktalarının birbirine yakın olduğu durumda, yani makinelerin yerleştirileceği alanın daha küçük olduğu sistemlerde, robotun hareket hızlarının değiştirilmesiyle yapılan tasarruf miktarının daha fazla olduğu test problemlerinden görülebilir. Test problemlerinde her bir problem tipindeki ilk problemlerde (sırasıyla 1, 2, 11, 12, 21, 22, 31, 32, 41, 42, 51 ve 52 numaralı problemler) enerji tüketim amacını ele almakla sistemde tasarruf sağlanabilir. Ancak, girdi ve çıktı noktalarının birbirinden daha uzak olduğu durumlarda enerji tüketim amacının ele alınmasının katkı sağlamayabileceği görülebilir. Dolayısıyla, gerçek hayatta böyle bir sistem ele alındığında probleme ait işlem süreleri ve makineler arasındaki mesafeler (veya girdi ve çıktı noktaları arasındaki mesafe) karşılaştırılarak enerji tüketim amacının ele alınması ile tasarruf sağlayıp sağlamayacağı anlaşılabilir.

3.5 Sonuç

Projenin bu kısmında tek tip parça üreten m makineli bir robotik hücrede tek tutuculu robot için çizelgeleme problemi çalışıldı. Makinelerdeki işlem süreleri ve makineler arası mesafeler verildiğinde çevrim zamanı ve robot enerji tüketimi hedeflerini minimize etme amacıyla yöntemler geliştirildi. İki kriterli bu problemde ϵ -kısıt yöntemiyle çevrim zamanı hedefi için bir üst limit belirlenerek enerji tüketimini minimize eden matematiksel programlama modeli (MINLP) geliştirildi. Modeldeki doğrusal olmayan terimler ikinci derece konik eşitsizliklerle ifade edilerek daha etkin olan MISOCP modeli önerildi. Matematiksel programlama modellerinin yetersiz kaldığı büyük boyutlu problemlerin çözümü için ise yaklaşık etkin çözüm kümesi türeten bir sezgisel algoritma (META) geliştirildi. Geliştirilen yöntemler hesaplamalı deneylerle sınandı. META'nın karar vericiye kısa sürede çok sayıda etkin çözüm alternatifi türetebildiği ve türettiği çözümlerin çözüm kalitesinin tatmin edici olduğu gözlemlendi.

4. ÇİFT TUTUCULU MALZEME TAŞIYICI ROBOTLU HÜCRELERDE ENERJİ DUYARLI ÇİZELGELEME

Dönem içerisinde İş paketi 3 kapsamında çift tutuculu robotlu hücreler ele alınmıştır. Bir önceki iş paketinde sunulan matematiksel model doğrulanmış ve geliştirilmiştir. Doğrusal yapıda olmayan bu matematiksel modelin çözümü, 2 makineli sistemlerde dahi uzun zaman almaktadır. Bu nedenle, doğrusal yapıda olmayan matematiksel modele alternatif olarak ikinci derece konik programlama formülasyonu sunulmuştur. Yapılan deneysel çalışma ile asıl model ve konik olarak yeniden formüle edilmiş model karşılaştırılmıştır. Sonuç olarak, ikinci derece konik programlama ile daha hızlı sonuçlar elde edildiği gözlemlenmiştir.

4.1 Giriş

Robotik hücrelerin etkinliğini artırmak için üretimde tercih edilen çift tutuculu robotların en önemli özelliği, bir makineden parça boşalttıktan sonra ilgili makineye parça yüklemesi için bir önceki makineye gitmesinin gerekmemesidir. Eğer diğer tutucuda bu makineye yüklenecek bir parça varsa zaman kaybetmeden yükleme yapabilir. Bu özellik hücrenin verimini artırırken, olası döngü sayısının artırdığı için problemi daha karmaşık hale getirmektedir. Tek tutuculu sistemde 2 makinede 2 adet 1-birim döngüsü mümkün iken, Sethi vd. [93] çalışmalarında çift tutuculu robotik hücrelerde 2-makineli bir üretim hücresi için 52 farklı döngü olduğunu ortaya çıkartmışlardır. Söz konusu çalışmada 2-makineli 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 .

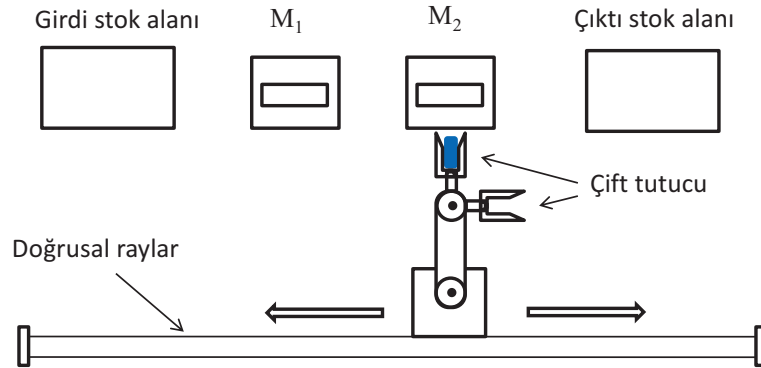
Drobouchevitch vd. [94] m -makineli çift tutuculu robotik hücre için 1-birim döngü sayısını hesaplayan bir algoritma geliştirmiştir. Bu algoritmaya göre, 10-makineli sistemde 6.4×10^{11} adet 1-birim döngüsü bulunmaktadır. Ayrıca, tek ve çift tutuculu robotik hücreler verimlilik açısından karşılaştırılmış ve çift tutuculu robotik hücrelerin daha üretken olduğu gösterilmiştir. Geismar vd. [95], tek ve çift tutuculu robotik hücreler için ayrı ayrı algoritmalar geliştirmişlerdir. Drobouchevitch vd. [24] 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 sıralamasını bulunmasına odaklanılmıştır. En kötü durumda 3/2-yaklaşık performans sağlayan bir sezgisel algoritma geliştirilmiştir . Sriskandarajah vd. [96], 2-makineli çift tutuculu ve farklı tip parça üreten bir sistem ele almışlardır. Robot döngüsü sabitlense bile parça sıralama probleminin bir çok 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. [97] 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 eniyi çevrimleri belirlemişlerdir. Foumani

ve Jenab [98], 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.

Bunların dışında, robotun birbiriyle bağımlı olarak hareket eden iki kolunun bulunduğu robotik hücreler ([99],[100]), üretim ortamında darboğaz olan makinelere paralel aynı özellikte makinelerin eklenmesiyle ortaya çıkan hibrid robotik hücreler ([101], [102]), makineler arasında ara stok alanının bulunduğu robotik hücreler ([103], [94], [104]), robot sayısının birden fazla olduğu, ve/veya bir karar değişkeni olduğu robotik hücreler ([105], [106]) literatürde ele alınmıştır. 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 çalışmaların tamamında robot hareket süreleri bilinen birer problem parametresidir.

Çift tutuculu bir robotik hücre Şekil 9'de gösterilmiştir. Bu tip robotlar aynı anda iki parça taşıyabilmekte veya bir makineyi boşattıktan sonra, başka bir makineye gitmeye gerek olmadan aynı makineyi diğer tutucuda bulunan parça ile yeniden yükleyebilmektedirler. Fakat bu hareketi yapabilmek için robotun aktif olan tutucusunu değiştirmesi gerekmektedir. Bu değişim süresi θ ile gösterilecektir.



Şekil 9. 2 makineli çift tutuculu robotik hücre

Bu proje kapsamında, bu tip hücrelerdeki robot hareket sıralaması ve robot hareket hızlarının belirlenmesi problemlerinin eş zamanlı olarak çözülmesi amaçlanmaktadır. Bu rapor döneminde, iş paketine uygun olarak önceki dönem geliştirilmiş olan ve m -makineli sistemler için probleme çözüm türetebilen genel matematiksel modelin doğrulaması yapılmış ve modelin eksiklik ve hataları giderilmiştir. Ayrıca, geliştirilen modelin çözüm süresinin problem büyüklüğüyle çok hızlı artmasından dolayı daha etkin bir formülasyon geliştirme çalışmaları yapılmış ve konik karesel programlama formülasyonu geliştirilmiştir. Bu raporda öncelikle bir önceki dönem geliştirilmiş olan taslak modelin son hali tekrar sunulacak, daha sonra ikinci derece konik programlama formülasyonu ve bununla yapılan deneysel çalışmaların sonuçlarından bahsedilecektir.

4.2 Matematiksel Programlama Formülasyonları

Tek tutuculu robotlu sistemlerde robot bir makineyi boşalttığıında 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 geçersizdir. 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 stoğuna bırakılacak) bir parça vardır.

Herhangi bir robot döngüsü yukarıdaki aktivitelerin uygun bir sıralaması şeklinde tanımlanabilir. Uygun sıralamadan kasıt, robotun zaten dolu bir makineyi yüklememesi, zaten boş olan bir makineyi boşaltmaması, robotun her iki tutucusu da doluysa bir başka boşaltma aktivitesi yapmaması ve robotun tutucularından herhangi birisinde M_m makinesini yüklemek için bir parça yoksa, bu makineyi yüklemeye çalışmaması anlamına gelmektedir. Ayrıca, bütün aktiviteler tamamlandıktan sonra, robotun döngüdeki başlangıç pozisyonuna geri dönmesi, bütün makinelerin, döngünün başlangıcındaki durumlarına (dolu veya boş) gelmeleri ve robotun her iki tutucusunun da yine döngü başlangıcındaki durumlarına (boş veya döngü başındakiyle aynı makineye yüklenmesi gereken bir parça taşınması) gelmiş olması gerekir. 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 birisiyle 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 $g_i = 0, 1, \dots, m + 1$ durumlarından bir tanesi olabilir.

- $U_m(m + 1, g_2)$ or $U_m(g_1, m + 1)$: 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 $g_i = 0, 1, \dots, m + 1$ durumlarından herhangi bir tanesi olabilir.

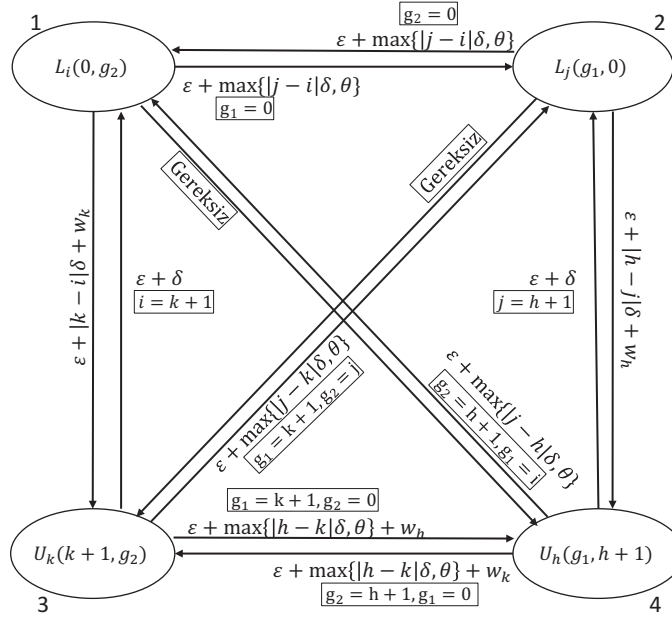
Bu tanımlamalara göre, herhangi bir çift tutuculu m -makineli robotik hücredeki bir 1-birim robot döngüsünde, giriş stoğu dahil olmak üzere her makine bir defa yüklenmeli (yani $m + 1$ adet L aktivitesi kullanılmalı), çıkış stoğu dahil her makine bir defa boşaltılmalıdır (yani $m + 1$ adet U aktivitesi 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 stoğundan 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 -makineli 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. $U_m(g_1, m + 1)$

Bu aktivitelerin, döngüde birbirlerini takip etmeleri durumunda ortaya çıkacak süreler parametrik olarak yazılabilmektedir. Bu mesafeler Şekil 10'de gösterilmiştir. Dolayısıyla, matematiksel model hem bu aktiviteleri pozisyonlara atamakta, hem de bu hareketleri gerçekleştirirken robotun kullandığı hızı belirlemektedir.



Şekil 10. Robot döngüsünde olabilecek dört alternatif aktivite arası parametrik mesafeler

Matematiksel modelde kullanılan notasyon aşağıda sunulmuştur:

Kümeler ve Parametreler:

ϵ : Makine yükleme/boşaltma zamanı

d_{mn} : M_m makinesinden M_n ($m \neq n$) makinesine mesafe

C : Enerji tüketim fonksiyonu sabiti. Bu sabit, birim mesafe başına etki eden ağırlık ve sürtünme kuvvetlerinden kaynaklanmaktadır.

P_m : M_m makinesindeki parça işleme süresi

\overline{Ct} : Çevrim zamanı için verilen bir üst limit değeri

θ : Robotun aktif tutucusunu değiştirme süresi.

G : Her tutucu değişiminde ortaya çıkan enerji tüketim değeri

$\mathcal{M} = \{0, 1, 2, \dots, a\}$: Makineler kümesi

$\mathcal{K} = \{1, 2, 3, 4\}$: Yükleme/boşaltma alternatifleri kümesi.

$\mathcal{P} = \{1, 2, \dots, h\}$: Aktivitelerin atanabileceği pozisyonların kümesi

B : Modelde kullanılan yeterince büyük bir sayı

ϵ : Modelde kullanılan yeterince küçük bir sayı

Karar Değişkenleri:

$$x_{nmp(p-1)} = \begin{cases} 1, & p \text{ pozisyonuna atanan } M_n \text{ makinesi ile } p-1 \text{ pozisyonuna atanan } M_m \\ & \text{makinesi arasında tutucu deđiřimi varsa} \\ 0, & \text{Diđer durumda} \end{cases}$$

$$y_{mpk} = \begin{cases} 1, & p \text{ pozisyonunda } M_m \text{ makinesinde bir } k \text{ tipi bir aktivite yapılıyorsa} \\ 0, & \text{Diđer durumda} \end{cases}$$

T_p : $p \in \mathcal{P}$ pozisyonuna atanan aktivitenin başlama zamanı

v_{mn} : Robotun M_m makinesinden M_n ($m \neq n$) makinesine giderken kullandığı hız

Matematiksel Model:

$$\text{Min } \sum_m \sum_n \sum_p \sum_k \sum_h y_{mpk} y_{n(p-1)h} C \delta_{mn} v_{mn}^k + \sum_m \sum_n \sum_p x_{mnp(p-1)} G + \varepsilon CT \quad (65)$$

Öyle ki

$$T_p \geq T_{p-1} + \varepsilon + \max \left\{ \theta, \frac{\delta_{mn}}{v_{mn}} \right\} - B(3 - y_{m(p-1)h} - y_{npk} - x_{mnp(p-1)}) \\ \forall m, n \in \mathcal{M}, \forall p \in \mathcal{P} : p \neq 1, \forall k, h \in \mathcal{K} : k \neq h \wedge (k, h \leq 2 \vee k, h \geq 3) \quad (66)$$

$$T_p \geq T_{p-1} + \varepsilon + \frac{\delta_{mn}}{v_{mn}} - B(2 - y_{m(p-1)h} - y_{npk}) \quad \forall m, n \in \mathcal{M}, \forall p \in \mathcal{P} : p \neq 1 \\ \forall k, h \in \mathcal{K} : (h = 1 \wedge k = 3) \vee (h = 3 \wedge k = 1) \vee (h = 2 \wedge k = 4) \vee (h = 4 \wedge k = 2) \quad (67)$$

$$T_p \geq T_{p-1} + \varepsilon + \max \left\{ \theta, \frac{\delta_{mn}}{v_{mn}} \right\} - B(3 - y_{m(p-1)h} - y_{npk} - x_{mnp(p-1)}) \quad \forall m, n \in \mathcal{M}, \forall p \in \mathcal{P} : p \neq 1 \\ \forall k, h \in \mathcal{K} : (h = 4 \wedge k = 1) \vee (h = 3 \wedge k = 2) \vee (h = 1 \wedge k = 4) \vee (h = 2 \wedge k = 3) \quad (68)$$

$$CT \geq T_p + \varepsilon + \frac{\delta_{a0}}{v_{a0}} - B(1 - y_{mpk}) \\ \forall m \in \mathcal{M}, \forall p \in \mathcal{P} : p = h, \forall k \in \mathcal{K} : k = 1 \quad (69)$$

$$CT \geq T_p + \varepsilon + \max \left\{ \frac{\delta_{a0}}{v_{a0}}, \theta \right\} - B(2 - y_{mpk} - x_{mnp0}) \\ \forall m \in \mathcal{M}, \forall p \in \mathcal{P} : p = h, \forall k \in \mathcal{K} : k \geq 2, k \neq 3 \quad (70)$$

$$T_p \geq T_r + \varepsilon + P_m - B(2 - y_{mpk} - y_{mrh}) \\ \forall p, r \in \mathcal{P} : p > r, \forall m \in \mathcal{M}, \forall k, h \in \mathcal{K} : h \leq 2, k \geq 3 \quad (71)$$

$$CT + T_p \geq T_r + \varepsilon + P_m - B(2 - y_{mpk} - y_{mrh}) \\ \forall p, r \in \mathcal{P} : p < r, \forall m \in \mathcal{M}, \forall k, h \in \mathcal{K} : h \leq 2, k \geq 3 \quad (72)$$

$$y_{113} = 1 \quad (73)$$

$$\sum_{s=r+1}^{p-1} y_{(n+1)s1} + 1 \geq y_{nr3} + y_{mp3} \quad \forall m, n \in \mathcal{M} : n \neq a, \forall p \in \mathcal{P} : p > r \quad (74)$$

$$\sum_{s=r+1}^P y_{(n+1)s1} + \sum_{s=1}^{p-1} y_{(n+1)s1} + 1 \geq y_{nr3} + y_{mp3} \quad \forall m, n \in \mathcal{M} : n \neq a, \forall p \in \mathcal{P} : p < r \quad (75)$$

$$y_{mp3} = 0 \quad \forall m \in \mathcal{M}, \forall p \in \mathcal{P} : p = h, \quad (76)$$

$$\sum_{s=r+1}^{p-1} y_{(n+1)s2} + 1 \geq y_{nr4} + y_{mp4} \quad \forall m, n \in \mathcal{M}, \forall p \in \mathcal{P} : p > r, \quad (77)$$

$$\sum_{s=r+1}^s y_{(n+1)s2} + \sum_{s=1}^{p-1} y_{(n+1)s2} + 1 \geq y_{nr4} + y_{mp4} \quad \forall m, n \in \mathcal{M} : n \leq m, \forall p \in \mathcal{P} : p < r, \quad (78)$$

$$\sum_{p=1}^h y_{(m)p3} = \sum_{p=1}^P y_{(m+1)p1} \quad \forall m \in \mathcal{M} : m \leq a \quad (79)$$

$$\sum_{p=1}^h y_{(m)p4} = \sum_{p=1}^P y_{(m+1)p2} \quad \forall m \in \mathcal{M} : m \leq a \quad (80)$$

$$\sum_{m=1}^a \left(\sum_{k=1}^b y_{mpk} \right) = 1 \quad \forall p \in \mathcal{P} \quad (81)$$

$$\sum_{k=3}^b \left(\sum_{p=1}^h y_{mpk} \right) = 1 \quad \forall m \in \mathcal{M} : m \neq a \quad (82)$$

$$\sum_{k=2}^b \left(\sum_{p=1}^h y_{mpk} \right) = 1 \quad \forall m \in \mathcal{M} : m \neq 1 \quad (83)$$

$$LB \leq v_{mn} \leq UB \quad \forall m, n \in \mathcal{M} \quad (84)$$

$$x_{mnp}, y_{mpk} \in \{0, 1\} \quad \forall m \in \mathcal{M}, \forall i \in \mathcal{S}, \forall p \in \mathcal{P}, \forall k \in \mathcal{K} \quad (85)$$

$$T_p, P_m, CT \geq 0 \quad \forall p \in \mathcal{P}, \forall m \in \mathcal{M} \quad (86)$$

Bu modelde minimize edilmek istenen amaç fonksiyonu robot hareket hızlarından kaynaklı enerji tüketimleri ve robotun tutucu değiştirme sayısına bağlı olan enerji tüketimlerinin toplamıdır. Ayrıca amaç fonksiyonunda çevrim zamanı çok küçük bir katsayı (ε) ile çarpılarak eklenmiştir. Bunun amacı, modelde yer alan bazı kısıtların doğru çalışmasının sağlanmasıdır. ε değeri çevrim zamanının enerji tüketimlerine kıyasla ikinci plana düşebileceği kadar küçük seçilmelidir.

Kısıt (66) ardışık pozisyonlara atanan aktivitelerin ikisi de yükleme veya ikisi de boşaltma aktiviteleriye, önceki pozisyona atanan aktivite tamamlanmadan sonraki aktivitenin yapılamayacağını ifade eder. $(p - 1)$ pozisyonuna atanan aktivite başladıktan sonra ε kadar sürede yükleme veya boşaltma işlemi tamamlanır. Robot daha sonra bulunduğu makinenin önünden yeni işlemi yapacağı makineye gider. Ardışık aktivitelerin ikisi de yükleme veya boşaltma olduğunda bu iki aktivitenin aynı makinede olamayacağı açıktır. Bu sebeple, robot mutlaka bir makineden diğerine gitmelidir. Diğer taraftan, yeni aktiviteyi yapabilmesi için aktif tutucusunu da değiştirmelidir. Bu işlem, robot bir makineden diğerine hareket ederken yapılabilir. Bu durumda arada geçecek süre bu iki aktiviteden süresi fazla olanın süresine eşittir. Bu kısıttaki \max terimi bunu hesaplar.

Kısıt (67), yine ardışık pozisyona atanan işlemleri ele alır fakat bu sefer bu aktivitelerin her ikisinin de yükleme veya boşaltma olmadığı ve robotun aktif tutucusunu değiştirmek zorunda olmadığı durum için aynı kısıtı ifade eder. Örneğin Şekil 10'de 1 numaralı alternatif hareket olan $L_i(0, g_2)$ 'den sonra 3 numaralı aktivite olan $U_k(k+1, g_2)$ aktivitesinin yapılması durumunda robotun aktif tutucusunu değiştirmesine gerek yoktur.

Kısıt (68), benzer bir kısıtı bu defa robotun ardışık pozisyona atanan aktiviteler arasında aktif tutucusunu değiştirmek zorunda olduğu durumlar için ifade eder. Bu duruma örnek olarak da Şekil 10'deki 1 numaralı alternatif hareket olan $L_i(0, g_2)$ 'den sonra 4 numaralı aktivite olan $U_h(g_1, h+1)$ 'in gelmesi durumu verilebilir. Kısıttaki \max terimi, aktif tutucu değişikliğinin robotun bir makineden diğerine giderken yapılabilmesinden kaynaklanmaktadır.

(69) ve (70) numaralı kısıtlar, çevrim zamanının son pozisyona atanan aktivite tamamlandıktan sonra robotun bulunduğu makinenin önünden giriş stoğuna gidişi arasındaki süreden daha büyük olması gerektiğini ifade eder. Bunlardan ilki, bu son hareket esnasında robotun aktif tutucusunu değiştirmesine gerek olmadığı durum için yazılmışken ikincisi, aktif tutucunun değişmesi gerektiği durumlar için yazılmıştır.

(71) ve (72) numaralı kısıtlar bir parçanın bir makinede boşaltılmadan önce makinedeki işleminin bitmesi gerektiğini yani o parçanın makinede en az işlem süresi kadar zaman geçirmesi gerektiğini ifade eder. Bunlardan ilkinde, makinenin yükleme aktivitesinin aktivite sıralamasında boşaltmadan daha önce yer aldığı durum için, ikincisi ise boşaltmanın yüklemeden önce yer aldığı durum için yazılmıştır. İki 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 (73), ilk pozisyona giriş stoğundan yeni bir parçanın alınması aktivitesini atar. Bu bir döngü olduğu için, döngünün hangi aktiviteyle başladığının önemi yoktur. Önemli olan, aktivitelerin nasıl sıralandığıdır. (74) ve (75) numaralı kısıtlar, Şekil 10'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, sonuçta ortaya çıkan aktivite sıralamasının olurlu olması için gereklidir. Aktivite sıralamasının 3 numaralı aktiviteyle başladığı varsayıldığı için, son pozisyona 3 numaralı aktivitenin gelmemesi gerekmektedir. Bu koşul kısıt (76) ile sağlanmaktadır.

3 numaralı aktivite ile 1 numaralı aktivite arasındaki ilişkiye benzer şekilde (77) ve (78) numaralı kısıtlar, Şekil 10'de verilen aktivitelerden 4 numaralı aktivitenin ardışık iki tekrarı arasında 2 numaralı aktivitenin en az bir defa yapılmasını sağlarlar.

Kısıtlar (79) ve (80) döngüde kullanılan 1 numaralı aktivite sayısını 3 numaralı aktivite sayısına, 4 numaralı aktivite sayısını da 2 numaralı aktivite sayısına eşitlemektedir. Bu da sonuçta çıkan döngünün olurlu olabilmesi için gereklidir.

Kısıt (81), her pozisyona bir aktivitenin atanmasını sağlar. Kısıt (82), çıkış stoğu hariç her makinenin döngü boyunca bir defa boşaltılmasını, Kısıt (83) ise giriş stoğu hariç her makinenin bir defa yüklenmesini sağlar. Kısıt (84), robot hareket hızlarının verilen alt ve üst sınırlar arasında olmasını sağlar. (85) ve (86) numaralı kısıtlar ise modelde kullanılan değişkenlere ait işaret kısıtlarıdır.

Projenin önceki iş paketinde tanımlanan bu matematiksel model, bu iş paketinde doğrulanmış ve geliştirilmiştir. Güncellenen karar değişkenleri, amaç fonksiyonu ve kısıtlar açıklamalarıyla birlikte aşağıda sunulmuştur.

Karar Değişkenleri:

$$S_{mnp} = \begin{cases} 1, & p \text{ pozisyonunda } M_m \text{ makinesinden } M_n \text{ makinesine hareket ediliyorsa} \\ 0, & \text{Diğer durumda} \end{cases}$$

Diğer bir deyişle, $S_{mnp} = \sum_k (y_{m(p-1)k}) \cdot \sum_h (y_{nph})$ eşitliğini sağlamakta ve amaç fonksiyonunda yer alan v_{mn} değişkeni ile de aşağıda ilişkilendirilerek uygun değerleri alması sağlanmaktadır.

v_{mn} : Robotun M_m makinesinden M_n ($m \neq n$) makinesine giderken kullandığı hız

a_{mn} : Robotun M_m makinesinden M_n ($m \neq n$) makinesine giderken kullandığı zaman

Tanımlanan SS_{mnp} karar değişkeni ile, amaç fonksiyonu (65)'de bulunan karar değişkenlerinin çarpıldığı ifadeler doğrusallaştırılmıştır. Bu durumda Matematiksel modelde (65) yerine (87) yazılmalıdır.

$$\text{Min } \sum_m \sum_n C \delta_{mn} v_{mn}^a + \sum_m \sum_n \sum_p x_{mnp(p-1)} G + \varepsilon CT \quad (87)$$

Kısıt (66),(68) ve (70)'de bulunan \max fonksiyonu doğrusallaştırılmıştır ve Kısıt (88) ve (89) şeklide iki ayrı kısıt şeklinde yazılmıştır. Ayrıca (66),(67) ve (68) kısıtları birleştirilerek Kısıt (90) şeklinde yazılmıştır. İlk modelde bu kısıtların ayrı ayrı yazılmasının nedeni, tutucu değişimi gereken pozisyonlar için, iki makine arasında geçen sürenin θ değişim değerinden küçük olmasını sağlamaktır. Kısıt (67) ile, tutucu değişiminin kontrolü $x_{mnp(p-1)}$ değişkeni ile yapılmıştır. Yani iki makine arasında geçen süre, v_{mn} ve $x_{mnp(p-1)}$ değişkenlerine bağlı olarak hesaplanmaktadır. Benzer şekilde kısıt (69), ve (70) de a_{mn} değişkeni yardımıyla birleştirilmiş ve kısıt (91) şeklinde yazılmıştır.

$$a_{mn} \geq \delta_{mn}/v_{mn} \quad \forall m, n \in \mathcal{M} \quad (88)$$

$$a_{mn} \geq \theta - B(1 - x_{nm(p-1)p}) \quad \forall m, n \in \mathcal{M}, \forall p \in \mathcal{P} : p \neq 1 \quad (89)$$

$$T_p \geq T_{p-1} + \epsilon + a_{mn} - B(2 - \sum_h y_{m(p-1)h} - \sum_k y_{nph}) \quad \forall m, n \in \mathcal{M}, \forall p \in \mathcal{P} : p \neq 1 \quad (90)$$

$$CT \geq T_p + \epsilon + a_{mn} - B(1 - \sum_k y_{mpk}) \quad \forall m, n \in \mathcal{M} : n = 1, \forall p \in \mathcal{P} : p = h \quad (91)$$

(92) ve (93) 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.

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

$$\forall k, h \in \mathcal{K} : (h = 4 \wedge k = 1) \vee (h = 3 \wedge k = 2) \vee (h = 1 \wedge k = 4) \vee (h = 2 \wedge k = 3) \quad (92)$$

$$x_{1n1p} + 1 \geq y_{11k} + y_{nph} \quad \forall m, n \in \mathcal{M}, \forall p \in \mathcal{P} : p \neq 1$$

$$\forall k, h \in \mathcal{K} : (h = 4 \wedge k = 1) \vee (h = 3 \wedge k = 2) \vee (h = 1 \wedge k = 4) \vee (h = 2 \wedge k = 3) \quad (93)$$

$\sum_k y_{m(p-1)k}$ ile $\sum_h y_{nph}$ değişkenlerinin çarpımı olarak tanımlanan S_{mnp} değişkeni ile ardışık iki pozisyonda, m ve n makineleri arasında hareket olup olmadığı belirlenir. (94) ve (99) arası kısıtlarda S_{mnp} 'nin tanımı doğrusal olarak yapılmıştır. Bu kısıtlardan ilk üçü, aynı çevrim içindeki robot hareketlerini, diğerleri bir çevrimin son aktivitesi ile sonraki çevrimin ilk aktivitesi arasındaki robot hareketini belirlemektedir.

$$S_{mn(p-1)} \leq \sum_h y_{m(p-1)h} \quad \forall m, n \in \mathcal{M}, \forall p \in \mathcal{P} : p \neq 1 \quad (94)$$

$$S_{mn(p-1)} \leq \sum_k y_{nph} \quad \forall m, n \in \mathcal{M}, \forall p \in \mathcal{P} : p \neq 1 \quad (95)$$

$$S_{mn(p-1)} \geq \sum_h y_{m(p-1)h} + \sum_k y_{nph} - 1 \quad \forall m, n \in \mathcal{M}, \forall p \in \mathcal{P} : p \neq 1 \quad (96)$$

$$S_{m1p} \leq \sum_h y_{mph} \quad \forall m \in \mathcal{M}, \forall p \in \mathcal{P} : p = h \quad (97)$$

$$S_{m1p} \leq \sum_k y_{11k} \quad \forall m \in \mathcal{M}, \forall p \in \mathcal{P} : p = h \quad (98)$$

$$S_{m1p} \geq \sum_h y_{mph} + \sum_k y_{11k} - 1 \quad \forall m \in \mathcal{M}, \forall p \in \mathcal{P} : p = h \quad (99)$$

Kısıt (88)'de v_{mn} değeri paydada olduğundan, modelde belirsizlik olmaması için $v_{mn} = 0$ değeri alamamaktadır. (86) ve (101) numaralı kısıtlar, çevrimde kullanılmayan ancak pozitif değer alan hız değişkenlerinin amaç fonksiyonundaki etkisinin en aza indirmektedir. Bunlardan ilki ile gerçekleşen hareketler için gerçek hız alt limiti sağlanırken, ikincisi ile hızların sıfırdan büyük olması sağlanmaktadır.

$$v_{mn} \geq LB \cdot S_{mnp} \quad \forall m, n \in \mathcal{M}, \forall p \in \mathcal{P} \quad (100)$$

$$v_{mn} \geq \varepsilon \quad \forall m, n \in \mathcal{M} \quad (101)$$

Karma Tamsayı Doğrusal Olmayan (MINLP) bir yapıda olan bu model GAMS arayüzü ile kodlanmış ve BARON çözdürücüsü ile denemeler yapılmıştır. Yapılan deneysel çalışmalardan 4.3 bölümünde bahsedilecektir. Fakat, bu modelde çözüm süresinin oldukça uzun olmasından dolayı bu modele alternatif olarak ikinci derece konik programlama modeli geliştirilmiştir. Raporun bu kısmında geliştirilen bu model açıklanacaktır.

Tamsayı İkinci Derece Konik Programlama (MISOCP) Modeli:

Bir önceki bölümde görüldüğü üzere amaç fonksiyonu (87) ve kısıt (88) doğrusal olmayan

terimler içermektedir. Bu nedenle, bu ifadelerin ikinci dereceden konik programlama yöntemleri kullanılarak ifade edilerek yine CPLEX çözücüsünde çözdürebilmek üzere matematiksel modeller geliştirilmiştir.

Amaç Fonksiyonu

Amaç fonksiyonunda bulunan, v_{mn}^a terimini ikinci dereceden şekilde yazabilmek için kullanılacak karar değişkenleri şu şekildedir:

$$w_{mn} = v_{mn}^a.$$

$q = 1$ (karesel yazım formatını sağlayabilmek için q değişkeni tanımlanmıştır ve değeri 1'e eşitlenerek çarpımlarda etkisiz olması sağlanmıştır).

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

- $a = 2$:

$$\text{Min } \sum_m \sum_n C \delta_{mn} w_{mn} + \sum_m \sum_n \sum_p x_{mnp(p-1)} G + \varepsilon CT \quad (102)$$

Öyle ki

$$q = 1 \quad (103)$$

$$v_{mn} \cdot v_{mn} \leq w_{mn} q \quad \forall m, n \in \mathcal{M} \quad (104)$$

Üssel kuvvet $a = 2$ olduğunda, kısıt (103) ve (104) modele eklenmelidir. q değişkeninin, eşitsizliğin sağ tarafına çarpan olarak eklenmesiyle, her iki taraf da karesel forma dönüştürülmüştür.

- $a = 3$:

Üssel kuvvet a 'nın 3 olduğu durumlar için amaç fonksiyonu (102) gibidir. Eklenecek kısıtların formüle edilme adımları aşağıda verilmiştir.

$$\begin{aligned} w w_{mn} \geq v_{mn}^3 &\Rightarrow w w_{mn} \geq v_{mn}^2 \cdot \frac{\delta_{mn}}{a_{mn}} \Rightarrow w w_{mn} \cdot \frac{a_{mn}}{\delta_{mn}} \geq v_{mn} \cdot v_{mn} \\ v_{mn} \cdot v_{mn} &\leq w_{mn} \frac{a_{mn}}{\delta_{mn}} \quad \forall m, n \in \mathcal{M} \end{aligned} \quad (105)$$

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

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

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

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

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

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

$$(v_{mn} - a_{mn})^2 + 4\delta_{mn} \leq (v_{mn} + a_{mn})^2 \quad \forall m, n \in \mathcal{M} \quad (106)$$

Hız üstel kuvveti $a = 3$ iken, amaç fonksiyonu (102) ve kısıtlar (105) ve (106) modele eklenmelidir.

- $a = 4$:

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

$$\text{Min} \sum_m \sum_n C\delta_{mn}w_{mn} + \sum_m \sum_n \sum_p x_{mnp(p-1)}G + \varepsilon CT \quad (107)$$

Öyle ki

$$v_{mn} \cdot v_{mn} \leq vv_{mn} \quad \forall m, n \in \mathcal{M} \quad (108)$$

$$vv_{mn} \cdot vv_{mn} \leq w_{mn} \quad \forall m, n \in \mathcal{M} \quad (109)$$

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.

Amaç fonksiyonundaki doğrusal olmayan ifadenin ikinci derece konik eşitsizliklerle yazımını gösterdikten sonra, şimdi de Kısıt (88)'deki doğrusal olmayan ifadenin dönüştürülmesi gösterilecektir.

Kısıt (88):

Kısıt (88), paydada bulunan v_{mn} terimi nedeniyle doğrusal olmayan bir yapıdadır. Bu kısıtın ikinci dereceden yeniden formüle edilme adımları aşağıdaki gibidir.

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

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

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

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

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

$$(v_{mn} - a_{mn})^2 + 4\delta_{mn} \leq (v_{mn} + a_{mn})^2 \quad \forall m, n \in \mathcal{M} \quad (110)$$

Karma tamsayı doğrusal olmayan modelde kısıt (66) ile eşitsizlik (110) değiştirildiğinde tamsayı 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.

4.3 MISOCP ve MINLP Modellerinin Hesaplamalı Karşılaştırması

Geliştirilen modellerin hesaplama performanslarını test etmek için veri kümeleri oluşturuldu. Deneylede kullanılacak veri kümelerinin boyutu makine sayısı bazında 2, 3, 4 olarak belirlendi. Ayrıca, makineler arası mesafe için farklı varsayımlar altında veriler türetili. Literatürde ardışık makinelerin arasındaki mesafenin özdeş olduğu ve ardışık olmayan makineler arasındaki mesafenin, ardışık makineler arası mesafenin toplanarak bulunduğu durum en sık kullanılan varsayımdır. Bu alternatif “özdeş toplamsal” ismiyle tanımlanacaktır. Buna göre M_i ile M_j makineleri arasındaki mesafe $\delta_{ij} = |i - j|\delta$ şeklinde bulunur. Burada δ ardışık makineler arası mesafedir. Bir diğer durum ise, ardışık makineler arası mesafelerin birbirinden farklı olabileceği fakat ardışık olmayan makineler arası mesafenin aradaki ardışık makineler arası mesafelerin toplamı şeklinde olduğu durumdur. Yani, M_i ile M_j $i < j$ makineleri arasındaki mesafe $\sum_{k=i}^{j-1} \delta_{k(k+1)}$ şeklinde bulunur. Bu durum da “genel toplamsal” şeklinde isimlendirilecektir. Bunların dışında, makine işlem sürelerinin dar ve geniş aralıklardan türetildiği iki durum gözönünde bulundurulmuştur. Elde edilen 12 farklı veri grubunda, rastgelelik unsuru göz önünde bulundurulurken, beşer veri kümesi oluşturulmuştur. Testlerde kullanılan parametrelerin dağılımı Tablo 16’de gösterilmiştir. Bu tabloda “DD” kısaltması düzgün dağılımı ifade etmektedir.

Tablo 16. Parametre Değerleri

Parametre	Dağılım	
	Özdeş Toplamsal (ÖT)	Genel Toplamsal (GT)
Mesafe	DD[1,15]	DD[1,15]
	Dar Aralık (DA)	Geniş Aralık (GA)
İşlem Süresi	DD[20,25]	DD[10,35]
θ	DD[$\min\{\frac{\delta_{mn}}{VUB}\}, \max\{\frac{\delta_{mn}}{VLB}\}$]	
ϵ	4	
Hız Limiti	Alt sınır (VLB)=1 Üst sınır (VUB)=5	

İlk aşamada geliştirilen MINLP ve MISOCP modellerinin performansları birbirleriyle karşılaştırılmıştır. Seçilen bazı veri kümeleri için $m = 2$ durumunda bu modellerin performans karşılaştırması Tablo 17’de verilmiştir. MISOCP GAMS arayüzü ve CPLEX çözücüsü ile çözdürülürken MINLP modeli yine GAMS arayüzü ve BARON çözücüsü ile çözülmüştür. Modeller için 3 saat (10800 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. Sonuçlar incelendiğinde, konik modelin en kötü durumda 10.2 saniyede optimal çözümü bulduğu görülmektedir. MINLP modeli, çözdürülen toplam 16 problemin sadece 2 tanesinde 3 saat içerisinde optimal çözümü bulabilmiştir. 4 problem için 3 saat içerisinde herhangi bir olurlu çözüm bulamamıştır. Bu nedenle bu durumlar için sapma miktarı da hesaplanamamıştır. Tablo 17’de bu durumlar MD ile gösterilmiştir. Konik model sonuçları ile karşılaştırıldığında, MINLP çözümü uzun zaman aldığından tüm veri kümeleri için MINLP koşturumu yapılmamıştır.

Tablo 17. MISOCP ve MINLP Model Çözüm Süreleri

Veri	m	İşlem	Süresi	Mesafe	Üstel Kuvvet	MISOCP		MINLP	
						Sapma (%)	Süre (sn)	Sapma (%)	Süre (sn)
1	2	DA	GT	2	0	1.61	0	270	
1	2	DA	GT	3	0	1.89	0	10492	
2	2	DA	GT	2	0	10.22	MD	ZL	
2	2	DA	GT	3	0	2.12	MD	ZL	
6	2	GA	GT	2	0	10.16	MD	ZL	
6	2	GA	GT	3	0	1.95	MD	ZL	
7	2	GA	GT	2	0	10.25	114	ZL	
7	2	GA	GT	3	0	2.33	696	ZL	
11	2	DA	ÖT	2	0	9.16	57	ZL	
11	2	DA	ÖT	3	0	2.22	319	ZL	
12	2	DA	ÖT	2	0	8.47	81	ZL	
12	2	DA	ÖT	3	0	2.17	527	ZL	
16	2	GA	ÖT	2	0	6.55	125	ZL	
16	2	GA	ÖT	3	0	2.23	679	ZL	
17	2	GA	ÖT	2	0	10.06	68	ZL	
17	2	GA	ÖT	3	0	2.19	406	ZL	

Ele alınan problem, çok kriterli bir optimizasyon problemidir. Enerji tüketimini ve çevrim zamanını aynı çözümde minimize etmek mümkün olmadığından, veri kümeleri için Pareto-etkin çözümlerin elde edilmesi gerekmektedir. Bu amaçla, ϵ -kısıt yaklaşımını uygulanmış ve çevrim zamanı hedefi modelde kısıt olarak ele alınmıştır. Kısıt kümesinde çevrim zamanı hedefi için üst limit uygulanıp, farklı üst limit değerleri ile farklı etkin çözümler elde edilebilmektedir.

Robot hızının kontrol edilebildiği robotik hücre çizelgelemede en küçük çevrim süresini bulmanın bir yolu robot tüm hareketlerde en yüksek hızıyla hareket ettiğini düşünerek çevrim süresini minimize edecek hareket sıralamasını bulmaktır. Sabit (en yüksek) hızlarla çevrim süresi minimize eden problemler Karma Tamsayılı Doğrusal bir yapıdadır. Bu model çözdürülerek olası en küçük çevrim zamanı değeri elde edilmiştir. Bu çözüm en düşük çevrim zamanına sahiptir. Fakat, bu çevrim zamanına karşılık gelen en düşük enerji tüketim değerine sahip olmayabilir. Bu sebeple, elde edilen çevrim zamanı değeri amaç fonksiyonu enerji tüketimini en küçükleme olan asıl modelde kısıt olarak eklenerek, en küçük enerji tüketimi belirlenmiştir. Bu modelden elde edilen çözüm olabilecek en düşük çevrim zamanı - en yüksek enerji tüketimi değerlerine sahip ilk etkin çözümdür. Etkin çözüm bulan modelimiz, en düşük çevrim süresi değerinden başlayarak çevrim zamanı limiti her iterasyonda belirli bir miktar gevşetilerek yeni etkin çözümler türetebilmektedir.

2 makineli veri kümelerinden elde edilen çözümler incelendiğinde aşağıda aktivite sıralamaları verilen yedi çevrimin sonuç verdiği görülmektedir.

1. $U_0(1, 0) \rightarrow L_1(0, 0) \rightarrow U_2(3, 0) \rightarrow L_3(0, 0) \rightarrow U_1(2, 0) \rightarrow L_2(0, 0)$
2. $U_0(1, 0) \rightarrow U_1(1, 2) \rightarrow L_1(0, 2) \rightarrow U_2(3, 2) \rightarrow L_3(0, 2) \rightarrow L_2(0, 0)$
3. $U_0(1, 2) \rightarrow L_1(0, 2) \rightarrow U_2(3, 2) \rightarrow L_2(3, 0) \rightarrow L_3(0, 0) \rightarrow U_1(0, 2)$
4. $U_0(1, 0) \rightarrow U_1(1, 2) \rightarrow L_1(0, 2) \rightarrow U_2(3, 2) \rightarrow L_2(3, 0) \rightarrow L_3(0, 0)$
5. $U_0(1, 2) \rightarrow L_1(0, 2) \rightarrow U_2(3, 2) \rightarrow L_2(3, 0) \rightarrow U_1(3, 2) \rightarrow L_3(0, 2)$
6. $U_0(1, 2) \rightarrow L_1(0, 2) \rightarrow U_2(3, 2) \rightarrow L_3(0, 2) \rightarrow L_2(0, 0) \rightarrow U_1(0, 2)$
7. $U_0(1, 0) \rightarrow L_1(0, 0) \rightarrow U_1(2, 0) \rightarrow L_2(0, 0) \rightarrow U_2(3, 0) \rightarrow L_3(0, 0)$

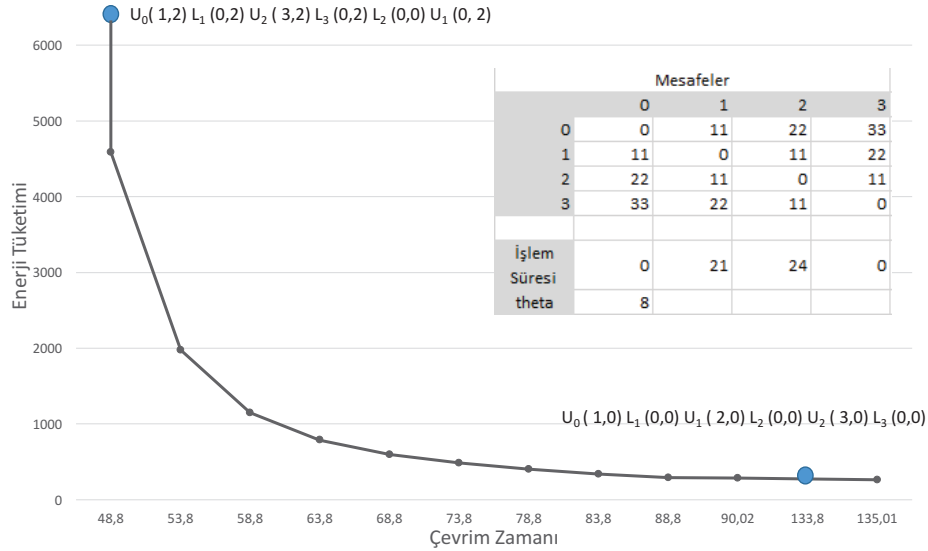
$m = 2$ olan 20 veri kümesi için, minimum çevrim zamanı \overline{CT} elde edildikten sonra, etkin çözümler her adımda \overline{CT} 5 birim artırılarak elde edilmiştir. Artırma işlemi 20 kez yapılmış, ayrıca minimum enerjiyi sağlayan maksimum çevrim zamanı da çözüm kümesine eklenmiştir. Toplamda gerçekleştirilen 440 koşturum sonucu, 2 makine için aşağıdaki sonuçlar elde edilmiştir.

- Bir veri kümesinin etkin çözümleri arasında en fazla 5 farklı 1-birim döngü yer almıştır.
- Etkin çözümlerde, 6 ve 7 numaralı 1-birim döngü mutlaka bulunmaktadır.
- 7 numaralı döngü, çevrim zamanı limiti yüksek olduğunda ortaya çıkan ve tek tutucu kullanımının yeterli olduğu (UPHILL) döngüdür.
- θ değeri büyük olduğunda, tutucu değişimi gerektirmeyen 3 numaralı döngü minimum çevrim zamanı çözümlerinde yer almaktadır.
- 4 tutucu değişimi gerektiren 2, 3 ve 4 numaralı döngüler, θ değeri küçük olduğunda minimum çevrim zamanı çözümlerinde yer almaktadır.
- 20 veri kümesi için alınan ilk çözümlerde 1, 2, 3, 4 ve 5 numaralı döngüler sırasıyla 6, 2, 1, 2 kez elde edilmiştir. 6 numaralı döngü ise başlangıç döngülerinde 8 kez elde edilmiştir. Ayrıca bu döngü, elde edilen etkin çözümlerin %73'ünde kullanılmaktadır.

Şekil 11, 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 48,8 olan iki çözüm bulunmaktadır. Bunlardan ilki, mavi renkle işaretlenmiş olan ve çevrim zamanı minimizasyon modeli ile elde edilen çözüm, diğeri ise, bu ilk modelden elde edilen çevrim zamanı değerinin enerji minimizasyonu modeline üst limit olarak eklenmesiyle elde edilen çözümdür. Bu ikinci çözüm daha düşük enerji tüketim değerine sahiptir. Dolayısıyla, ikinci çözüm ilk çözümü domine etmektedir ve ilk çözüm etkin çözümler kümesinde yer almamaktadır.

Grafik incelendiğinde, Pareto önyüzde 2 farklı döngünün önyüzün farklı noktalarında ortaya çıktığı görülmektedir. Bunlardan ilki yukarıda açıklanan 6 numaralı döngüdür. Çevrim zamanı

üst limitinin belirli bir değerden düşük olduğu noktalarda hep bu döngü en iyi sonuçları vermiştir. Çevrim zamanı üst limiti 133,8'i aştığı anda ise 7 numaralı döngü en iyi sonuçları vermeye başlamıştır. Ayrıca, robotun tüm hızlarını üst limitte kullanmasına göre, hız ayarlaması yapılarak %30 enerji tasarrufu sağlanmıştır. Tablo 18, 2 makineli bu örnek için toplamda şekilde gösterilen 12 etkin çözümü elde etmek için kullanılan çevrim zamanı üst limitlerini ve her problemin çözüm süresini göstermektedir.



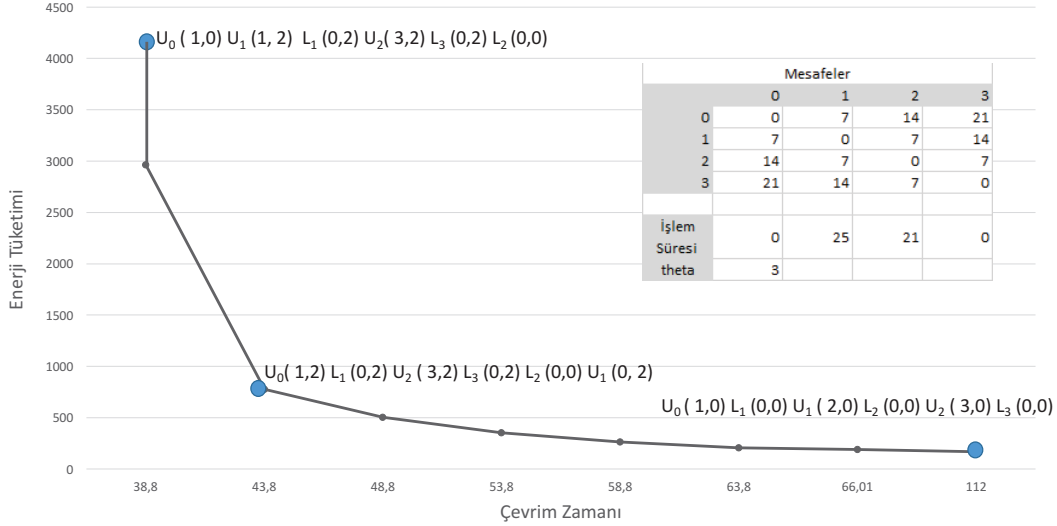
Şekil 11. 2 makine, ÖT, DV veri kümesi için örnek etkin çözüm kümesi

Tablo 18. 2 makine, ÖT, DV veri kümesi için çözüm süreleri

CT	MISOCP		MINLP	
	Sapma (%)	Çözüm Süresi (sn)	Sapma (%)	Çözüm Süresi (sn)
48.8	0.0	8.5	76	ZL
53.8	0.0	10.8	0.0	4399
58.8	0.0	10.5	0.0	5533
63.8	0.0	11.2	0.0	454
68.8	0.0	10.9	0.0	628
73.8	0.0	10.4	0.0	558
78.8	0.0	3.9	0.0	346
83.8	0.0	6.9	0.0	101
88.8	0.0	10.4	0.0	71
90.02	0.0	11.4	0.0	171
133.8	0.0	7.5	0.0	113
135.01	0.0	7.6	0.0	81
Toplam		109.9		23255

Şekil 12, 2 makineden oluşan, makineler arası uzaklığın özdeş toplamsal ve işlem sürelerinin geniş aralıktan seçildiği bir veri kümesi için elde edilen etkin çözümler kümesini göstermektedir. Grafik incelendiğinde, çevrim zamanı üst limiti değerlerine bağlı olarak 3 farklı döngünün ortaya çıktığı görülmektedir. θ değeri 3 olan bu veri kümesi için çözümler, toplamda 4 defa tutucu değişimi gerektiren 3 numaralı döngüyle başlamış ve \overline{CT} değeri arttıkça 6 numaralı

döngü ile devam etmiştir. Çevrim zamanı limiti yeterince büyük olduğunda ise döngü 7 ile sonlanmıştır. Problemin çözümü ile yine %30 enerji tasarrufu sağlanmıştır. Tablo 19, bu örnek için etkin çözümleri elde etmek için gerekli süreleri göstermektedir.



Şekil 12. 2 makine, ÖT, YV veri kümesi için örnek Pareto önyüz

Tablo 19. 2 makine, ÖT, DV veri kümesi için çözüm süreleri

CT	MISOCP		MINLP	
	Sapma (%)	Çözüm Süresi (sn)	Sapma (%)	Çözüm Süresi (sn)
38.8	0	10.1	0	3646
43.8	0	9.0	0	3183
48.8	0	5.9	0	2222
53.8	0	10.8	0	1571
58.8	0	10.9	0	274
63.8	0	5.0	0	385
66.01	0	8.4	0	162
112	0	6.6	0	85
Toplam		66.7		11529

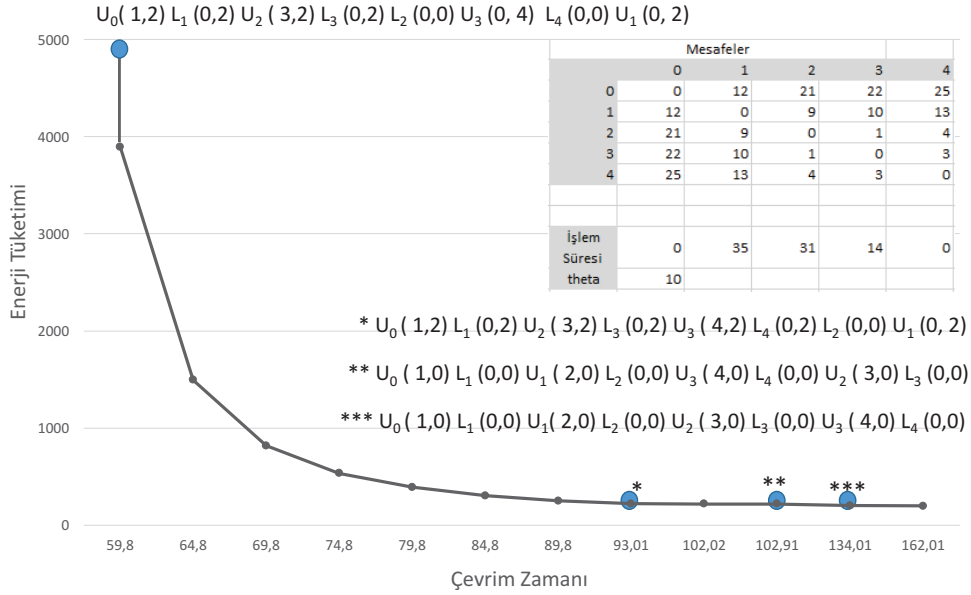
2 makinelik veri kümelerinin yanı sıra, 3 makineden oluşan örnekler için de etkin çözümlerin bulunması çalışması yapılmıştır. Elde edilen çözümlere göre, 3 makine için olası döngülerden 19 tanesinin çözümlerde ortaya çıktığı gözlemlenmiştir. Ayrıca minimum çevrim zamanını sağlayan çözümlerde 5 farklı 1-birim döngü gözlemlenmiştir. Bu döngülerden 20 veri kümesinin 13'ünde elde edilen döngü aşağıda verilmiştir.

$$U_0(1,0) \rightarrow L_1(0,0) \rightarrow U_3(4,0) \rightarrow L_4(0,0) \rightarrow U_2(3,0) \rightarrow L_3(0,0) \rightarrow U_1(2,0) \rightarrow L_2(0,0)$$

20 veri kümesi için alınan etkin çözümlerin % 25 inde elde edilen döngü de aşağıda verilmiştir.

$$U_0(1,2) \rightarrow L_1(0,2) \rightarrow U_2(3,2) \rightarrow L_2(3,0) \rightarrow U_3(3,4) \rightarrow L_4(3,0) \rightarrow L_3(0,0) \rightarrow U_1(0,2)$$

Şekil 13, 3 makineden oluşan, makineler arası uzaklığın genel toplamsal ve işlem süreleri varyansının yüksek olduğu bir veri kümesi için alınan etkin çözümler kümesini göstermektedir. Grafik incelendiğinde, 4 farklı döngü elde edildiği görülmektedir. etkin çözümler doğrusu boyunca, robot hareket sıralamasının değiştiği noktalar işaretlenmiştir. Problemin çözümü ile ulaşılan enerji tasarrufu bu veri kümesi için %24'tür. Etkin çözümlerin, çözüm süreleri Tablo 20'de verilmiştir. MINLP modelinde her problem için 1 saat zaman limiti kullanılmıştır. Tablodan da görülebileceği üzere bütün problemlerde MINLP modeli zaman limitine ulaşmıştır.



Şekil 13. 3 makine, GT, YV veri kümesi için örnek Pareto önyüz

Tablo 20. 3 makine, GT, YV veri kümesi için çözüm süreleri

CT	MISOCP		MINLP	
	Sapma (%)	Çözüm Süresi (sn)	Sapma (%)	Çözüm Süresi (sn)
59.8	0	24.0	MD	ZL
64.8	0	25.1	MD	ZL
69.8	0	25.0	MD	ZL
74.8	0	26.2	MD	ZL
79.8	0	22.8	MD	ZL
84.8	0	25.6	MD	ZL
89.8	0	22.2	MD	ZL
93.0	0	23.7	MD	ZL
102.0	0	21.9	MD	ZL
102.9	0	21.7	MD	ZL
134.0	0	20.7	MD	ZL
162.0	0	20.1	MD	ZL
Toplam		279,9		43200

2 ve 3 makinede olduğu gibi, 4 makine boyutundaki veri setleri için de etkin çözümler alınmış ve çözümler incelenmiştir. Olası en küçük çevrim zamanı ile minimum enerji tüketimi sağlanan etkin çözümlerde 6 farklı döngü açığa çıktığı görülmüştür. En çok elde edilen 1-birim döngü

aşağıda sunulmuştur. Bu döngü toplamda elde edilen çözümlerin %48'inde ortaya çıkmıştır.

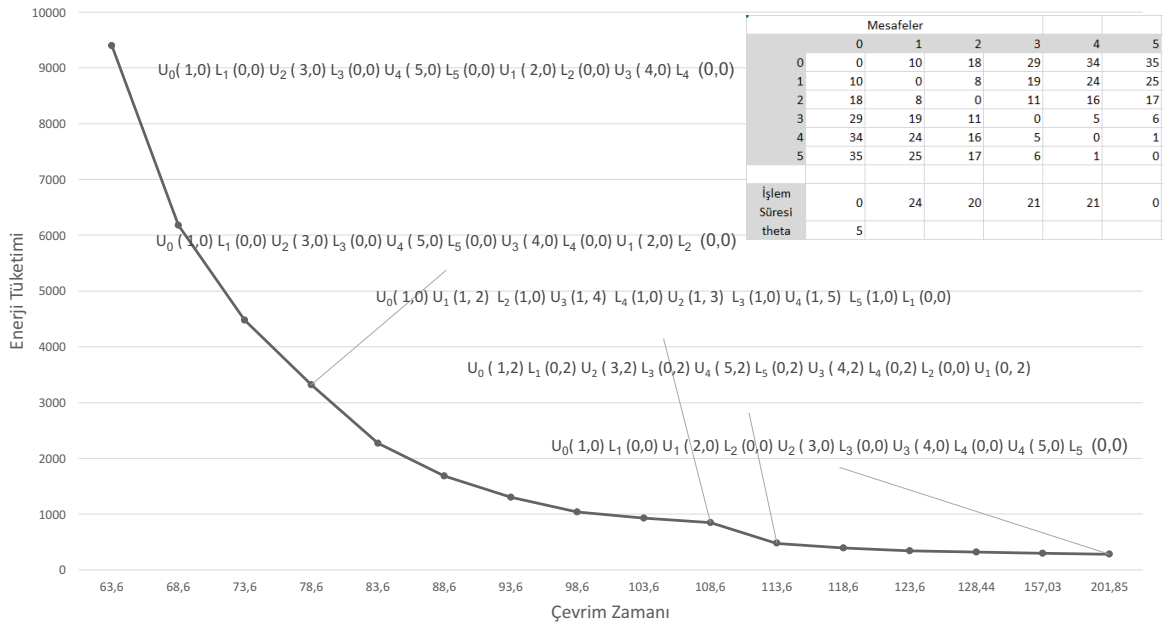
$$U_0(1, 0) \Rightarrow L_1(0, 0) \Rightarrow U_2(3, 0) \Rightarrow L_3(0, 0) \Rightarrow U_4(5, 0) \Rightarrow L_5(0, 0) \Rightarrow U_1(2, 0) \Rightarrow L_2(0, 0) \Rightarrow U_3(4, 0) \Rightarrow L_4(0, 0)$$

Çevrim zamanının arttırılmasıyla, etkin çözümlerde 28 tane daha farklı çevrim elde edilmiştir. Elde edilen 34 adet 1-birim döngüden, en çok elde edilen iki döngü aşağıda verilmiştir. Bu döngülerin, 440 etkin çözüm içinde ortaya çıkma oranları sırasıyla %25 ve %12'dir.

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

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

Şekil 14, 4 makineden oluşan, makineler arası uzaklık genel toplamsal ve işlem süreleri dar aralıktan elde edilen veri kümesi için alınan etkin çözümler kümesini göstermektedir. Grafik incelendiğinde, 5 farklı döngü elde edildiği görülmektedir. θ değerinin oldukça büyük olması nedeniyle, etkin çözümlerin bir çoğunda tutucu değişimi gerektirmeyen çözümler elde edilmiştir. Ayrıca, ilk etkin çözüm olarak, veri kümelerinin %48'inde de elde edilen yukarıda verilen döngü ortaya çıkmıştır. Etkin çözümlerin, çözüm süreleri Tablo 20'de verilmiştir.



Şekil 14. 3 makine, GT, YV veri kümesi için örnek Pareto önyüz

Tablo 21. 3 makine, GT, YV veri kümesi için çözüm süreleri

CT	MISOCP		MINLP	
	Sapma (%)	Çözüm Süresi (sn)	Sapma (%)	Çözüm Süresi (sn)
63,6	0	142,266	MD	ZL
68,6	0	249,5	MD	ZL
73,6	0	223,563	MD	ZL
78,6	0	216,594	MD	ZL
83,6	0	290,188	MD	ZL
88,6	0	263,5	MD	ZL
93,6	0	299,875	MD	ZL
98,6	0	230,437	MD	ZL
103,6	0	319,782	MD	ZL
108,6	0	126,015	MD	ZL
113,6	0	297,344	MD	ZL
118,6	0	262,407	MD	ZL
123,6	0	277,781	MD	ZL
128,44	0	201,141	MD	ZL
157,03		182,828	MD	ZL
201,85		123,953	MD	ZL
Toplam		3707.2		172800

Belirtilen makine sayılarının yanında, genel toplamsal uzaklık ve işlem süresi geniş aralıktan elde edilen 5 ve 6 makinelik beşer problem verisi oluşturulmuştur. Minimum enerji tüketimini sağlayan minimum çevrim zamanı probleminin çözümleri MISOCP ile 5 makineli veri kümelerinde ortalama 1978 saniyede optimal olarak bulunmuştur. 6 makine için ise 2 saat zaman limit altında optimal çözümlere ulaşmak mümkün olmamıştır. Sonuç olarak, MISOCP çözüm süresini oldukça kısaltmış olsa da, veri kümesinin boyutu arttıkça modelin yetersiz kaldığı görülmektedir. Büyük boyuttaki veri setleri için de makul sürelerde çözümler alabilmek için sezgisel bir algoritma geliştirilmiştir. Bir sonraki bölümde bu algoritma detaylı olarak anlatılacaktır. Ayrıca, sezgisel algoritma ve MISOCP için yapılan performans testlerine de ilerleyen bölümlerde yer verilmiştir.

4.4 Etkin Çözüm Kümesi 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 algoritmasının amacı kısa sürede çok sayıda yaklaşık etkin çözümler bulmaktır. Yaklaşık etkin çözüm ifadesi elde edilecek çözümün sezgisel yolla bulunmuş olması nedeniyle kullanılmıştır.

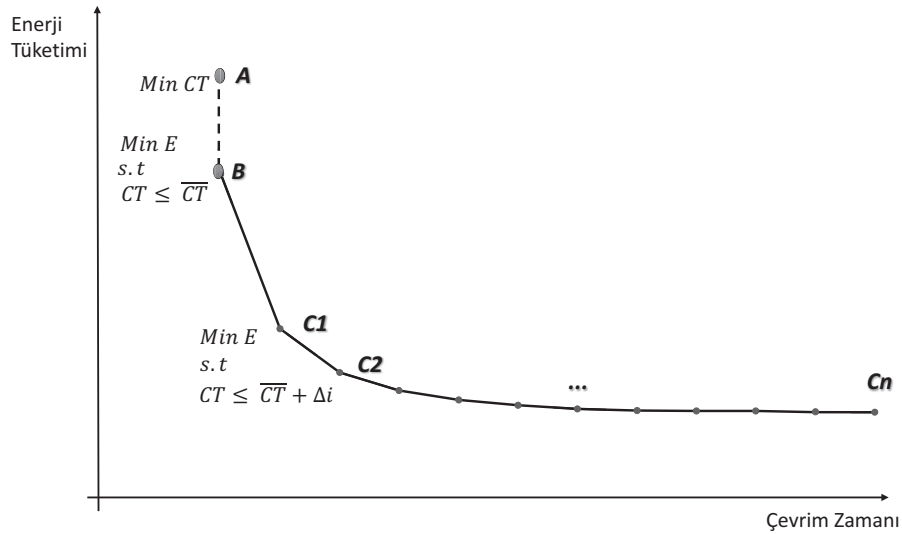
ETA'nın genel çalışma mantığı şu şekildedir. ETA ilk olarak mümkün olan en kısa çevrim süresini bulmaya çalışır. Bunun için öncelikle robotun tüm hareketlerinde hızı en yüksek hız eşitlenir (Adım 0). Böylece robotun makineler arası hareket süreleri minimum değerlerine eşitlenir. Elde edilen minimum hareket süreleri dikkate alınarak minimum çevrim süresini veren aktivite sıralamasını bulmak bir sonraki aşamadır (Aşama 1). Bu aşamada Tavlama Benzetimi metasezgiseli kullanılarak en iyi aktivite sırası bulunmaya çalışılır.

En yüksek hız değerleri ile bulunan aktivite sıralaması bir çevrim çizelgesi verir. Bu

çizelgenin belli zamanlarında robotun ve makinelerin boşa beklediği durumlar oluşabilir. Boşa bekleme sürelerinin bir kısmı robot yavaşlatılarak azaltılabilir, böylece çevrim süresi değişmeden enerji tüketimi azaltılabilir. Aşama 2'de hangi aktivitelerde çevrim süresi değişmeden robotun yavaşlatılabileceği belirlenir. Aşama 3'te seçilen aktiviteler için enerji tüketimini minimize eden hızlar belirlenerek ilk yaklaşık etkin çözüm bulunmuş olur.

En kısa çevrim süresini bulmaya çalıştığımız ilk yaklaşık etkin çözüm bulduktan sonra ETA, çevrim süresini her iterasyonda arttırarak yeni yaklaşık etkin çözümler bulur. Bu iteratif adımlar Aşama 4'ü oluşturmaktadır.

ETA'nın elde edeceği yaklaşık etkin çözüm kümesi örnek olarak Şekil 15'deki grafikteki gibi olacaktır. Aşama 1'de A noktası, Aşama 3'te B noktası Aşama 4'te ise C1, C2,...,Cn noktaları türetilecektir.



Şekil 15. Etkin Çözüm Kümesi

Sezgisel algortimanın genel işleyişi ise 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 ve aktivite sıralamalarını belirle

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

Aşama 3: Esnek olarak belirlenen hızları, tüm bekleme süreleri olabildiğince azaltacak kadar yavaşlat, böylece elde edilen çevrim zamanı alt limitini sağlayan en küçük enerji tüketimini hesapla

Aşama 4: Çevrim zamanı alt limitini λ kadar artır.

Aşama 4.1: Aşama 2'de belirlenen kritik hızları, yol aldıkları zaman λ kadar artacak olacak miktarda yavaşlat, enerji tüketimini hesapla

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

→ Değilse, esnek hızların a yol aldıkları zamanı ϵ kadar artacak şekilde bu hızları yavaşlat, λ değerini ϵ kadar azalt. Adım 4.1'e geri dön

Aşama 4.2: Aşama 1'deki arama algoritmasını kullanarak, çevrim zamanı alt limitinden küçük olan döngüleri bul

Elde edilen döngülerin çevrim zamanları, alt limite eşit olduğunda gerçekleşen enerji tüketimini, Aşama 4.1'de tanımlanan yöntemi kullanarak hesapla

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

Yukarıda genel aşamaları verilen sezgisel algoritmanın, bu bölümde detaylı anlatımı yapılacaktır.

4.4.1 ETA: Aşama 1

Aşama 1'de minimum çevrim süresine sahip çizelgeyi bulmak hedeflenmektedir. Bunun için de Aşama 0'da belirtildiği gibi robotun tüm hareketleri maksimum hızda yapacaktır. Minimum çevrim süresine sahip çevrimin bulunması için bir başlangıç çözümünden başlayarak çözüm uzayı taranması ve iyi bir çözüm bulunması hedeflenmiştir.

Başlangıç çözümü olarak algoritmaya aktivite sıralaması verilmektedir. Ayrıca tavlama benzetimi algoritması da eldeki çözümden alternatif aktivite sıralamaları türetmektedir. Elde edilen aktivite sıralamalarının olurlu olup olmadıklarını tesbit edebilmek, aktivite sıralamalarının verdiği çevrim sürelerini hesaplayabilmek için çeşitli alt algoritmalar gerekmektedir. Aşağıda öncelikle bu alt algoritmalar verilecektir.

İlk olarak her aktivite sırasında robotun hangi tutucusunun aktif olacağı belirlenmelidir. Bunun için aşağıdaki algoritma kullanılmıştır:

Aktif_Tutucuyu_Bul

Her i değeri için, $i \in 2..Pozisyon\ sayısı$

Eğer i . pozisyonda m makinesine yükleme, $(i-1)$. pozisyonda ise $m-1$ makinesinden boşaltma aktivitesi gerçekleşiyor ise,
İki pozisyon arasında tutucu değişimi olmamıştır; i . pozisyondaki aktif tutucu $(i-1)$ 'deki aktif tutucuya eşittir.

Değilse,Eğer i . pozisyonda yükleme, $(i-1)$. pozisyonda ise boşaltma aktivitesi gerçekleşiyor ve i . pozisyondaki makine $(i-1)$ 'den hemen sonraki makine değilse
İki pozisyon arasında tutucu değişimi olmuştur; i . pozisyondaki aktif tutucu $(i-1)$ 'deki aktif olmayan tutucudur.

Değilse,Eğer i . pozisyonda yükleme, $(i-1)$. pozisyonda da yükleme aktivitesi gerçekleşiyor ise
İki pozisyon arasında tutucu değişimi olmuştur; i . pozisyondaki aktif tutucu $(i-1)$ 'deki aktif olmayan tutucudur.

Değilse,Eğer i. pozisyonda boşaltma, (i-1). pozisyonda da boşaltma, aktivitesi gerçekleşiyor ise

İki pozisyon arasında tutucu değişimi olmuştur; i. pozisyondaki aktif tutucu (i-1)'deki aktif olmayan tutucudur.

Değilse,Eğer i. pozisyonda boşaltma, (i-1). pozisyonda ise aynı makineye yükleme aktivitesi gerçekleşiyor ise,

İki pozisyon arasında tutucu değişimi olmamıştır; i. pozisyondaki aktif tutucu (i-1)'deki aktif tutucuya eşittir.

Değilse,Eğer i. pozisyonda boşaltma, (i-1). pozisyonda ise yükleme aktivitesi gerçekleşiyor ve bu iki aktivite farklı makinelerde gerçekleşiyor ise,

Eğer (i+1). pozisyonda da boşaltma aktivitesi var ise,

İki pozisyon arasında tutucu değişimi olmuştur; i. pozisyondaki aktif tutucu (i-1)'deki aktif olmayan tutucudur.

Değilse İki pozisyon arasında tutucu değişimi olmamıştır; i. pozisyondaki aktif tutucu (i-1)'deki aktif tutucuya eşittir.

Aktif tutucular belirlendikten sonra, aşağıda verilen algoritma ile pozisyonlardaki tutucu indislerine karar verilir.

Tutucu İndislerini Bul

Her i değeri için, $i \in 1..Pozisyon\ sayısı$

Eğer i. pozisyonda m makinesine x tutucusu ile boşaltma aktivitesi gerçekleşiyorsa,

→ (m+1) makinesine j. pozisyonda yükleme aktivitesi gerçekleşiyor olsun

Her k değeri için, $k \in i..(j-1)$

→ k. pozisyonda, x tutucusunun indisini **(m+1)** olarak belirle

Değilse,Eğer i. pozisyonda m makinesine x tutucusu ile yükleme aktivitesi gerçekleşiyorsa,

→ i. pozisyondan sonra, x tutucusu ile boşaltma aktivitesi j. pozisyonda gerçekleşiyor olsun

Her k değeri için, $k \in i..(j-1)$

→ k. pozisyonda, x tutucusunun indisini **0** olarak belirle

Her aktivite için aktif tutucu ve tutucu indisleri belirlendikten sonra çevrimin olurluluğunu kontrol etmek için aşağıdaki algoritma kullanılır.

olurluluğu kontrol et

Her i değeri için, $i \in 2..Pozisyon\ sayısı$

Eğer i. pozisyonda m makinesine x tutucusu ile yükleme aktivitesi gerçekleşiyorsa,

Eğer (i-1). pozisyonda x tutucusunun indisi m değerine eşit değilse, aktivite sıralaması olursuzdur. Döngüden çıkılır.

Değilse,Eğer (i-1). pozisyonda m makinesinden x tutucusu ile boşaltma aktivitesi gerçekleşiyorsa,

Eğer (i-1). pozisyonda x tutucusunun indisi (m+1) değerine eşit değilse,
ya da

Eğer i. pozisyonda x tutucusunun indisi (m+1) değerine eşit değilse,
aktivite sıralaması olursuzdur. Döngüden çıkılır.

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. Böylece pozisyonların başlama zamanları ve nihayetinde de aktivite sıralaması ile elde edilen çevrim zamanı belirlenmektedir.

Makinelerarası_Geçen_Zamanı_Bul

Her i değeri için, $i \in 1..Pozisyon\ sayısı$

Robot i. pozisyonda m, (i+1). pozisyonda n makinesinde olsun

Eğer i. pozisyondaki aktif tutucu (i+1). pozisyondaki aktif tutucuya eşitse

→ i. pozisyondaki mesafe $dist_i$ değerini m ve makineleri arası uzaklığa

→ i. pozisyondaki geçen zaman δ_i değerini $dist_i V_i$ değerine eşitle

Değilse → i. pozisyondaki mesafe $dist_i$ değerini m ve makineleri arası uzaklığa

→ i. pozisyondaki geçen zaman δ_i değerini $\max\{\theta, dist_i V_i\}$ değerine eşitle

Makinelerarası geçen zaman bulduktan sonra eldeki çizelgenin çevrim zamanını bulmak için aşağıda verilen algoritma kullanılır:

Çevrim_Zamanını_Bul

CT:=0, CTkontrol:=1 olsun

CT değeri CTkontrol değerine eşitlenene kadar aşağıdaki döngüyü tekrarla {

Her i değeri için, $i \in 1..Pozisyon\ sayısı$ {

CT değerini CTkontrol değerine ata

Robot i. pozisyonda m makinesinde olsun

Eğer i. pozisyonda ile yükleme aktivitesi gerçekleşiyorsa

→ i. pozisyonun başlama zamanı T_i değerini, $T_{(i-1)} + \delta_{(p-1)} + \epsilon$ toplamına eşitle

Değilse ilgili makineye yükleme işlemi j. pozisyonda yapılıyor olsun,

Eğer $j < i$ ise

→ i. pozisyonun başlama zamanı T_i değerini, $\max\{T_{(i-1)} + \delta_{(i-1)} + \epsilon ; T_j + \epsilon + P_m\}$ toplamına eşitle

Değilse

→ i. pozisyonun başlama zamanı T_i değerini, $\max\{T_{(i-1)} + \delta_{(i-1)} + \epsilon ; -CT + T_j + \epsilon + P_m\}$ toplamına eşitle

Eğer i . pozisyon son pozisyonsa, CT değerini, $T_i + \delta_{(i-1)} + \epsilon$ toplamına eşitle

}
}

Aktivite sıralaması belirlenen bir çevrimin enerji tüketimi ise şöyle hesaplanmaktadır:

Enerji_Tüketimini_Bul

EnerjiTüketimi:=0

Her i değeri için, $i \in 1..Pozisyon\ sayısı$ {

→ EnerjiTüketimi:=EnerjiTüketimi+ $dist_i \cdot V_i^a$

Eğer i ve $(i+1)$ pozisyonları arasında tutucu değişimi gerçekleştiyse

→ EnerjiTüketimi:=EnerjiTüketimi+ G

}

Aşama 1'de bahsedilen tavlama benzetimi algoritmasıyla yeni iyi döngüler, başlangıç çözümü üzerinden iki aktivitenin rasgele yerlerinin değiştirilmesiyle elde edilir. Başlangıç çözümü olarak Sethi vd. [93] tarafından belirli koşullar altında optimal olduğu kanıtlanan aktivite sıralaması ele alınmıştır. m makine olan bir sistemde, başlangıç aktivite sıralaması şöyledir;

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

Aktif_Tutucuyu_Bul ve **Tutucu_İndislerini_Bul** algoritmaları kullanılarak pozisyonlardaki tutucu indisleri aşağıdaki gibi bulunmaktadır.

$$U_0(1, 0) \Rightarrow U_1(1, 2) \Rightarrow L_1(0, 2) \Rightarrow U_2(3, 2) \Rightarrow L_2(3, 0) \Rightarrow U_3(3, 4) \cdots \Rightarrow L_{(m-1)}(0, m) \Rightarrow U_m(m+1, m) \Rightarrow L_m(m+1, 0) \Rightarrow L_{(m+1)}(0, 0)$$

Bu durumda, başlangıç aktivite sıralaması için **Makinelerarası_Geçen_Zamanı_Bul** ve **Çevrim_Zamanını_Bul** methodları kullanılarak elde edilen çevrim zamanı **mevcutCT** olsun. Aşağıda verilen tarama algoritması başlangıç çözümünden başlayarak çevrim süresi en az olan çözümü bulmaya çalışır.

Tavlama Tarama_Algoritması

Adım 0: Başlangıç sıcaklığı:=100, soğuma oranı:=0.99, durma sıcaklığı:= 0 ve $\Delta CT:=0$ olsun.

Sıcaklık , **durma sıcaklığından büyük olduğu müddetçe yap** {
sayaç:= 0

Sayaç , **10 değerinden küçük olduğu müddetçe yap** {

Adım 1: sayaç değerini bir artır

Adım 2: Rasgele iki aktivitenin yerini değiştirerek olurlu aday döngüyü belirle

Adım 3: Aday döngü için makinelerarası_geçen_zamanı_bul ve çevrim_zamanını_bul methodları kullanılarak elde edilen çevrim zamanı **adayCT** olsun

Adım 4: ΔCT , mevcutCT - adayCT farkına eşit olsun

Eğer ΔCT değeri sıfırdan büyükse, yani gelişme sağlanmışsa

⇒ Mevcut çözümü aday çözüme, mevcutCT değerini de adayCT değerine eşitle

Eğer mevcutCT elde edilen en iyi çevrim zamanı ise

⇒ Çözüm kümesini temizle, mevcut çözümü kümeye ekle

Değilse, Eğer mevcutCT daha önce elde edilmiş çözümlerin çevrim zamanına eşitse, ve bu çözüm daha önce kümeye eklenmemişse,

⇒ Mevcut çözümü kümeye ekle

Değilse , yani gelişme sağlanmamışsa

⇒ Mevcut çözümün kabul edilme olasılığı p_{kabal} değerini hesapla:

$$p_{kabal} := 1 / (1 + e^{\Delta CT / sBcaklBk})$$

⇒ (0,1) aralığında olan h rasgele sayısını seç

Eğer p_{kabal} değeri h rasgele sayısından büyük ise

⇒ Mevcut çözümü aday çözüme, mevcutCT değerini de adayCT değerine eşitle

Değilse Çözümü kabul etme, aramaya devam et

}

Adım 5: sıcaklığı, soğuma oranı kadar soğut

}

Adım 6: Çözüm kümesinde bulunan, aynı çevrim zamanına sahip farklı enerji tüketimi olan döngülerden, en küçük enerji tüketimi olan çözümü minimum çevrim zamanına sahip çözüm olarak belirle

4.4.2 ETA: Aşama 2

Aşama 1'de en küçük çevrim zamanını veren çözüm, diğer bir deyişle grafik 15'teki A noktası bulunmuştur. Aşama 2'de ise, A noktasındaki çözüm ile aynı çevrim zamanına sahip ancak esnek olan robot hareket hızlarının olabildiğince azaltılmasıyla daha az enerji tüketimine sahip B noktası çözümü bulunacaktır. Robotun hangi hareketlerindeki hızların esnek yani hareketin yavaşlatılabilir, hangilerinin ise kritik yani yavaşlatılmasıyla çevrim zamanını artıran olduğunu tespit etmek için sistemdeki bekleme miktarları göz önüne alınacaktır.

Sistemde iki farklı bekleme türü gerçekleşebilir. Bunlardan ilki robotların boşaltma yapacağı makinenin önünde, makine işlem süresinin bitmesini beklemesidir. Bu bekleme tipi robot beklemesi olarak adlandırılabilir. İkinci tür olan makine beklemesi ise, işlem süresini tamamlayan makineni boşaltma işlemi için robotu beklemesidir.

Kritik hızların tespit edilmesini sağlayan sezgisel algoritma ise şöyledir:

Kritik_Hızları_Bul

Her m değeri için, $m \in 1..Makine\ sayısı$

⇒ m makisine yükleme aktivitesi i . pozisyonda, boşaltma aktivitesi ise j . pozisyonda gerçekleşiyor olsun

Eğer Boşaltma öncesi robot beklemesi gerçekleşiyor ve $j < i$ ise

Her k değeri için, $k \in j+1..i$

Eğer k'nın tüm değerleri için, ilgili pozisyonda robot beklemesi gerçekleşmiyorsa m makinesi kritik yol üzerindedir ve j ile i arasındaki hızlar da kritiktir.

Sonraki k

Değilse,Eğer Boşaltma öncesi robot beklemesi gerçekleşiyor ve $i < j$ ise

Her k değeri için, $k \in \{i + 1..Pozisyon Sayısı\} \cup \{1..j\}$

Eğer k'nın tüm değerleri için, ilgili pozisyonda robot beklemesi gerçekleşmiyorsa m makinesi kritik yol üzerindedir ve j ile i arasındaki hızlar da kritiktir.

Sonraki k

Sonraki m

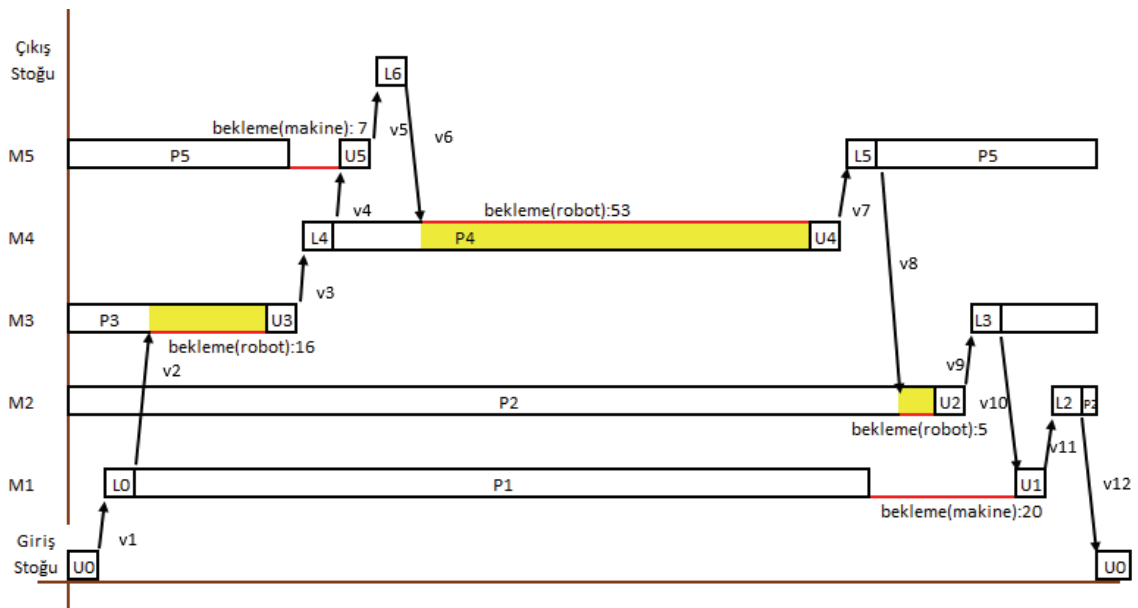
Şekil 16'de 5 makineli bir sistem için aktivitelerin başlama ve bitiş zamanlarını, makine işlem sürelerini göz önünde bulundurarak oluşturulmuş Gantt şemasını göstermektedir. Ele alınan sistemde makineler arası uzaklık özdeş toplamsal özellik göstermektedir ve ardışık uzaklık 10 birimdir. Diğer ortam özellikleri ise şu şekildedir:

- $\epsilon = 4, \theta = 0, VUB = 10, VLB = 0$
- $P_1 = 100, P_2 = 120, P_3 = 40, P_4 = 65, P_5 = 60,$

Gerçekleşen aktivite sıralaması ise,

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

şeklinde gösterilebilir.



Şekil 16. 5 makine için Örnek Gantt Şeması

Gantt şeması incelendiğinde, 2, 3 ve 4. makineler önünde robot beklemesi, 1 ve 5. makinelerde ise makine beklemesi gerçekleştiği görülmektedir. Bu durumda, 2, 3 ve 4. makineler kritik yol üzerinde olmaya aday makinelerdir. Örneğin 2. makine için, **Kritik_Hızları_Bul** algoritmasının adımları takip edilecek olursa bu makinenin kritik yol üzerinde olup olmadığı belirlenebilir. 2. makine için boşaltma aktivitesi U_2 , yükleme aktivitesi L_2 ise 12. pozisyonda gerçekleşmektedir. 10, 11 ve 12. pozisyonların hiç birinde ise robot beklemesi gerçekleşmemektedir. Öyleyse 2. makine kritik yol üzerindedir ve U_2 ve L_2 arasındaki hızlar olan v_9, v_{10}, v_{11} hızları kritik hızlardır. Aynı şema üzerinden 4. makine incelenecek olursa bu makinenin kritik yol üzerinden olmadığı görülecektir. Zira U_4 ve L_4 arasında bulunan 9 ve 3. pozisyonlarda robot beklemesi gerçekleşmektedir. Bu sebeple U_4 ve L_4 arasındaki hızların kritik olup olmadığıyla ilgili karara varmak mümkün değildir.

Verilen aktivite sıralaması için **Kritik_Hızları_Bul** algoritması kullanıldığında, v_9, v_{10}, v_{11} hızlarının kritik, diğer hızların ise esnek olduğu sonucuna ulaşılmaktadır.

4.4.3 ETA: Aşama 3

Hangi hızların esnek olduğuna karar verildikten sonra, bu hızların çevrim zamanını değiştirmeden ne kadar yavaşlatılabileceği bulunmalıdır. Diğer bir deyişle, her bir esnek hız değerinin çevrim zamanını artırmayan gerçek alt sınır değerleri hesaplanmalıdır. Örneğin, sadece v_3, v_7 ve v_8 hızları yavaşlayacak olsun. Bu hızlarla alınan toplam zamanın $\delta_3 = 1, \delta_7 = 1$ ve $\delta_8 = 3$ olmak üzere toplamda 5 birim olduğu görülmektedir. Bu hızlardan hemen sonra gerçekleşen robot beklemesi de 5 birim olduğuna göre, bu iki hız, yolda geçirdikleri zamanı 9 birim olacak şekilde yavaşlatılabilir. v_7 hızının yavaşlatılması L_5 aktivitesinin başlamasını, v_3 hızının yavaşlatılması da L_4 aktivitesinin başlamasını erteleyecektir ancak, 4 ve 5. makineler kritik yol üzerinde bulunmadığından bu yavaşlama çevrim zamanını etkilememektedir. Sözlü şekilde ifade edilen bu kısıt matematiksel olarak şöyle yazılabilir:

$$\frac{dist_3}{v_3} + \frac{dist_7}{v_7} + \frac{dist_8}{v_8} \leq \delta_3 + \delta_7 + \delta_8 + 5$$

Esnek hızlar olan $v_{12}, v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8$ 'in tümünü göz önünde bulundurularak yazılan kısıt ise şu şekilde ifade edilebilir:

$$10 \max \left\{ 0, \frac{dist_{12}}{v_{12}} + \frac{dist_1}{v_1} + \frac{dist_2}{v_2} - 16 \right\} + \frac{dist_3}{v_3} + \max \left\{ 0, \frac{dist_4}{v_4} + \frac{dist_5}{v_5} + \frac{dist_6}{v_6} - 53 \right\} + \frac{dist_7}{v_7} + \frac{dist_8}{v_8} \leq \delta_{12} + \delta_1 \cdots + \delta_8 + 5$$

Belirtilen eşitsizlikte $\max \{0, \dots\}$ gibi ifadelerin bulunmasının sebebi, hızların sebep oldukları diğer robot bekleme durumlarıdır. Örneğin, v_4, v_5, v_6 değerlerinin hemen sonrasında 53 birimlik bir bekleme gerçekleşmektedir. Dolayısıyla, dokuzuncu pozisyonda gerçekleşen 5 birimlik beklemenin ortadan kalkmasında v_4, v_5, v_6 hızlarının da yavaşlatılmasının da rolü olacaksa, bu yavaşlamanın öncelikle yedinci pozisyonda gerçekleşen beklemeyi ortadan kaldırması gerekmektedir. Eğer bu 3 hızın yavaşlamasıyla geçen zaman en fazla 53 birim ise ilgili kısıt

şu şekilde yazılabilir;

$$\max \left\{ 0, \frac{dist_{12}}{v_{12}} + \frac{dist_1}{v_1} + \frac{dist_2}{v_2} - 16 \right\} + \frac{dist_3}{v_3} + \frac{dist_7}{v_7} + \frac{dist_8}{v_8} \leq \delta_{12} + \delta_1 \cdots + \delta_8 + 5$$

Tüm bekleme durumları göz önünde bulundurulduğunda, elde edilecek kısıtlar şöyledir;

$$\frac{dist_{12}}{v_{12}} + \frac{dist_1}{v_1} + \frac{dist_2}{v_2} + \frac{dist_3}{v_3} + \frac{dist_4}{v_4} + \frac{dist_5}{v_5} + \frac{dist_6}{v_6} + \frac{dist_7}{v_7} + \frac{dist_8}{v_8} \leq \delta_{12} + \delta_1 \cdots + \delta_8 + 5 + 53 + 1 \quad (111)$$

$$\frac{dist_{12}}{v_{12}} + \frac{dist_1}{v_1} + \frac{dist_2}{v_2} + \frac{dist_3}{v_3} + \frac{dist_7}{v_7} + \frac{dist_8}{v_8} \leq \delta_{12} + \delta_1 + \delta_2 + \delta_3 + \delta_7 + \delta_8 + 5 + 1 \quad (112)$$

$$\frac{dist_3}{v_3} + \frac{dist_7}{v_7} + \frac{dist_8}{v_8} \leq \delta_3 + \delta_7 + \delta_8 + 1 \quad (113)$$

Bir robotik hücrede her bir aktivite sıralaması, makine işlem süreleri, hız üst limitleri gibi ortam değerlerine bağlı olarak farklı kısıtlar ortaya çıkabilir. Bu kısıtların sistematik olarak elde edilebilmesi için geliştirilen algoritma şu şekildedir:

Kısıtları_Bul

F_n , n. kısıtta bulunan esnek hızlar kümesi, $tBekleme_n$ n. kısıtta bulunacak bekleme süresi, max_n n. kısıtta bulunan $\max \{0, \dots\}$ ifadelerinin sayısı olsun, $n:=1$.

Kritik bir hızdan sonraki ilk esnek hız i., önceki son esnek hız da j. pozisyonlarda gerçekleşiyor olsun.

Adım 1: (j+1). pozisyonda gerçekleşen robot beklemesi ile ilgili kısıtı bul:

Her k değeri için, $k \in j+1..i$ (Eğer $i < j$ ise $k \in j + 1..Pozisyon Sayısı\} \cup \{1..i\}$)

Eğer k. pozisyonda robot beklemesi gerçekleşiyor ise

$\Rightarrow tBekleme_n$ değerini k. pozisyondaki bekleme miktarı kadar artır

$\Rightarrow max_n$ değerini 1 artır

$\Rightarrow F_n := F_n \cup \{v_k\}$

Adım 2: Kritik yolda gerçekleşebilecek makine beklemeleriyle ilgili kısıtları bul

Her k değeri için, $k \in j+1..i$ (Eğer $i < j$ ise $k \in j + 1..Pozisyon Sayısı\} \cup \{1..i\}$)

Eğer k. pozisyonda makineye yükleme aktivitesi gerçekleşiyor ve aynı makineden boşaltma işlemi kritik yol üzerindeki pozisyonlarda gerçekleşiyor ise

$\Rightarrow n := n+1$, $tBekleme_n := tBekleme_n + k$. pozisyonda gerçekleşen makine beklemesi

Her l değeri için, $l \in j+1..k$ (Eğer $i < j$ ise $l \in j + 1..Pozisyon Sayısı\} \cup \{1..k\}$)

Eğer l. pozisyonda robot beklemesi gerçekleşiyor ise

$$\begin{aligned} &\Rightarrow tBekleme_n \text{ değerini } l. \text{ pozisyon} \text{daki bekleme miktarı kadar artır} \\ &\Rightarrow max_n \text{ değerini } 1 \text{ artır} \\ &\Rightarrow F_n := F_n \cup \{v_l\} \end{aligned}$$

Geliştirilen kısıt bulma algoritmasının adımları ile, Gantt Şeması ile verilen örnek için, dokuzuncu pozisyonda bulunan 5 birimlik robot beklemesinin, ve on birinci pozisyonda gerçekleşen 20 birimlik makine beklemesinin aşılmaması ile ilgili kısıtlar oluşturulmuştur. Adım 1 ile Kısıt(111)'yi temsil eden 1. kısıt için hız kümesi ve bekleme değeri şu şekilde elde edilmiştir:

$$F_1 = v_{12}, v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8; \quad tBekleme_1 = 5 + 53 + 16 = 74 \quad (114)$$

Adım 2 ile de 2. kısıt için hız kümesi ve bekleme değeri

$$F_1 = v_{12}, v_1; \quad tBekleme_2 = 20 \quad (115)$$

olarak belirlenmiştir. Kısıt (112) ve (113)'un da kısıt (111)'den yola çıkarak elde edilmesi için aşağıda anlatılan kısıt sadeleştirme algoritması kullanılmıştır.

Kısıtı_Sadeleştir

Girdi olarak verilen kısıt n için

i F_n kümesine son eklenen hız, j de ilk eklenen hız olsun,

Her $k=i, k \geq j$ değeri için

Eğer k. pozisyonda m makinesinde robot beklemesi gerçekleşiyor ise

Her $l=k, v_l \in F_n$

$$\Rightarrow F_n := F_n \setminus \{v_l\}$$

Eğer l. pozisyonda L_m aktivitesi gerçekleşiyorsa ya da $l=j$ ise DUR

$$\Rightarrow tBekleme_n \text{ değerini } k. \text{ pozisyonda gerçekleşen bekleme kadar azalt}$$

Kısıt (112) ve (113)'u elde edebilmek için oluşturulan **kısıtı _ sadeleştir** yöntemi, **Kısıtları_Bul** algoritmasına Adım 3 ile entegre edilebilir:

Adım 3: Adım 1 ve 2'de elde edilen her bir kısıt n için

$$\Rightarrow cKısıt, n. \text{ kısıtın kopyası olsun}$$

Adım 3.1: max_n kere,

$$\Rightarrow cKısıt'ı \text{ **Kısıtı_Sadeleştir** yöntemiyle sadeleştir, kısıt kümesine ekle}$$

Adım 3 uygulandığında, Adım 3.1 max_1 değeri 2 olduğundan, 1. kısıt için 2 kere tekrarlanacaktır. Bu durumda ilk tekrardan, kısıt(112) için, ikinci tekrardan ise kısıt(113) için gerekli hız ve bekleme değerlerini içeren F_n küme ve $tBekleme_n$ olacaktır.

$$F_3 = v_{12}, v_1, v_2, v_3, v_7, v_8; \quad tBekleme_3 = 5 + 16 = 21 \quad (116)$$

$$F_4 = v_3, v_7, v_8; \quad tBekleme_4 = 5 \quad (117)$$

Adım 2'den elde edilen kısıt (115)'de max_2 değeri 0 olduğundan, Adım 3'ün uygulanmasına gerek kalmamıştır.

Tanımlanan kısıtlarda bulunan $dist_i$ ve δ_i ifadeleri hız_alt_sınırlarını_bul algoritması ile hesaba katılmış ve elde edilen her bir kısıtın sağlanabilmesi için, kısıt kümesinde bulunan hızların en fazla kaç birim yavaşlatılabileceği bulunmuştur.

Hız_Alt_Sınırlarını_Bul

Tüm kısıtlar için hız alt sınırları sabitlenene kadar aşağıdaki adımları tekrarla

Adım 0: maxVLB:=0

Kısıt kümesinde bulunan her kısıt n için

Adım 1: $tDist = 0, tDelta = 0, tBek = tBekleme_n, tHiz_n = 0$

İlgili hız değeri F_n kümesinde bulunan her pozisyon i için

Eğer v_i hız değeri daha önce hesaplanmadıysa

$$\Rightarrow tDist := tDist + dist_i$$

$$\Rightarrow tDelta := tDelta + \frac{dist_i}{v_i}$$

Değilse

$$\Rightarrow tBek := tBek + tDelta - \frac{dist_i}{v_i}$$

Sonraki i

Adım 2: Kısıt n için hız alt sınır değerini hesapla

$$\Rightarrow tHiz_n := tDist \setminus (tBek + tDelta)$$

$$\Rightarrow \maxVLB := \max\{ \maxVLB, tHiz_n \}$$

Sonraki n

Adım 3: En büyük hız alt sınır değerine sahip kısıttaki v_i hız değerlerini sabitle

$$\Rightarrow v_i := \max\{ \maxVLB, v_i \}$$

Şekil 16'de verilen örnek için **Kısıtları_Bul** algoritmasıyla bulunan kısıtların, **Hız_Alt_Sınırlarını_Bul** algoritmasının ilk tekrarında elde edilen değerleri şu şekildedir;

Kısıt (114) için: $tBek = 74, tDelta = 14, tDist = 140, tHiz_1 = 1.59$

Kısıt (115) için: $tBek = 20, tDelta = 3, tDist = 30, tHiz_1 = 1.30$

Kısıt (116) için: $tBek = 21, tDelta = 10, tDist = 100, tHiz_1 = 3.23$

Kısıt (117) için: $tBek = 5, tDelta = 5, tDist = 10, tHiz_1 = 5.0$

En büyük hız alt sınır değeri kısıt (117)'de gerçekleştiği için, bu kısıtta bulunan v_3, v_7, v_8 hızlarının değeri 5 değerine sabitlenmiş ve henüz sabitlenmeyen hızlar için **Hız_Alt_Sınırlarını_Bul** algoritması ikinci kez tekrarlanmıştır. Elde edilen değerler şu şekildedir;

Kısıt (114) için: $tBek = 69, tDelta = 9, tDist = 90, tHiz_1 = 1.19$

Kısıt (115) için: $tBek = 20, tDelta = 3, tDist = 30, tHiz_1 = 1.30$

Kısıt (116) için: $tBek = 16, tDelta = 5, tDist = 50, tHiz_1 = 2.38$

İkinci en büyük hız alt sınır değeri kısıt (116)'de gerçekleşmiş ve daha önce sabitlenmemiş olan v_{12}, v_1, v_2 hızlarının değerleri sabitlenmiştir. Kısıt (114) ve (115)'nin de sabitleme işlemi yapıldığında elde edilen hız değerleri şu şekildedir:

$v_{12} = 2.38, v_1 = 2.38, v_2 = 2.38, v_3 = 5, v_4 = 0.702, v_5 = 0.702, v_6 = 0.702, v_7 = 5, v_8 = 5$.

Böylece, üç yedi ve dokuzuncu pozisyonlarda gerçekleşen robot beklemeleri sıfıra eşitlenmiş ve verilen aktivite sıralaması için % 65 enerji tasarrufu sağlanmıştır.

4.4.4 ETF: Aşama 4

Aşama 3'te, verilen bir bir aktivite sıralaması için minimum enerji tüketimini sağlayan hız değerleri bulunmuştur. Adım 3'te korunmaya çalışılan çevrim zamanı limiti, λ kadar artırıldığında, minimum enerji tüketimini sağlayan aktivite sıralaması ve hız değerleri, yani Pareto etkin çözümler grafiğinde bulunan bir sonraki çözüm bu aşamada belirlenecektir. Aşama 3'te elde edilen B noktasından C 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.

Aşama 4.1'de bahsedilen, kritik hızların aldıkları yolu λ kadar artıracak yavaşlatma algoritması şu şekildedir; **Kritik_Hızları_Yavaşlat**

Adım 0: $tDist=0, tDelta=0, tHiz=0$

Esnek bir hızdan sonraki ilk kritik hız i ., önceki son kritik hız da j . pozisyonlarda gerçekleşiyor olsun.

Adım 1: Her k değeri için, $k \in i..j$ (Eğer $i > j$ ise $k \in i..Pozisyon\ Sayısı \cup \{1..j\}$)

→ $tDist := tDist + dist_k$

→ $tDelta := tDelta + delta_k$

Sonraki k

Adım 2: $tHiz := tDist \setminus (\lambda + tDelta)$

Adım 3: Kritik hızların değerini $tHiz$ değerine eşitle

Kritik hızların yavaşlatılmasıyla hem çevrim zamanı artırılmış, hem de enerji tüketim değeri azaltılmış olacaktır. Ancak, eğer $tHiz$ değeri, esnek hızların herhangi birisinden daha küçükse, aynı çevrim zamanını esnek hızların da yavaşlatılmasıyla daha az enerji tüketimiyle sağlamak mümkündür.

Bu durumda Aşama 3'te geliştirilen hız alt sınırı bulma algoritması kullanılmıştır. Bu algoritmanın 2. adımında bulunan hız hesaplama denklemi $tHiz_n := tDist \setminus (tBek + tDelta + \epsilon)$ olarak güncellenerek yeni esnek hız değerleri hesaplanmakta ve λ değeri ϵ kadar azaltılarak kritik_hızları_yavaşlat algoritması tekrarlanmaktadır. Bu algoritmadan elde edilen yavaşlatılmış kritik hız değeri esnek hızların herhangi birinden küçük olduğu sürece, λ değeri ϵ kadar azaltılarak belirtilen adımlar tekrar edilmektedir.

Aşama 4.2'de ise aynı çevrim zamanıyla daha küçük enerji tüketimini sağlayan olası bir aktivite sıralaması aranmaktadır. Aşama 1'de geliştirilen arama_algoritmasının 2. adımında belirlenen aday döngü, eğer verilen çevrim zamanı sınırını sağlıyor ise diğer adımlara geçilir. Eğer sağlamıyor ise yeni Adım 2 tekrarlanılarak olurlu aday döngü belirlenir. Çevrim zamanı sınırını sağlayan aktivite sıralamalarından minimum enerji tüketimini sağlayan döngü Aşama 4.2'nin aday çözümü olarak belirlenir.

Aşama 4.1 ve 4.2'nin aday çözümlerinden minimum enerji tüketimini sağlayan pareto etkin çözüm olarak kabul edilir.

4.4.5 ETA: Aşama 5

İstenen pareto etkin çözümü sayısınca, çevrim zamanı sınırı λ kadar artırılarak Aşama 4 tekrar edilir.

4.5 Hesaplamalı Deney Sonuçları

Bu bölümde öncelikli amaç projede geliştirilen etkin çözüm kümesi türetme algoritmasının (ETA) hesaplama performansını ortaya koymaktır. Bu bölümde ETA ve MISOCP performans karşılaştırmaları yapılacaktır. Son olarak ise tek tutuculu ve çift tutuculu robot alternatiflerinin karşılaştırılacaktır.

Deneylerde Bölüm 4.3 de oluşturulmuş olan 2 ve 3 makinelik veri setleri kullanılmıştır. MISOCP ve ETA yöntemlerinin türettiği etkin çözümler kümesinin ilk 5 çözümü için enerji ve çözüm süresi karşılaştırılmıştır.

Tablo 22. 2 makine, SA ve MIQCP karşılaştırması

Veri Seti	P	d	θ	Etkin Çözüm No.	MISOCP		ETA		Aktivite Sıralaması
					Çözüm Süresi (sn)	Sapma (%)	Çözüm Süresi (sn)		
1	GA	GT	1	1	3.06	0.00	2.61	Aynı	
				2	4.67	18.00	3.90	Farklı	
				3	4.77	0.00	5.25	Aynı	
				4	11.69	0.00	3.61	Aynı	
				5	14.59	0.00	3.45	Aynı	
2	GA	GT	5	1	12.44	5.21	1.40	Aynı	
				2	12.14	0.02	3.40	Aynı	
				3	13.00	0.00	3.39	Aynı	
				4	12.30	2.85	3.47	Aynı	
				5	12.06	2.49	3.66	Aynı	
3	GA	GT	2	1	12.84	7.18	0.42	Aynı	
				2	12.31	0.02	2.64	Aynı	
				3	13.70	21.82	2.63	Farklı	
				4	13.33	0.06	4.94	Aynı	
				5	12.36	0.00	3.67	Aynı	
4	GA	GT	3	1	13.05	0.00	0.42	Aynı	
				2	12.53	0.00	6.24	Aynı	
				3	13.63	0.00	4.28	Aynı	
				4	12.73	0.04	5.08	Aynı	
				5	12.84	0.01	3.17	Aynı	
5	GA	GT	1	1	13.05	0.00	2.80	Aynı	
				2	12.53	0.00	6.24	Aynı	
				3	13.63	0.00	4.28	Aynı	
				4	12.73	0.04	5.08	Aynı	
				5	12.84	0.01	3.17	Aynı	
6	GA	GT	7	1	11.89	0.58	2.50	Aynı	
				2	12.02	0.03	11.07	Aynı	
				3	13.58	0.02	3.90	Aynı	
				4	12.22	0.02	3.70	Aynı	
				5	12.20	0.01	3.58	Aynı	
7	DA	GT	15	1	12.92	0.00	0.50	Aynı	
				2	12.77	0.00	2.99	Aynı	
				3	12.36	0.01	2.13	Aynı	
				4	12.61	0.02	2.17	Aynı	
				5	12.14	0.04	2.38	Aynı	
8	DA	GT	3	1	11.75	2.17	2.20	Aynı	
				2	13.44	0.02	6.24	Aynı	
				3	11.59	0.05	3.89	Aynı	
				4	13.03	0.03	3.43	Aynı	
				5	11.58	2.76	3.42	Aynı	

Tablo 23. 2 makine, ETA ve MISOCP karşılaştırması(Devam)

Veri Seti	P	d	θ	Etkin Çözüm No.	MISOCP		ETA		Aktivite Sıralaması
					Çözüm Süresi (sn)	Sapma (%)	Çözüm Süresi (sn)		
9	DA	GT	11	1	12.77	0.00	0.38	Aynı	
				2	9.25	0.02	2.22	Aynı	
				3	13.34	0.00	2.31	Aynı	
				4	12.97	0.03	2.35	Aynı	
				5	12.75	0.01	4.02	Aynı	
10	DA	GT	1	1	11.80	0.00	2.10	Aynı	
				2	12.30	3.73	4.36	Farklı	
				3	12.70	0.00	6.09	Aynı	
				4	14.50	0.00	8.25	Aynı	
				5	14.59	0.05	6.81	Aynı	
12	GA	ÖT	8	1	12.08	0.00	0.74	Aynı	
				2	13.81	0.02	4.80	Aynı	
				3	12.36	0.03	4.77	Aynı	
				4	12.77	0.02	3.45	Aynı	
				5	11.75	0.02	3.43	Aynı	
13	GA	ÖT	7	1	5.92	0.00	0.36	Aynı	
				2	12.16	0.00	2.32	Aynı	
				3	11.97	0.02	1.98	Aynı	
				4	12.41	0.01	3.43	Aynı	
				5	11.70	0.01	3.53	Aynı	
14	GA	ÖT	1	1	3.95	82.39	0.77	Farklı	
				2	13.17	0.01	3.00	Aynı	
				3	3.33	0.00	4.02	Aynı	
				4	13.14	6.25	2.94	Aynı	
				5	11.89	6.25	2.95	Aynı	
15	GA	ÖT	4	1	4.09	25.76	1.40	Farklı	
				2	13.64	0.04	3.13	Aynı	
				3	12.16	0.00	4.05	Aynı	
				4	11.52	0.00	3.47	Aynı	
				5	11.20	0.00	3.47	Aynı	
16	DA	ÖT	9	1	12.50	0.00	0.71	Aynı	
				2	10.92	0.01	3.24	Aynı	
				3	5.25	0.01	14.59	Aynı	
				4	11.98	0.04	3.43	Aynı	
				5	12.00	0.05	3.41	Aynı	
17	DA	ÖT	7	1	11.86	0.00	0.78	Aynı	
				2	12.20	0.01	4.82	Aynı	
				3	12.52	0.03	3.41	Aynı	
				4	11.80	2.70	3.85	Aynı	
				5	11.88	2.19	3.95	Aynı	
18	DA	ÖT	4	1	11.89	11.60	0.47	Aynı	
				2	11.59	0.04	3.86	Aynı	
				3	12.22	0.03	4.36	Aynı	
				4	12.06	0.00	4.50	Aynı	
				5	11.44	0.00	3.81	Aynı	
19	DA	ÖT	9	1	12.09	0.00	0.37	Aynı	
				2	12.17	0.28	2.33	Aynı	
				3	11.53	0.00	2.88	Aynı	
				4	12.09	0.03	13.73	Aynı	
				5	11.77	0.04	3.60	Aynı	
20	DA	ÖT	3	1	12.13	0.00	3.10	Aynı	
				2	12.53	0.02	3.54	Aynı	
				3	12.22	0.05	3.66	Aynı	
				4	12.27	0.06	3.55	Aynı	
				5	11.42	0.01	3.80	Aynı	

Tablo 22 ve 23'de 2 makineden oluşan veri kümeleri kullanılarak, sezgisel algoritmanın MISOCP ile elde edilen optimal değere göre hata yüzde oranları belirlenmiştir. Ayrıca, iki çözüm yönteminin çözüme ulaşma süreleri ile her bir çözüm noktasında elde edilen aktivite sıralamasının aynı olup olmadığı da incelenmiştir. ETA, tablolarda bulunan çözüm noktalarının %78'inde optimal enerji tüketim değeri ve aktivite sıralamalarını elde etmiştir. Geriye kalan

%22'lik dilimin ise %13'ünde ise enerji tüketimi değerine en fazla %4 uzaklıkta olan değerlere ulaşılmıştır.

Tablo 22 ve 23 incelendiğinde, ETA'nın elde ettiği sonuçların en yüksek hata yüzde oranına sahip olduğu veri kümelerinin, makine işlem sürelerinin geniş aralıktan türetildiği kümeler olduğu görülmektedir. Bu durumda, ETA'nın düşük varyanslı işlem sürelerinde daha etkili sonuçlar vereceği çıkarımı yapılabilir. Geniş aralıklı işlem süreleri ile türetilen problem kümelerinde en yüksek hata yüzdesine sahip olanlar 14 ve 15. kümelerdir. Etkin çözüm kümelerinin ilk noktalarında optimal aktivite sıralaması elde edilemeyen bu iki veri seti, mutlak hata değerleri hesaplandığında aynı bantta bulunsa da, hata yüzde oranına göre değerlendirildiğinde veri seti 14 için hayli yüksek değerler elde edilmektedir.

Ayrıca, elde edilen sonuçlar incelendiğinde ETA'nın makineler arası uzaklığının özdeş ya da genel toplamsal olduğu durumlarda performansının belirgin ölçüde değişmediği görülmektedir. Pozisyonlar arası aktif tutucu değişimi süresi olan θ 'nın da ETA'nın performansını etkilemediği görülmüştür. Öyle ki θ değeri göreceli olarak daha yüksek olan veri setleri 7 ve 9'da optimal değerlere en fazla %0.04 oranında uzaklıkta olan değerler ve optimal aktivite sıralamaları elde edilmişken, θ değeri 1 olan veri seti 14'de en kötü performans elde edilmiştir.

Tablo 23'de yer verilmeyen veri seti 11 için ise ETA ve MISOCP performans karşılaştırmaları Tablo 24'te bulunmaktadır.

Tablo 24. Veri seti 11, ETA ve MISOCP karşılaştırması

P	d	θ	Etkin Çözüm No.	MISOCP				ETA	
				CT	Çözüm Süresi (sn)	CT	Sapma (%)	Çözüm Süresi (sn)	Aktivite Sıralaması
GA	ÖT	1	1	50.4	12.70	50.4	8.65	0.00	Aynı
			2	55.4	12.64	57.4	-7.90	3.49	Aynı
			3	60.4	13.74	62.4	-11.73	3.39	Aynı
			4	65.4	11.91	66.9	-6.78	3.32	Aynı
			5	70.4	14.58	71.9	-6.01	3.33	Aynı

Veri seti 11 için sonuçlar incelendiğinde, 1. etkin çözümden 2. noktaya geçerken, sezgisel algoritma ile istenen çevrim zamanı sınırı artışından daha yüksek bir artış gerçekleşmiştir. Diğer bir deyişle grafik 15'teki B noktasından C noktasına değil, C noktasının sağında bulunan bir pareto etkin çözüme geçilmiştir. Sezgisel algoritma ile daha büyük bir çevrim zamanı elde edildiğinden, daha küçük enerji tüketimleri gerçekleşmiştir. Ortaya çıkan aktivite sıralamaları her iki methodla da özdeş çıksa da, Çevrim zamanları farklı olan bu noktaların pareto etkin çözüm noktalar olup olmadığı hakkında bir karara varmak mümkün değildir.

Tablo 25. 3 makine, ETA ve MISOCP karşılaştırması

Veri Seti	P	d	θ	Etkin Çözüm No.	MISOCP		ETA		Aktivite Sıralaması
					Çözüm Süresi (sn)	Sapma (%)	Çözüm Süresi (sn)		
21	DA	ÖT	4	1	40.02	0.00	4.35	Aynı	
				2	35.22	-0.02	19.24	Aynı	
				3	41.52	0.01	17.34	Aynı	
				4	43.84	0.00	19.41	Aynı	
				5	40.70	0.00	17.48	Aynı	
22	DA	ÖT	1	1	30.00	0.00	3.23	Aynı	
				2	29.69	0.03	20.52	Aynı	
				3	30.86	0.02	18.52	Aynı	
				4	29.09	0.00	16.45	Aynı	
				5	29.64	-0.05	22.60	Aynı	
23	DA	ÖT	4	1	26.44	0.00	4.11	Aynı	
				2	27.50	-0.06	9.72	Aynı	
				3	30.39	-0.01	10.90	Aynı	
				4	24.17	0.00	18.76	Aynı	
				5	23.59	0.00	25.54	Aynı	
24	DA	ÖT	15	1	26.30	0.00	3.65	Aynı	
				2	41.20	-0.02	8.32	Aynı	
				3	39.59	-0.02	8.37	Aynı	
				4	19.73	0.02	8.41	Aynı	
				5	31.92	0.01	7.81	Aynı	
25	DA	ÖT	1	1	28.91	0.00	4.34	Aynı	
				2	34.64	0.00	18.05	Aynı	
				3	600.03	0.02	18.65	Aynı	
				4	600.03	0.00	17.59	Aynı	
				5	38.39	0.00	17.75	Aynı	
26	GA	ÖT	14	1	43.28	0.00	3.73	Aynı	
				2	25.34	-0.02	8.35	Aynı	
				3	27.78	-0.02	8.09	Aynı	
				4	30.97	0.02	7.54	Aynı	
				5	33.05	0.01	7.80	Aynı	
27	GA	ÖT	10	1	34.03	0.01	2.39	Aynı	
				2	22.28	0.01	8.38	Aynı	
				3	32.95	-0.03	9.94	Aynı	
				4	42.11	0.02	8.26	Aynı	
				5	33.41	0.00	8.11	Aynı	
28	GA	ÖT	8	1	27.55	0.00	2.19	Aynı	
				2	30.03	-0.02	8.19	Aynı	
				3	29.94	-0.02	7.78	Aynı	
				4	38.67	0.00	7.94	Aynı	
				5	32.20	0.04	8.81	Aynı	
29	GA	ÖT	1	1	27.64	9.29	1.61	Aynı	
				2	24.11	0.02	5.10	Aynı	
				3	38.95	0.00	5.55	Aynı	
				4	37.14	0.00	5.19	Aynı	
				5	34.39	0.05	5.33	Aynı	
30	GA	GT	1	1	32.75	0.00	4.17	Aynı	
				2	600.03	0.02	15.92	Aynı	
				3	600.05	0.01	15.63	Aynı	
				4	600.06	2.68	15.74	Aynı	
				5	600.03	2.36	16.47	Aynı	
31	GA	GT	2	1	35.69	-0.01	4.14	Aynı	
				2	43.19	0.03	32.89	Aynı	
				3	48.75	-0.04	32.45	Aynı	
				4	36.88	0.01	15.86	Aynı	
				5	36.75	-0.05	15.93	Aynı	
32	GA	GT	3	1	34.41	7.40	2.53	Farklı	
				2	37.99	38.10	24.81	Farklı	
				3	44.50	-0.01	16.96	Aynı	
				4	42.33	0.01	15.78	Aynı	
				5	42.47	3.08	15.46	Aynı	

Tablo 26. 3 makine, ETA ve MISOCP karşılaştırması(Devam)

Veri Seti	P	d	θ	Pareto Çözüm No.	MISOCP		ETA		Aktivite Sıralaması
					Çözüm Süresi (sn)	Sapma (%)	Çözüm Süresi (sn)		
33	GA	GT	1	1	23.36	0.00	2.18	Aynı	
				2	30.03	0.01	17.20	Aynı	
				3	32.23	4.97	18.83	Aynı	
				4	36.38	11.57	19.44	Aynı	
				5	36.23	3.08	15.27	Aynı	
34	GA	GT	1	1	32.88	0.00	2.51	Aynı	
				2	26.81	-0.01	39.04	Aynı	
				3	39.88	5.12	17.71	Aynı	
				4	44.27	3.92	15.66	Aynı	
				5	41.19	3.16	15.76	Aynı	
35	GA	GT	8	1	31.30	0.00	2.21	Aynı	
				2	29.95	-0.01	7.60	Aynı	
				3	28.88	-0.03	7.50	Aynı	
				4	36.91	0.04	7.60	Aynı	
				5	40.69	0.00	7.55	Aynı	
36	DA	GT	9	1	22.59	-0.01	2.12	Aynı	
				2	26.53	-0.02	7.55	Aynı	
				3	24.27	-0.03	7.59	Aynı	
				4	39.84	-0.03	7.76	Aynı	
				5	33.14	-0.03	7.55	Aynı	
37	DA	GT	5	1	33.19	0.00	2.67	Aynı	
				2	36.42	-0.02	15.42	Aynı	
				3	39.70	0.02	19.46	Aynı	
				4	35.55	4.32	15.89	Aynı	
				5	31.49	3.49	14.45	Aynı	
38	DA	ÖT	9	1	23.44	-0.12	2.19	Aynı	
				2	23.33	0.01	7.63	Aynı	
				3	24.66	0.00	7.69	Aynı	
				4	37.56	-0.04	7.66	Aynı	
				5	39.30	15.76	7.60	Farklı	
39	DA	GT	1	1	35.63	0.00	3.58	Aynı	
				2	21.55	-1.58	17.88	Aynı	
				3	23.52	-0.06	21.68	Aynı	
				4	48.19	-0.04	21.87	Aynı	
				5	44.28	3.08	18.33	Aynı	
40	DA	GT	7	1	24.98	0.00	2.17	Aynı	
				2	29.80	0.00	7.68	Aynı	
				3	34.67	0.00	22.39	Aynı	
				4	37.78	20.41	7.61	Farklı	
				5	41.55	0.02	15.85	Aynı	

Tablo 25 ve 26'de 3 makineli veri setleri için MISOCP model ve ETA'nın performans karşılaştırmaları yapılmıştır. İncelenen 20 veri kümesi ve toplamda 100 etkin çözüm için, ETA'nın %81 oranla optimal sonucu yakaladığı söylenebilmektedir. Göreceli olarak yüksek hata yüzde oranına sahip 29,32,33 ve 34. veri setleri incelendiğinde makine işlem sürelerinin geniş aralıktan türetildiği görülmektedir. Bu sonuç 2 makine için yapılan çıkarımları destekler niteliktedir.

Diğer yandan, makine sayısı arttığında artış gösteren pozisyon ve alternatif aktivite sıralama sayıları sebebiyle, bazı dar aralıktan türetilmiş veri setlerinde de istenilen etkinlikte çözümler elde edilememiştir. Veri seti 38 ve 40'ın dört ve beşinci çözüm noktalarında optimal aktivite sıralamaları elde edilememiş ve bu sebeple de optimal enerji tüketim değerlerine erişilememiştir. Sonuçlar θ değeri açısından değerlendirildiğinde, performans sonuçlarının θ değerine göre ayrışmadığı görülmektedir. Zira, tutucu değişimi süresi göreceli olarak yüksek veya düşük olduğunda da ETA ile elde edilen iyi ve kötü sonuç kümeleri bulunmaktadır. Benzer örnekler makineler arası uzaklık türüne göre ayrışan veri kümeleri için de verilebilir.

Öyleyse, makineler arası uzaklığın özdeş ya da genel toplamsal türünden olması ETA'nın performansını belirgin olarak etkilememektedir.

Elde edilen tüm sonucu ve çıkarımları genel olarak özetlemek gerekirse, 2 ve 3 makine için yapılan testlerde olduğu gibi grafik 15'de de B,C1,C2,C3 ve C4 olarak belirtilen etkin çözümlerin ilk 5 noktası karşılaştırma noktası olarak kullanılmıştır. Tüm hızlar en üst limitlerindeyken elde edilen minimum çevrim zamanı çözüme denk gelen A noktası ise tüm örnek gruplarında optimal olarak elde edilmiş ve performans ölçümlerine dahil edilmemiştir.

Elde edilen sonuçların %96'ında optimal aktivite sıralamaları ETA ile elde edilmiştir. Optimal aktivite sıralaması ile enerji tüketimi amaç fonksiyonunda gerçekleşen maksimum hata yüzde oranı %11, ortalama hata yüzde oranı ise %0.49'dur.

Performans sonuçları incelendiğinde, geniş aralıktan türetilen işlem süresine sahip örneklerin ETA'nın hata yüzde oranlarını etkilediği söylenebilir. Bunun yanında, geliştirilen algoritmanın farklı uzaklık türleri ve θ değerlerinde etkin çözümler vermeye yatkın olduğu yapılabilecek çıkarımlar arasındadır.

MISOCP modeli ile kısa sürede optimal çözümü elde edilen örnekler dışından, makul sürede çözüm alınamayan 5 makineli veri setleri için de ETA'nın performans testleri yapılmıştır. Tablo 27, alınan bazı koşullar için MISOCP ve ETA karşılaştırmalarını içermektedir.

Tablo 27. 5 makine, ETA ve MISOCP karşılaştırması

Veri Seti	Pareto Çözüm No.	MISOCP		ETA		Aktivite Sıralaması
		Sapma (%)	Çözüm Süresi (sn)	Sapma (%)	Çözüm Süresi (sn)	
1	minCT	0	1169	0	105.3	Aynı
	1	ÇY	ZL	-	105.4	-
	2	ÇY	ZL	-	90.3	-
	3	2051	ZL	87	93.2	Farklı
	4	1400	ZL	84	83.042	Farklı
	5	811	ZL	81	86.767	Farklı
2	minCT	0	3019	0	128.677	Aynı
	1	ÇY	ZL	-	129	-
	2	ÇY	ZL	-	87.089	-
	3	ÇY	ZL	-	91.102	-
	4	ÇY	ZL	-	95.693	-
	5	ÇY	ZL	-	93.016	-

Tüm hızlar üst limitlerine eşitken başarılabilir en küçük çevrim zamanı iki yöntem kullanılarak da hatasız olarak bulunmuştur. Çözüm elde etme süresi, MISOCP model ile en iyi ihtimalle yaklaşık 20 dakika iken, ETA ile bu süre 2 ila 3 dakika arasına düşmektedir. Veri seti 61 için, pareto optimal noktalar olan, 1 ve 2. noktalarda MISOCP ile 1 saat zaman diliminde herhangi bir sonuç alınamamış olup, ETA ile 2 dakikadan az sürede çözümler elde edilmiştir. 3, 4 ve 5. noktalarda ise, MISOCP model ile iyi birer alt sınır elde edilememiş ve bu sebeple yüksek hata yüzde oranları ortaya çıkmıştır. ETA ise bu üç noktada daha kaliteli çözümler bulmuş ve daha kısa sürede iyi çözümler elde edilebileceğini bu örnek için göstermiştir.

Ayrıca belirtmek gerekir ki hesaplamalı deneylerde beşer örnek için ETA ve MISOCP karşılaştırması yapılmış olsa da ETA kısa sürede çok fazla sayıda etkin çözüm türetebilmekte ve karar vericiye yüksek sayıda çözüm alternatifi sunabilmektedir.

Veri seti 62'de olduğu gibi, MISOCP model, denenen diğer 5 veri seti ile de zaman limitine takılmış ve etkin noktalar 1,2,3,4 ve 5 için bir çözüm üretememiştir. Bu sebeple Tablo27'e MISOCP modeli ile çözüm elde edilemeyen veri setleri eklenmemiştir.

4.6 Tek ve Çift Tutuculu m -Makinelik Robotik Hücrelerin Karşılaştırması

Bu bölümde tek ve çift tutuculu robotik hücreler enerji tüketimi amacı açısından karşılaştırılacaktır. Sırasıyla robotun tek tutuculu olduğu, robotun çift tutuculu olduğu ve enerji tüketiminin amaç fonksiyonunu etkilemediği ve son olarak robotun çift tutuculu olduğu ve enerji tüketiminin amaç fonksiyonunu etkilediği durumlar ele alınmaktadır.

Tablo 28'de daha önceki testlerde kullanılan ve özellikleri Tablo 11'de tablo numarası verilen 3-makinelik test problemlerinden C_f ve C_e parametrelerinin eşit (C) olduğu problemler kullanılmıştır. Ayrıca, bu test problemlerinde olduğundan farklı olarak bu testlerde 1-birim döngüler ele alındığından n değeri 1 olarak alınmıştır. Çift tutuculu robotik hücreler için kullanılan G parametresi ise (tutucu değiştirme sırasındaki enerji tüketim katsayısı) yapılan testlerde iki farklı değer almaktadır. G değeri ilk olarak sıfır, ikinci olarak ise bahsedilen C değerinin iki katı olarak alınmıştır. Ayrıca, çift tutuculu robotun bulunduğu sistemde θ değeri, yani robotun tutucusunu değiştirme süresi ε değerine eşit (1) olarak alınmıştır.

Test problemleri için bulunan optimal 1-birim döngüyü belirtmek için aşağıdaki notasyon tanımlanmıştır:

U_i : Robotun M_i makinesini boşaltması

L_i : Robotun M_i makinesini yüklemesi

$D_{i,j}$: Robotun M_i makinesinden M_j makinesine tutucularından en az biri dolu olarak hareket etmesi

$B_{i,j}$: Robotun M_i makinesinden M_j makinesine tutucuları boş olarak hareket etmesi

Bu notasyonu kullanarak aşağıdaki döngüleri tanımlayabiliriz:

$$S1 : U_0 - D_{0,1} - L_1 - U_1 - D_{1,2} - L_2 - U_2 - D_{2,3} - L_3 - U_3 - D_{3,4} - L_4 - B_{4,0}$$

$$S2 : U_0 - D_{0,1} - L_1 - U_1 - D_{1,2} - L_2 - B_{2,3} - U_3 - D_{3,4} - L_4 - B_{4,2} - U_2 - D_{2,3} - L_3 - B_{3,0}$$

$$S3 : U_0 - D_{0,1} - L_1 - B_{1,2} - U_2 - D_{2,3} - L_3 - B_{3,1} - U_1 - D_{1,2} - L_2 - B_{2,3} - U_3 - D_{3,4} - L_4 - B_{4,0}$$

$$S4 : U_0 - D_{0,1} - L_1 - B_{1,2} - U_2 - D_{2,3} - L_3 - U_3 - D_{3,4} - L_4 - B_{4,1} - U_1 - D_{1,2} - L_2 - B_{2,0}$$

$$S5 : U_0 - D_{0,1} - L_1 - B_{1,3} - U_3 - D_{3,4} - L_4 - B_{4,1} - U_1 - D_{1,2} - L_2 - U_2 - D_{2,3} - L_3 - B_{3,0}$$

$$S6 : U_0 - D_{0,1} - L_1 - B_{1,3} - U_3 - D_{3,4} - L_4 - B_{4,2} - U_2 - D_{2,3} - L_3 - B_{3,1} - U_1 - D_{1,2} - L_2 - B_{2,0}$$

$$S7 : U_0 - D_{0,1} - U_1 - L_1 - D_{1,2} - U_2 - L_2 - D_{2,3} - U_3 - L_3 - D_{3,4} - L_4 - B_{4,0}$$

$$S8 : U_0 - D_{0,1} - L_1 - D_{1,2} - U_2 - L_2 - D_{2,3} - U_3 - D_{3,4} - L_4 - D_{4,3} - L_3 - B_{3,1} - U_1 - D_{1,0}$$

$$S9 : U_0 - D_{0,1} - L_1 - B_{1,2} - U_2 - D_{2,3} - U_3 - D_{3,4} - L_4 - D_{4,3} - L_3 - B_{3,1} - U_1 - D_{1,2} - L_2 - B_{2,0}$$

$$S10 : U_0 - D_{0,1} - L_1 - B_{1,2} - U_2 - D_{2,3} - L_3 - B_{3,1} - U_1 - D_{1,2} - L_2 - B_{2,3} - U_3 - D_{3,4} - L_4 - B_{4,0}$$

$$S11 : U_0 - D_{0,1} - L_1 - D_{1,2} - U_2 - D_{2,3} - L_3 - U_3 - D_{3,4} - L_4 - D_{4,2} - L_2 - B_{2,1} - U_1 - D_{1,0}$$

$$S12 : U_0 - D_{0,1} - L_1 - U_1 - D_{1,2} - L_2 - B_{2,3} - U_3 - D_{3,4} - L_4 - B_{4,2} - U_2 - D_{2,3} - L_3 - B_{3,0}$$

$$S13 : U_0 - D_{0,1} - L_1 - U_1 - D_{1,2} - L_2 - B_{2,3} - U_3 - D_{3,4} - L_4 - D_{4,3} - L_3 - B_{3,2} - U_2 - D_{2,0}$$

$$S14 : U_0 - D_{0,1} - L_1 - D_{1,2} - U_2 - L_2 - D_{2,3} - U_3 - L_3 - D_{3,4} - L_4 - B_{4,1} - U_1 - D_{1,0}$$

$$S15 : U_0 - D_{0,1} - U_1 - L_1 - D_{1,2} - L_2 - U_2 - D_{2,3} - U_3 - L_3 - D_{3,4} - L_4 - B_{4,0}$$

$$S16 : U_0 - D_{0,1} - L_1 - D_{1,2} - U_2 - D_{2,3} - L_3 - D_{3,2} - L_2 - B_{2,3} - U_3 - D_{3,4} - L_4 - B_{4,1} - U_1 - D_{1,0}$$

$$S17 : U_0 - D_{0,1} - U_1 - L_1 - D_{1,2} - U_2 - L_2 - D_{2,3} - U_3 - D_{3,4} - L_4 - D_{4,3} - L_3 - B_{3,0}$$

Tablo 28. Tek ve Çift Tutuculu Robotun Bulunduğu Farklı Sistemlerin Enerji Tüketimi Açısından Karşılaştırması

No.	Alt Prob. Numarası	Çift Tutuculu Robot															
		Tek Tutuculu Robot				$G = 0$								$G = 2C$			
		Döngü	Çevrim Zamanı	Enerji Tüketimi	Çözüm Süresi (sn)	Döngü	Çevrim Zamanı	Enerji Tüketimi	Çözüm Süresi (sn)	Döngü	Çevrim Zamanı	Enerji Tüketimi	Çözüm Süresi (sn)				
1	1	S6	20.67	183.5	10.028	S8	20.67	30.0	23.375	S8	20.67	38.1	25.328				
	2	S6	32.00	24.0	17.261	S8	25.00	16.0	40.641	S9	32.00	22.0	29.671				
	3	S6	43.33	24.0	15.126	S8	25.00	16.0	12.297	S10	36.06	20.0	24.531				
	4	S4	54.67	20.0	22.608	S8	25.00	16.0	21.782	S4	46.07	18.0	26.422				
	5	S1	66.00	16.0	9.122	S8	25.00	16.0	40.281	S1	66.00	16.0	23.656				
3	1	S3	60.00	1404.0	12.542	S8	60.00	432.0	27.703	S8	60.00	440.5	28.297				
	2	S4	83.50	580.9	16.125	S8	83.50	202.0	16.625	S8	83.50	210.7	28.250				
	3	S2	107.0	304.1	14.423	S8	107.00	117.0	39.016	S8	107.00	125.2	27.219				
	4	S1	130.5	173.6	20.212	S8	113.00	104.0	22.922	S13	126.29	108.0	26.532				
	5	S1	154.0	104.0	18.714	S8	113.00	104.0	13.860	S1	154.00	104.0	25.156				
5	1	S2	102.0	2160.0	13.264	S8	102.00	818.4	31.172	S8	102.00	826.4	32.562				
	2	S1	137.0	935.1	11.184	S8	137.00	432.0	32.938	S8	137.00	440.0	32.250				
	3	S1	172.0	475.5	18.267	S8	172.00	266.6	74.516	S8	172.00	274.6	34.688				
	4	S1	207.0	287.2	15.612	S7	203.00	192.0	27.016	S14	202.02	200.0	28.843				
	5	S1	242.0	192.0	7.165	S7	203.00	192.0	26.593	S1	242.00	192.0	28.000				
7	1	S2	138.67	3149.8	22.263	S8	138.67	1305.8	75.328	S8	138.67	1313.8	37.156				
	2	S1	186.50	1178.1	24.548	S8	186.50	697.0	78.656	S8	186.50	705.0	41.547				
	3	S1	234.33	646.0	14.931	S8	234.33	432.6	32.813	S8	234.33	440.6	34.594				
	4	S1	282.17	407.2	10.264	S8	282.17	294.5	28.625	S8	282.17	302.2	30.531				
	5	S1	330.0	280.0	17.125	S14	290.07	280.0	62.578	S1	330.00	280.0	28.094				
9	1	S1	172.67	3311.8	17.592	S8	172.67	1860.8	71.407	S8	172.67	1868.8	34.625				
	2	S1	234.00	1472.0	12.017	S8	234.00	984.8	29.328	S8	234.00	992.8	32.157				
	3	S1	295.33	827.9	11.598	S8	295.33	608.4	73.062	S8	295.33	616.4	32.047				
	4	S1	356.67	529.9	8.124	S8	356.67	412.9	24.703	S8	356.67	420.9	29.187				
	5	S1	418.00	368.0	11.531	S7	379.07	368.0	71.500	S1	418.00	368.0	30.453				
11	1	S6	20.67	112.8	9.156	S8	20.67	30.0	41.578	S8	20.67	38.1	27.203				
	2	S6	30.50	27.3	10.046	S8	25.00	16.0	20.906	S8	25.00	24.0	28.093				
	3	S6	40.33	24.0	12.711	S8	25.00	16.0	25.922	S10	34.04	20.0	24.609				
	4	S2	50.17	20.0	14.165	S8	25.00	16.0	40.063	S12	42.06	18.0	23.187				
	5	S1	60.00	16.0	8.291	S8	25.00	16.0	10.750	S1	60.00	16.0	17.672				
13	1	S3	60.00	1404.0	14.514	S8	60.00	432.5	55.797	S8	60.00	440.5	27.407				
	2	S2	82.00	536.4	10.169	S8	82.00	211.1	24.235	S8	82.00	219.1	32.422				
	3	S2	104.00	297.1	17.256	S8	104.00	124.7	26.266	S8	104.00	132.7	27.016				
	4	S1	126.00	167.3	21.068	S8	113.02	104.0	42.312	S13	122.18	108.0	24.172				
	5	S1	148.00	104.0	8.945	S8	114.27	104.0	24.187	S1	148.00	104.0	25.516				
15	1	S2	98.00	2160.0	18.354	S8	98.00	893.7	72.766	S8	98.00	901.7	32.937				
	2	S1	132.50	903.7	11.150	S8	132.50	464.2	24.031	S8	132.50	472.2	32.234				
	3	S1	167.00	467.8	17.471	S8	167.00	283.7	16.156	S8	167.00	291.7	33.406				
	4	S1	201.50	285.3	20.402	S8	201.00	192.0	71.469	S8	201.00	200.0	26.328				
	5	S1	236.00	192.0	15.944	S7	203.00	192.0	74.359	S1	236.00	192.0	31.515				
17	1	S2	134.67	3149.8	12.171	S8	134.67	1390.3	78.563	S8	134.67	1398.3	32.766				
	2	S1	182.00	1152.6	9.563	S8	182.00	733.7	80.656	S8	182.00	741.7	33.782				
	3	S1	229.33	639.1	16.297	S8	229.33	452.5	77.578	S8	229.33	460.5	33.688				
	4	S1	276.67	405.5	10.562	S8	276.67	306.7	28.875	S8	276.67	314.7	27.265				
	5	S1	324.00	280.0	12.519	S15	304.04	280.0	68.282	S1	324.00	280.0	29.438				
19	1	S1	166.67	3311.8	21.563	S8	166.67	2005.2	62.922	S8	166.67	2013.2	33.953				
	2	S1	228.00	1472.0	14.692	S8	228.00	1039.5	30.688	S8	228.00	1047.5	33.672				
	3	S1	289.33	828.0	15.469	S8	289.33	634.7	75.406	S8	289.33	642.7	35.172				
	4	S1	350.67	529.9	21.044	S8	350.67	427.5	67.531	S8	350.67	435.5	29.781				
	5	S1	412.00	368.0	11.152	S7	388.08	368.0	26.015	S1	412.00	368.0	29.594				
21	1	S6	22.67	100.4	12.562	S8	22.67	21.9	45.094	S8	22.67	29.9	28.735				
	2	S6	32.00	24.0	17.932	S8	25.00	16.0	37.656	S10	32.00	20.0	26.672				
	3	S6	41.33	24.0	9.571	S8	25.00	16.0	20.563	S10	32.00	20.0	22.860				
	4	S2	50.67	20.0	14.563	S8	25.00	16.0	30.719	S12	44.05	18.0	24.172				
	5	S1	60.00	16.0	10.976	S8	25.00	16.0	13.453	S1	60.00	16.0	24.859				
23	1	S3	60.00	1404.0	17.552	S8	60.00	432.5	43.234	S8	60.00	440.5	27.875				
	2	S2	82.00	536.4	12.463	S8	82.00	211.1	27.141	S8	82.00	219.1	20.578				
	3	S2	104.00	297.1	10.639	S8	104.00	124.7	50.516	S8	104.00	132.6	29.313				
	4	S1	126.00	167.3	17.136	S8	113.01	104.0	48.844	S11	122.01	108.0	29.828				
	5	S1	148.00	104.0	9.263	S1	114.40	104.0	41.141	S1	148.00	104.0	25.250				
25	1	S2	98.00	2160.0	12.579	S8	98.00	893.7	28.781	S8	98.00	901.7	37.000				
	2	S1	132.50	903.7	15.204	S8	132.50	464.2	32.735	S8	132.50	472.2	30.297				
	3	S1	167.00	467.8	31.066	S8	167.00	283.7	77.047	S8	167.00	291.7	28.265				
	4	S1	201.50	285.3	9.992	S8	201.02	192.0	23.344	S8	201.02	200.0	30.609				
	5	S1	236.00	192.0	13.561	S17	202.01	192.0	69.296	S1	236.00	192.0	31.687				
27	1	S2	134.67	3149.8	24.690	S8	134.67	1390.3	87.687	S8	134.67	1398.3	37.297				
	2	S1	182.00	1152.6	23.543	S8	182.00	733.7	83.062	S8	182.00	741.7	31.968				
	3	S1	229.33	639.1	25.569	S8	229.33	452.5	81.375	S8	229.33	460.5	39.219				
	4	S1	276.67	405.5	10.592	S8	276.67	306.7	28.281	S8	276.67	314.5	33.531				
	5	S1	324.00	280.0	14.623	S1	300.03	280.0	146.422	S1	324.00	280.0	31.172				
29	1	S1	166.67	3311.8	11.695	S8	166.67	2005.2	106.937	S8	166.67	2013.2	31.703				
	2	S1	228.00	1471.9	10.461	S8	228.00	1039.5	29.812	S8	228.00	1047.5	37.203				
	3	S1	289.33	827.9	18.036	S8	289.33	634.7	36.172	S8	289.33	642.7	29.891				
	4	S1	350.67	529.9	9.596	S8	350.67	427.5	29.484	S8	350.67	435.5	29.718				
	5	S1	412.00	368.0	13.018	S1	394.13	368.0	24.406	S1	412.00	368.6	29.297				

		Çift Tutuculu Robot											
		Tek Tutuculu Robot				G = 0				G = 2C			
No.	Alt Prob. Numarası	Döngü	Çevrim Zamanı	Enerji Tüketimi	Çözüm Süresi (sn)	Döngü	Çevrim Zamanı	Enerji Tüketimi	Çözüm Süresi (sn)	Döngü	Çevrim Zamanı	Enerji Tüketimi	Çözüm Süresi (sn)
31	1	S6	22.00	156.97	8.517	S8	22.00	24.7	16.734	S8	22.00	32.7	26.281
	2	S6	33.00	35.12	13.809	S8	25.00	16.0	16.938	S8	25.00	24.0	26.812
	3	S6	44.00	28.00	21.312	S8	25.00	16.0	40.375	S11	39.06	20.0	25.343
	4	S2	55.00	20.00	8.104	S8	25.00	16.0	20.656	S12	48.05	18.0	22.469
	5	S1	66.00	16.00	11.154	S8	25.00	16.0	19.875	S1	66.00	16.0	24.703
33	1	S2	65.33	1170.0	20.350	S8	65.33	354.5	35.891	S8	65.33	362.5	31.625
	2	S2	87.50	512.08	13.716	S8	87.50	182.6	45.484	S8	87.50	190.6	34.797
	3	S2	109.67	285.86	16.773	S8	109.67	111.0	34.860	S8	109.67	119.1	35.172
	4	S1	131.83	167.97	19.279	S8	113.00	104.0	14.844	S13	126.17	108.0	27.047
	5	S1	154.0	104.0	10.444	S8	113.00	104.0	56.516	S1	154.00	104.0	28.031
35	1	S2	102.00	2160.0	14.905	S8	102.00	818.5	29.312	S8	102.00	826.5	31.750
	2	S1	137.00	935.12	18.157	S8	137.00	432.1	78.359	S8	137.00	440.1	33.953
	3	S1	172.00	475.54	7.278	S8	172.00	266.6	77.765	S8	172.00	274.6	32.532
	4	S1	207.0	287.15	9.656	S7	203.00	192.0	70.313	S8	202.03	200.0	29.093
	5	S1	242.0	192.0	8.762	S8	203.00	192.0	23.954	S1	242.00	192.0	25.438
37	1	S2	138.83	3154.4	24.487	S8	138.83	1302.5	71.469	S8	138.83	1310.5	29.453
	2	S1	186.63	1176.0	17.126	S8	186.63	696.1	30.797	S8	186.63	704.1	27.813
	3	S1	234.42	645.5	11.891	S8	234.42	432.4	27.000	S8	234.42	440.4	30.578
	4	S1	282.21	407.1	9.215	S8	282.21	294.5	68.703	S8	282.21	302.5	27.703
	5	S1	330.00	280.0	8.750	S7	304.06	280.0	72.734	S1	330.00	280.0	24.922
39	1	S1	172.67	3311.8	13.146	S8	172.67	1860.9	66.406	S8	172.67	1868.9	28.984
	2	S1	234.00	1471.9	9.794	S8	234.00	984.7	27.640	S8	234.00	993.0	16.015
	3	S1	295.33	827.9	8.466	S8	295.33	608.3	32.172	S8	295.33	616.5	32.000
	4	S1	356.67	529.9	7.689	S8	356.67	412.9	26.500	S8	356.67	421.0	24.953
	5	S1	418.00	368.0	10.053	S7	405.13	368.0	56.984	S1	418.00	368.0	28.735
41	1	S6	22.00	148.9	15.673	S8	22.00	24.5	22.953	S8	22.00	32.7	27.125
	2	S6	31.50	39.7	13.085	S8	25.00	16.0	19.547	S8	25.00	24.0	24.062
	3	S6	41.00	28.0	12.693	S8	25.00	16.0	20.594	S11	37.06	20.0	28.687
	4	S2	50.50	20.0	16.829	S8	25.00	16.0	16.578	S12	44.05	18.0	14.531
	5	S1	60.00	16.0	8.514	S8	25.00	16.0	20.562	S1	60.00	16.0	25.484
43	1	S2	61.33	1170.0	10.611	S8	61.33	410.7	23.984	S8	61.33	418.7	23.531
	2	S2	83.00	520.0	14.388	S8	83.00	205.5	42.797	S8	83.00	213.5	32.625
	3	S2	104.67	292.5	24.354	S8	104.67	123.0	72.610	S8	104.67	131.0	27.500
	4	S1	126.33	165.9	17.080	S8	113.00	104.0	40.938	S11	124.17	108.0	30.672
	5	S1	148.00	104.0	8.858	S8	113.00	104.0	68.844	S1	148.00	104.0	25.266
45	1	S2	98.00	2160.0	24.418	S8	98.00	893.7	29.563	S8	98.00	901.7	35.140
	2	S1	132.50	903.7	16.033	S8	132.50	464.2	25.453	S8	132.50	472.2	33.484
	3	S1	167.00	467.8	22.728	S8	167.00	283.7	46.843	S8	167.00	291.7	31.938
	4	S1	201.50	285.3	17.502	S8	201.00	192.0	73.953	S8	201.02	200.0	31.609
	5	S1	236.0	192.0	11.614	S7	203.02	192.0	75.781	S1	236.00	192.0	27.281
47	1	S2	134.83	3154.4	18.739	S8	134.83	1386.6	27.750	S8	134.83	1394.6	27.750
	2	S1	182.13	1150.6	16.751	S8	182.13	732.7	26.344	S8	182.13	740.7	31.860
	3	S1	229.42	638.5	14.969	S8	229.42	452.2	29.750	S8	229.42	460.2	31.719
	4	S1	276.71	405.4	12.494	S8	276.71	306.7	29.062	S8	276.71	314.7	24.813
	5	S1	324.00	280.0	11.727	S7	302.06	280.0	67.610	S1	324.00	280.0	24.875
49	1	S1	166.67	3312.0	15.617	S8	166.67	2005.3	59.407	S8	166.67	2013.3	31.469
	2	S1	228.00	1472.0	10.008	S8	228.00	1039.6	27.703	S8	228.00	1047.6	35.765
	3	S1	289.33	828.0	9.865	S8	289.33	634.8	27.562	S8	289.33	642.8	21.797
	4	S1	350.67	529.9	11.201	S8	350.67	427.6	26.890	S8	350.67	435.6	31.625
	5	S1	412.00	368.0	9.428	S7	388.09	368.0	73.000	S1	412.00	368.0	30.437
51	1	S6	24.00	136.4	11.359	S8	24.00	18.2	24.078	S8	24.00	26.2	25.641
	2	S6	33.00	35.1	10.025	S8	25.00	16.0	19.750	S16	33.00	22.0	26.000
	3	S3	42.00	28.0	14.983	S8	25.00	16.0	22.016	S11	37.09	20.0	14.984
	4	S2	51.00	20.0	17.152	S8	25.00	16.0	22.593	S12	46.09	18.0	22.328
	5	S1	60.00	16.0	13.955	S8	25.00	16.0	11.672	S1	60.00	16.0	26.578
53	1	S2	62.33	1121.6	13.669	S8	62.33	395.5	64.484	S8	62.33	403.5	28.094
	2	S2	83.75	508.2	14.457	S8	83.75	201.4	25.000	S8	83.75	209.4	28.703
	3	S2	105.17	289.2	11.831	S8	105.17	121.7	23.063	S8	105.17	129.7	34.844
	4	S1	126.58	164.9	12.321	S8	113.03	104.0	41.766	S11	122.37	108.0	30.515
	5	S1	148.00s	104.0	9.863	S8	113.13	104.0	41.781	S1	148.00	104.0	20.546
55	1	S2	98.00	2160.0	15.375	S8	98.00	893.7	29.063	S8	98.00	901.7	21.094
	2	S1	132.50	903.7	11.924	S8	132.50	464.2	26.610	S8	132.50	472.2	29.438
	3	S1	167.00	467.8	10.277	S8	167.00	283.7	28.266	S8	167.00	291.7	31.797
	4	S1	201.50	285.3	10.123	S8	201.01	192.0	24.297	S8	201.01	200.0	29.015
	5	S1	236.00	192.0	9.140	S17	202.03	192.0	22.672	S1	236.00	192.0	20.719
57	1	S2	134.83	3154.4	15.651	S8	134.83	1386.6	30.297	S8	134.83	1394.6	29.141
	2	S1	182.10	1151.0	18.900	S8	182.10	732.9	28.547	S8	182.10	740.9	30.360
	3	S1	229.37	638.9	12.753	S8	229.37	452.4	27.953	S8	229.37	460.4	32.344
	4	S1	276.63	405.6	12.140	S8	276.63	306.9	24.344	S8	276.63	314.9	25.156
	5	S1	323.90	280.2	11.517	S1	300.03	280.2	20.063	S1	323.90	280.2	25.250
59	1	S1	166.67	3312.0	17.595	S8	166.67	2005.3	28.125	S8	166.67	2013.3	33.796
	2	S1	228.00	1472.0	11.061	S8	228.00	1039.6	28.563	S8	228.00	1047.6	31.437
	3	S1	289.33	828.0	10.635	S8	289.33	634.8	19.203	S8	289.33	642.8	31.422
	4	S1	350.67	529.9	12.177	S8	350.67	427.6	16.312	S8	350.67	435.6	27.656
	5	S1	412.00	368.0	9.387	S1	379.18	368.0	25.032	S1	412.00	368.7	22.281

Test problemlerinden elde ettiğimiz sonuçlara baktığımızda, öncelikle çift tutuculu robotun olduğu sistemlerin tek tutuculu robotun bulunduğu sistemlere göre enerji tüketimi açısından daha iyi olduğu açıkça görülebilir. Çift tutuculu robotlar aynı anda iki parça taşıyabileceğinden

ilk noktaya karşılık gelen döngü tek tutuculu robotun olduğu bir sistemde olurlu olmayan $S8$ döngüsüdür. Bu döngü için en düşük çevrim zamanı aynı problemin tek tutuculu robot durumundaki ilk noktadaki çevrim zamanı üst sınırından daha düşüktür ve bu döngüde robot hareket hızları yavaşlatılabilir. Ancak, tek tutuculu robot için ilk nokta çevrim zamanı amacı açısından optimal çözümün çevrim zamanına eşit alındığından ilk noktaların hepsi için çift tutuculu robotun bulunduğu sistemler görece oldukça düşük enerji tüketim değerine sahiptir. Dolayısıyla, çevrim zamanının düşük olmasının istendiği durumlarda çift tutuculu robotun kullanılmasının enerji tüketimi amacı açısından daha iyi olacağı sonucuna varılabilir.

Son noktalarda, yani çevrim zamanı üst sınırı yeterince büyük olduğunda, robotun tutucu sayısının enerji tüketimi açısından herhangi bir etkisi kalmamaktadır. Ancak, robotun tutucu değiştirirken harcadığı enerjinin önemsenmeyecek kadar küçük olduğu durumlarda çift tutuculu robotun bulunduğu sistemlerde robot daha çevrim zamanına sahip bir döngüde aynı enerji tüketimine sahip olabilmektedir. Örneğin, 1 numaralı test probleminde tek tutuculu robotun olduğu sistemde çevrim zamanı üst sınırının 66 olduğu durumdaki enerji tüketim değeri $G = 0$ için çevrim zamanı üst sınırı 25 olduğunda sağlanabilmektedir. Benzer bir durumun bazı problemlerde $G > 0$ için de geçerli olduğu görülebilir. Yine 1 numaralı test problemine bakacak olursak; tek tutuculu robot için çevrim zamanı üst sınırı 54,67 iken bulunan enerji tüketim değeri, $G = 2C$ ve sistemdeki robot çift tutuculu olduğunda çevrim zamanı üst sınırı 36,06 iken elde edilebilmektedir. Yani, bu problem için çift tutuculu robot kullanıldığında aynı enerji tüketim değeri yaklaşık % 34'lük daha düşük bir çevrim zamanı içerisinde sağlanabilmektedir. Ayrıca, çift tutuculu sistemlerde daha farklı döngüler de olurlu olabildiğinden aynı çevrim zamanı üst sınırı için çift tutuculu sistemler daha düşük enerji tüketimine sahip olabilmektedir. Örneğin, 47 numaralı problemin 4. alt probleminde tek tutuculu sistemde optimal çözüm $S1$ döngüsüne sahip olup enerji tüketimi 405,4 iken; çift tutuculu sistemler $S8$ döngüsüne sahiptir ve sırasıyla 306,7 ve 314,7'lik enerji tüketim değerlerine sahiptir. Dolayısıyla, aynı çevrim zamanı üst sınırı için çift tutuculu robot kullanımı sırasıyla % 24,3 ve 22,4'lük enerji tasarrufu sağlamaktadır.

Sonuç olarak, tüm problemlerde son nokta dışında tüm noktalarda çift tutuculu robotun olduğu sistemlerin hem çevrim zamanı hem enerji tüketimi açısından tek tutuculu robotun olduğu sistemlere göre daha iyi olduğu görülebilir. Son noktalarda ise enerji tüketimi açısından çift veya tek tutuculu robot kullanımı eşdeğerdir. Ancak, çift tutuculu robot kullanımı ile çevrim zamanından kazanç sağlanabilir.

4.7 Sonuç ve Değerlendirme

Bu bölümde tek tip parça üreten çift tutuculu robotlu hücrelerdeki robot hareket sıralamasının belirlenmesi ve robot hareket hızlarının belirlenmesi problemleri eşzamanlı olarak ele alınmıştır. Bu problem için m -makinelik genel robotik hücrelerdeki problemi çözebilmek için bir Karma Tamsayı Doğrusal Olmayan matematiksel model geliştirilmiş ve yapılan testlerle doğrulaması gerçekleştirilmiştir. Fakat küçük problem boyurlarında 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. Sezgisel algortima JAVA programlama dilinde kodlanmıştır. Test problemleri türetilerek sezgiselin ve matematiksel modellerin kapsamlı performans testleri yapılmıştır. Son olarak, teknoloji ve yatırım maliyeti açısından tek tutuculu robotlara göre daha yüksek ihtiyaçları olan çift tutuculu robotların sisteme getirdikleri faydalar belirlenmeye çalışılmıştır. Bu amaçla tek ve çift tutuculu robotlar çevrim zamanı ve enerji tüketimi açısından birbirleriyle karşılaştırılmıştır. Yapılan testlerde çift tutuculu robotların, farklı test verilerinde Pareto önyüz üzerindeki noktaların neredeyse tamamında daha iyi performans gösterdiği belirlenmiştir.

5. DEĞİŞİK TİP PARÇA ÜRETEK TEK TUTUCULU ROBOTİK HÜCRELER

5.1 Giriş

Proje kapsamında ele alınan problemlerden bir tanesi de değişik tip parça üreten tek tutuculu robotlu hücrelerde robot hareket sıralaması, parça sıralaması ve robot hareket hızlarının eş zamanlı belirlenmesi problemidir. Bu problem, farklı tip parçalar içerdiği ve parça sıralamasının belirlenmesi kararlarının da diğer kararlarla beraber verilmesi gerektiği için tek tutuculu tek tip parça üreten sistemlerde ve çift tutuculu tek tip parça üreten sistemlere göre daha karmaşık bir problemidir. Bu problem versiyonunda, hız değerleri sabit olsa bile Hall vd. [49] 2-makineli bir sistemde bile 1-birim döngülerin optimallik garantisinin olmadığını ispatlamıştır. Bu sebeple, bu çalışmada geliştirilecek çözüm yöntemleri optimal n -birim döngüleri belirleyecek şekilde geliştirilecektir.

Bu alandaki çalışmaların tamamında robotun yaptığı bütün hareketleri mümkün olan en yüksek hızda gerçekleştirdiği varsayılmaktadır. Dolayısıyla, robot hareketleriyle ilgili süreler sabit birer problem parametresidir. Ancak bu varsayım bazı anlarda robotun bir sonraki hareket başlayana kadar boşta beklemesine sebep olmaktadır. Bu durum robotun fazladan enerji harcamasına sebep olmaktadır. Robotların harcadıkları enerji miktarı hız, taşınan yükün ağırlığı gibi faktörlere bağlıdır. Hız değeri değiştirilerek harcanan enerji miktarı kontrol edilebilir. Aynı zamanda eğer yapılması gereken iş henüz tamamlanmamışsa hem hızlı gidip hem de fazladan beklemek yerine, hızını düşürerek bu sayede enerji israfı bertaraf edilebilir. Dolayısıyla, bu parametrelerin birer karar değişkeni olması üretim hızının yanında, sistemde harcanan enerjinin de optimize edilmesini sağlayarak sistemden maksimum faydanın elde edilmesini sağlayacaktır.

Projenin bu kısmında, optimal parça ve robot hareket sıralamasının yanında robotların optimal hızlarının da belirlenmesi hedeflenmektedir. Üretim hızı maksimizasyonu ve enerji tüketimi minimizasyonunun beraberce ele alındığı 2-kriterli bir optimizasyon modeli oluşturulacaktır. Ele alınan hedefler birbirleriyle çelişen nitelikte olduğundan tek bir optimal çözüm yerine, başatlanmayan (etkin, nondominated) çözümler kümesi belirlenecektir.

Makine sayısının ve parça sayısının küçük olduğu örneklerde bile karmaşıklık oldukça fazladır. Bu sebeple, öncelikle m -makineli genel robotik hücrelerdeki problemler için kesin çözüme ulaşmak üzere Karma Tamsayılı Doğrusal Olmayan (MINLP) yapıda olan iki alternatif model geliştirilmiştir. Bu modellerin çözümü 2 makineli sistemlerde dahi uzun zaman almaktadır. Bu nedenle, problem versiyonları için yapıldığı şekilde alternatif olarak ikinci derece konik (MISOCP) bir model geliştirilmiştir. Çeşitli testler yapılarak modeller doğrulanmış ve performansları test edilmiştir. Her ne kadar MISOCP, çözüm süresini önemli derecede iyileştirmiş olsa da problem boyutu küçük bir miktar artırıldığında dahi makul sürelerde çözüm elde edilememektedir. Bu sebeple kısa sürede kaliteli çözümler elde edebilmek için sezgisel bir algoritma geliştirilmiştir. Bu algoritma parça sıralamasını belirleyen bir Genetik Algoritma (GA) ile belirlenmiş parça sıralamasına karşılık gelen en iyi robot hareket sırasını (RHS) ve robot hareket hızlarını belirlemek üzere bir Tabu Arama (TA) algoritmasından oluşan hibrit bir metasezgiseldir.

Geliştirilen matematiksel modellerin ve sezgisel algoritmanın performansları yapılan deneysel çalışmalarla test edilmiştir. Ayrıca, hız kontrolü yapılan sistemlerle, hızın sabit

olduğu sistemler birbirleriyle karşılaştırılarak hız kontrolünün enerji tasarrufu açısından katkısı sayısallaştırılmıştır.

5.2 Literatür Taraması

Modern üretim sistemlerinin kullanımının artması ile robotik hücrelerin etkin kullanımı amacıyla 1990'larda başlayan çalışmalar gün geçtikçe artmaktadır. Robotik Hücre Çizelgeleme literatüründe yapılan çalışmalarda birim ürün üretim zamanını en küçükmek dolayısıyla çıktı miktarını en büyükmek üzere robot hareketlerinin ve üretilecek parçaların çizelgelenmesi amaçlanmaktadır. Literatürdeki birçok çalışma tek tip parçanın üretildiği sistemleri ele almışlardır. Dolayısıyla, bu tür sistemlerde, parça çizelgeleme problemi ortadan kalkarak tek problem olarak robot hareketlerinin çizelgelenmesi problemine dönüşmektedir. Bununla beraber literatürdeki tamamına yakın çalışmalarda tekrar edilebilir hareketler dizisi içerisinden en iyi robot hareket döngüsünün bulunmaya çalışıldığı (döngüsel çizelgeleme) görülmektedir.

Literatürdeki çalışmalar ilk olarak üretim sisteminin en basit haliyle ele alındığı yaklaşım olan robotun tek tutucusunun bulunduğu ve bu tutucuda tek bir parçanın taşınabildiği (tek tutuculu) robotik hücreler üstünde yoğunlaşmıştır. Bu tip robotik hücrelerde tek tip parça üretimi ele alınsa da rekabetin artmasıyla gelişen teknoloji ve müşteri istekleri doğrultusunda farklı türde parçaların üretildiği robotun ve üretim hücresinin değişik yapılar ve teknolojik özelliklere sahip olduğu çok çeşitli problemler de çalışılmıştır. Robotik hücre çizelgeleme problemleri üretilecek parça tipi, makine sayısı, robot sayısı ve kaç tutuculu olduğu, makineler arası stok alanı gibi özelliklere bağlı olarak değişiklik göstermektedir. Bahsedilecek olan çalışmalar bu faktörlerin değişik varsayımlarından oluşmaktadır.

Amaç fonksiyonu açısından ele alındığı zaman, bir çok çalışmada birim çevrim süresini (cycle-time) enküçükmek amaçlanmaktadır. Yayılma zamanı (makespan) en küçükleme amacını ele alan az sayıda çalışma da bulunmaktadır. Robotik hücre çizelgelenmesi önceki bölümlerde de bahsedildiği gibi Sethi vd. [20] tarafından yapılan çalışma ile literatürde yerini almıştır. Bu çalışmada, m makineli sistem için $m!$ adet 1-birim döngüsü bulunduğu gösterilmiştir. Ayrıca iki makineli robotik hücreler için 1-birimlik döngülerden birinin optimal olacağı ispatlanmış ve 3 makineli sistem için de optimal olabileceği gösterilmiştir. Crama vd. [35] tarafından yapılan çalışmada, 3 makine için 1-birim döngülerin optimal olduğu ispatlanmış ve m -makineli robotik hücre için en iyi 1-birim döngüyü veren bir dinamik programlama algoritması geliştirilmiştir. Brauner ve Finke [45] ve Brauner ve Finke [47], $m \geq 4$ hücreler için n -birim döngülerin 1-birim döngülerden daha iyi sonuçlar verebileceğini göstermişlerdir. Dawande vd.[36] herhangi iki makine arasındaki transfer zamanının sabit olarak kabul edildiği robotik hücre çizelgeleme problemini ele almıştır. Bu varsayım altında optimal 1-birim döngüsünü bulan bir algoritma geliştirilmiştir.

Sethi vd. [20] robotik hücre çizelgelenmesinde birçok konuyu ilk ele alan çalışma olmuştur. Değişik tip parça üretimi de ilk defa bu çalışmada yer almıştır. 2 ve 3 makineli sistemleri ele alan bu çalışmada problemin karmaşıklığı üzerinde çalışmalar yapılmıştır. Robot hareket çevrimi sabitlenip ve her bir parça için talep değerleri verildiğinde, En Küçük Parça Kümesi (MPS:Minimal Part Set)'ndeki optimal parça sıralamasını belirleyen polinom zamanlı bir algoritma sunulmuştur. Üretilecek parçalar literatürde en küçük parça kümesi adı verilen

yaklaşım ile belirlenmektedirler. Örneğin, talep miktarları parça A için 40, parça B için 35, parça C için 25 olarak verilmiş ise, buna karşılık gelen en küçük parça kümesi A'dan 8, B'den 7 ve C'den de 5 adet olmak üzere toplam 20 parçadan oluşmaktadır. Dolayısıyla, bu örnek için $n = 20$ 'dir ve bu 20 parça döngüsel olarak tekrar tekrar üretilecektir. Bilge ve Ulusoy [107] m -makinelikli esnek bir üretim sistemi için otomatik güdümlü araçların çizelgelemesi hakkında çalışmışlardır. Sriskandarajah vd.[51] m -makinelikli bir robotik hücrede verilen robot hareket sıralaması için parça sıralaması problemini karmaşıklık sınıflarına ayırmışlardır. Bu sınıflandırmanın sonucunda olası $m!$ tane 1-birim döngü için parça sıralaması probleminin, $(2m - 2)$ tanesinin polinom zamanlı çözülebildiği ve geriye kalan problemlerin NP-tam oldukları belirtilmiştir. Aneja ve Kamoun [108], 2 makinelikli sistemler için robot hareket ve parça üretim sıralamasını Gezgin Satıcı Problemi'nin (Travelling Salesman Problem:TSP) özel bir hali olarak bulan $O(n \log n)$ karmaşıklığına sahip bir algoritma geliştirmişlerdir. 3 makinelikli bir sistem için problemin karmaşıklığını inceleyen Hall vd. [49] ise robot hareket çevrimi ile parça sıralaması problemini eş zamanlı bulan $O(n^4)$ karmaşıklığına sahip bir algoritma önermişlerdir. Çalışmada ayrıca 3 makinelikli bir robotik hücrede verilen bir döngü için (1-birim) parça sıralamasının belirlenmesi problemi incelenmiş ve robot çevrimi sabitlenmiş bile parça sıralamasının bulunması probleminin olası 6 çevrimin 4'ü için NP-Zor olduğu ispatlanmıştır. Hurink ve Knust [109] bir adet robotun kullanıldığı akış tipi üretim sistemleri için çizelgeleme problemlerinin NP-zor olduğunu ispatlamış ve tabu arama algoritması önermiştir. Soukhal ve Martineau [110] değişik tip parçaların üretildiği sistemler için tam sayılı bir model ve parça sıralamasını belirlemek amacıyla genetik algoritma geliştirmişlerdir. Carlier vd. [111] aynı problem için makinelerdeki bekleme zamanını (blocked time) da dikkate alarak yayılma zamanını en küçükleyen bir ayrışma algoritması geliştirmişlerdir. İlk olarak parça sıralamasını belirleyip, daha sonra bu parça sıralamasını kullanarak robot hareket sıralamasını belirleyen bir yöntem geliştirmişlerdir. Ayrıca bu çalışmada beş farklı altı sınıra geliştirilmiş ve algoritma performansı bu altı sınıra değerleri kullanılarak karşılaştırılmıştır. Zahrouni ve Kamoun [112] değişik tip parçaların üretildiği 3 makinelikli bir akış tipi üretim sistemi için NEH ([113]) algoritmasından ilham alarak bir çözüm kurucu algoritma geliştirmişlerdir. Batur vd. [114] tarafından yapılan çalışmada, 3 farklı ürünün üretildiği, 2 makinenin olduğu bir sistemde üretim esnekliğinin etkisi gösterilmiştir. Gezgin Satıcı Probleminin bir varyasyonu olan bir formülasyon geliştirilmiştir. Esnekliğin avantajı olarak tüm makinelerin, tüm işlemleri yapabildiği varsayılmıştır. b-Bu yüzden toplam işlem zamanları makinalara bölüştürülebilmektedir. İlk durumda her parça işlem süresi bir makinede tamamlamak zorundayken ikinci durumda birinci parça için iki ayrı makineye de işlem süresi dağıtılmıştır. İkinci durumda, robot fazladan hareket etmek zorunda olsa da makinelerdeki bekleme zamanını azalttığı için makineler daha etkin kullanılmıştır. Sonuçta ikinci durumda ilk duruma göre döngü zamanı %17 azalmıştır. Bu sonuç ile beraber üretimde esnekliğin döngü zamanına etkisi gösterilmiştir.

Son yıllarda yapılan çalışmalarda ise, Fazel Zarandi vd. [115] makine ayar zamanları ve boşaltma/yükleme zamanlarını dikkate alarak 2 makinelikli bir sistem ele almışlardır. Verilen parça üretim sıralamasını kullanarak optimal robot hareket sıralamasını bulan Karma Tam Sayılı Doğrusal bir model geliştirmişlerdir. Ek olarak, büyük çaplı problemlerin çözümü için dal sınır algoritması ve tavlama benzetimi algoritması önermişlerdir. Son olarak, Elmi ve Topaloglu [116]

üretim sistemini etapların oluşturduğu ve her bir etapta m adet makine bulunan bir problem ele almışlardır. Yine transferden bir robotun sorumlu olduğu ve değişik tip parçaların üretildiği bir sistem için Karma Tam Sayılı doğrusal bir model geliştirmişlerdir. Büyük çaplı problemlerin çözümü için tavlama benzetimi algoritması önermişlerdir. Gultekin vd. [117], m -makineli değişik tip parça üreten bir robotik hücre için parça sırlaması ve robot hareket sıralamasını beraberce belirlemek üzere bir matematiksel programlama formülasyonu geliştirmişlerdir. Bu formülasyon, literatürde genellikle yapıldığı gibi problemi sadece 1-birim döngülere indirgememekte ve genel n -birim döngüleri ele almaktadır. Çalışmada ayrıca, problemin çözümü için GA'nin ve TA'nın entegre edildiği hibrit bir metasezgisel yöntem geliştirmişlerdir. Bu çalışmada ele alınan problemin mevcut projeden farkı, hızların kontrol edilebilir değil, sabit olmasıdır. İlerleyen bölümlerde de görüleceği üzere, hızların kontrol edilebilir olması ve enerji tüketimi ve çevrim zamanı olmak üzere iki farklı amacın kullanılması, mevcut problemi önecekine göre oldukça karmaşıklştırmaktadır. Bunun yanında, önceki çalışmada geliştirilmiş olan hibrit metasezgisel yöntem, mevcut projede Pareto önyüz üzerindeki bazı noktaların türetilmesinden kullanılmıştır.

Yukarıda bahsedilen bütün çalışmalarda robot hareket süreleri ve makine işlem süreleri birer problem parametresidir ve önceden bilinmektedir. Mevcut projede ise bu süreler birer karar değişkenidir. Ayrıca bahsedilen çalışmalarda tek amaç çevrim zamanının enküçüklenmesi iken mevcut projede bu amacın yanında toplam harcanan enerjinin minimizasyonu da beraberce ele alınmaktadır. Dolayısıyla önceden de bahsedildiği gibi 2-kriterli bir optimizasyon problemi çözdürülecektir.

Mevcut proje ile ilişkisi en fazla olan çalışmalar Gultekin vd. [85] ve Gultekin vd. [86] yılı çalışmalarıdır. Bu çalışmalarda robotik hücrede kullanılan CNC makinelerinin hızlarının belirli maliyetlere katlanılarak artırılıp azaltılabileceği (dolayısıyla işlem zamanlarının artırılıp azaltılabileceği) varsayımı altında, 2 ve 3 makineli akış atölyeleri için hem çevrim zamanını ve hem de üretim maliyetlerini en küçükleyen iki-kriterli modeller ele alınmıştır. Bunlardan ilkinde hangi işlemin hangi makinede işleneceği belirliyen ikincisinde ilave olarak işlemlerin makinelere atanması problemi de ele alınmıştır. Fakat, robot hareket hızlarının sabit varsayılması ve maliyete etkisinin olmadığı varsayımı, sadece 2 ve 3 makineli sistemlerin ele alınmış olması, genel m -makineli sistemlerin ele alınmamış olması ve bu sistemler için genel bir matematiksel modelleme formülasyonu geliştirilmemiş olması, ayrıca sadece tek tip parça üreten tek tutuculu sistemlerin ele alınmış olması, mevcut problem kapsamında ise buna ilave olarak farklı tip parça üretiminin de ele alınacak olması problemin önemli bir farklılığıdır.

Amaç fonksiyonunun yayılma zamanının en küçüklenmesi olduğu 2-makineli problem Kise [118] tarafından incelenmiştir. Bu çalışmada ilgili problem için literatürde bilinen Gilmore-Gomory algoritmasına dayanan bir algoritma önerilmiştir. Aynı problemde eğer makineler arası robot transfer zamanı iş bağımlı ise asimetrik gezgin satıcı problemine benzer ve NP-zordur ([119]). Kamalabadi vd. [120] farklı tip parça üreten 3 makineli robotik hücreleri ele almışlar ve parçacık sürü optimizasyonu dayalı bir algoritma geliştirilmiştir. Fathian vd. [121] farklı tip parça üreten 2-makineli robotik hücreler için petri ağlarından yola çıkarak bir matematiksel model geliştirmişlerdir. Sethi vd. [93] çalışmalarında 2-makineli sistemlerde parametrelerin belirli şartları sağlaması durumunda en iyi 1-birim döngüsünü belirlemişler, m -makineli sistemlerde de çevrim zamanı için bir alt sınır değeri elde etmişlerdir. Daha maliyetli olan çift tutuculu robotik

hücrelerin, tek tutuculu robotik hücrelere göre avantajlarını incelemişler ve hücre parametreleri ve makine sayısı verildiğinde çift tutuculu robot ile tek tutuculu robot performansını karşılaştıran basit bir sezgisel geliştirmişlerdir.

Bu çalışmalarda da robot hareket süreleri ve makine işlem süreleri birer problem parametresidir. Mevcut projede ise bu süreler birer karar değişkenidir. Enerji minimizasyonu problemi bu çalışmalarda ele alınmamıştır.

Sonraki bölümde, ele alınan bu problem için proje kapsamında geliştirilmiş olan matematiksel modeller açıklanacaktır. Bu modeller, Pareto etkin çözümler türetmek amacıyla epsilon-kısıt yaklaşımı kullanılarak oluşturulmuştur. Buna göre, çevrim zamanı amacı, bir üst sınır eklenerek kısıta dönüştürülmüş ve problem toplam enerji tüketimini enküçükmek üzere tek amaç olarak çözdürülmüştür. Çevrim zamanı üzerine farklı üst limit değerleri verilerek farklı etkin çözümler türetilmesi sağlanacaktır.

5.3 Matematiksel Modeller

Bu bölümde m -makinelili bir robotik hücredeki en iyi n -birim döngüsünü, parça üretim sıralamasını ve robot hareket hızlarını eş zamanlı olarak belirleyen alternatif matematiksel modeller geliştirilecektir.

Matematiksel Model 1 (MINLP1)

Bu modelde kullanılacak notasyon şu şekildedir:

Kümeler:

N = Parça tipi kümesi $\{1, 2, \dots, n\}$

M = Makineler kümesi $\{0, 1, \dots, (m + 1)\}$

A = Aktiviteler kümesi $\{0, 1, \dots, m\}$

P = Pozisyon kümesi $\{1, 2, \dots, n.(m + 1)\}$

T = Tekrar sayısı $\{1, 2, \dots, n\}$

Parametreler:

ε = Makine yükleme boşaltma süresi

$d_{j,h}$ = Makineler arası mesafe

a, b = Enerji tüketim fonksiyonu hesabı için üssel kuvvet olarak kullanılan sayılar

C_e = Boş hıza ait enerji tüketim fonksiyonu katsayısı

C_f = Dolu hıza ait enerji tüketim fonksiyonu katsayısı

P_m^n = n parçasının m makinesindeki işlem süresi

\overline{CT} = Çevrim zamanı için üst sınır

lb/ub = Robot hızları için alt/üst sınır

G = İki makine arası boş hareket süresinin alt sınır değerinden daha küçük bir sayı

M = Büyük bir sayı.

Karar Degiskenleri:

$$x_{i,j}^p : \begin{cases} 1, & \text{Eğer } p \text{ pozisyonunda } i \text{ aktivitesi } j. \text{ kez yapılıyorsa} \\ 0, & \text{diğer durumlarda} \end{cases} \quad (i \in A, k \in P, j \in T)$$

$$y_{i,j}^n : \begin{cases} 1, & \text{Eğer } i \text{ aktivitesinin } j. \text{ tekrarı } n \text{ tipi parçaya ait ise} \\ 0, & \text{diğer durumlarda} \end{cases} \quad (i \in A, n \in N, j \in T)$$

$$z_{i,j,p}^n = x_{i,j}^p y_{i,j}^n : \begin{cases} 1, & \text{Eğer } i \text{ aktivitesinin } n \text{ parçasına ait } j. \text{ tekrarı } p \text{ pozisyonunda ise} \\ 0, & \text{diğer durumlarda} \end{cases} \quad (i \in A, p \in P, j \in T, n \in N)$$

ϑ_p^f : Robotun p pozisyonunda sahip olduğu dolu hız ($p \in P$)

ϑ_p^e : Robotun p pozisyonunda sahip olduğu boş hız ($p \in P$)

δ_p : p pozisyonundaki alınan boş süre ($p \in P$)

d_p^e : p pozisyonundaki boş hızla alınan mesafe ($p \in P$)

d_p^f : p pozisyonundaki dolu hızla alınan mesafe ($p \in P$)

T_p : p pozisyonundaki aktivitenin başlama süresi ($p \in P$)

CT : Çevrim zamanı

$$\text{Min } C_f \sum_{p \in P} d_p^f (v_p^f)^a + C_e \sum_{p \in P} d_p^e (v_p^e)^b$$

Öyle ki:

$$\sum_{p \in P} x_{i,j}^p = 1 \quad \forall i \in A, \quad \forall j \in T \quad (118)$$

$$\sum_{j \in T} \sum_{i \in A} x_{i,j}^p = 1 \quad \forall p \in P \quad (119)$$

$$\sum_{j \in T} y_{i,j}^n = 1 \quad \forall i \in A, \forall n \in N \quad (120)$$

$$\sum_{n \in N} y_{i,j}^n = 1 \quad \forall i \in A, \forall j \in T \quad (121)$$

$$z_{i,j,p}^n \leq x_{i,j}^p \quad \forall i \in A, \forall p \in P, \forall j \in T, \forall n \in N \quad (122)$$

$$z_{i,j,p}^n \leq y_{i,j}^n \quad \forall i \in A, \forall p \in P, \forall j \in T, \forall n \in N \quad (123)$$

$$z_{i,j,p}^n \geq x_{i,j}^p + y_{i,j}^n - 1 \quad \forall i \in A, \forall p \in P, \forall j \in T, \forall n \in N \quad (124)$$

$$\sum_{n \in N} z_{0,1,1}^n = 1 \quad (125)$$

$$T_1 = 0 \quad (126)$$

$$\sum_{p \in P} p \cdot x_{i,j}^p \leq \sum_{p \in P} p \cdot x_{i,l}^p \quad \forall i \in A, \forall j, l \in T, j < l \quad (127)$$

$$\sum_{j \in T} \sum_{\substack{p \in P: \\ h < p < s}} x_{i-1,j}^p \geq x_{i,l}^h + x_{i,l+1}^s - 1 \quad \forall k, h \in P : k < h \quad \forall i \in A \setminus \{0\}, \forall l \in T \setminus \{n\} \quad (128)$$

$$\sum_{j \in T} \sum_{\substack{p \in P: \\ h < p < s}} x_{i+1,j}^p \geq x_{i,l}^h + x_{i,l+1}^s - 1 \quad \forall k, h \in P : k < h \quad \forall i \in A \setminus \{m\}, \forall l \in T \setminus \{n\} \quad (129)$$

$$\sum_{j \in T} \sum_{\substack{p \in P: \\ h < p < s}} z_{i+1,j,p}^n \geq \sum_{j \in T} z_{i,j,h}^n + \sum_{\substack{j \in T \\ t \neq n}} \sum_{t \in N} z_{i+1,j,s}^t - 1 \quad \forall i \in A \setminus \{m\}, \forall n \in N, \forall s, h \in P : h < s \quad (130)$$

$$T_k \geq T_p + 2\varepsilon + \frac{d_k^f}{v_k} \cdot \sum_{j \in T} x_{i-1,j}^p + \sum_{j \in T} \sum_{n \in N} z_{i,j,k}^n \cdot P_i^n \quad \forall i \in A \setminus \{0\}, \quad \forall p, k \in P : p < k \quad (131)$$

$$CT + T_k \geq T_p + 2\varepsilon + \frac{d_k^f}{v_k} \cdot \sum_{j \in T} x_{i-1,j}^p + \sum_{j \in T} \sum_{n \in N} z_{i,j,k}^n \cdot P_i^n \quad \forall i \in A \setminus \{0\}, \quad \forall p, k \in P : k < p \quad (132)$$

$$T_{p+1} \geq T_p + 2\varepsilon + \frac{d_p^f}{v_p} + \delta_p \quad \forall i, q \in A, \forall p \in P \setminus \{n(m+1)\} \quad (133)$$

$$CT \geq T_{n(m+1)} + 2\varepsilon + \frac{d_{n(m+1)}^f}{v_{n(m+1)}^f} + \delta_{n(m+1)} \sum_{j \in T} x_{i,j}^{n(m+1)} \quad \forall i \in A \quad (134)$$

$$\delta_p \leq G + M \left(2 - \sum_{j \in T} x_{i,j}^p - \sum_{j \in T} x_{i+1,j}^{p+1} \right) \quad \forall i \in A \setminus \{m\}, \forall p \in P \setminus \{n(m+1)\} \quad (135)$$

$$d_p^f = \sum_{i \in A} d_{i,i+1} \cdot \sum_{j \in T} x_{i,j,p} \quad \forall p \in P \quad (136)$$

$$d_p^e \geq d_{i+1,q} \left(\sum_{j \in T} x_{i,j}^p + \sum_{j \in T} x_{q,j}^{p+1} - 1 \right) \quad \forall q, i \in A \quad \forall p \in P \setminus \{n(m+1)\} \quad (137)$$

$$d_{n(m+1)}^e \geq \sum_{i \in A} d_{i+1,0} \cdot x_{i,T,n(m+1)} \quad (138)$$

$$v_p^e \leq ub \left(2 - \sum_{j \in T} x_{i,j}^p - \sum_{j \in T} x_{i+1,j}^{p+1} \right) \quad \forall i \in A \setminus \{m\}, \forall p \in P \setminus \{n(m+1)\} \quad (139)$$

$$v_p^e \geq lb \left(\sum_{j \in T} x_{i,j}^p - \sum_{j \in T} x_{q,j}^{p+1} \right) \quad \forall i, q \in A : q \neq i+1, \forall p \in P \setminus \{n(m+1)\} \quad (140)$$

$$v_p^e \leq ub \sum_{i \in A} x_{i,T}^{n(m+1)} \quad (141)$$

$$v_p^e \geq lb \sum_{i \in A} x_{i,T}^{n(m+1)} \quad (142)$$

$$v_p^e \geq \frac{d_p^e}{\delta_p} \quad \forall p \in P \quad (143)$$

$$\overline{CT} \geq CT \quad (144)$$

$$\delta_p \geq G \quad \forall p \in P \quad (145)$$

$$v_p^f \geq lb \quad \forall p \in P \quad (146)$$

$$v_p^f \leq ub \quad \forall p \in P \quad (147)$$

$$v_p^e \leq ub \quad \forall p \in P \quad (148)$$

$$v_p^e \geq 0 \quad \forall p \in P \quad (149)$$

$$T_p \geq 0 \quad \forall p \in P \quad (150)$$

$$CT \geq 0 \quad (151)$$

$$x_{i,j}^p, y_{i,j}^n, z_{i,j,p}^n \in \{0, 1\} \quad \forall i, q \in A, \forall p \in P, \forall n \in N, \forall j \in T \quad (152)$$

Bu modelde amaç fonksiyonu, bütün döngüler tamamlandığında yani mevcut parça tipleri üretildiğinde robotun harcamış olduğu toplam enerji miktarının en küçüklenmesini sağlamaktadır. Toplam enerji tüketim fonksiyonu olarak adlandırılan bu ifade robotun boş/dolu olarak sahip olduğu hızlardan ve katettiği mesafelerden oluşmaktadır. Ayrıca boş veya dolu olmasına göre sahip olduğu katsayılar mevcuttur. Kısıt (118) ile herhangi bir aktivitenin herhangi bir tekrarının yalnızca 1 pozisyona atanması şartı sağlanmaktadır. Her pozisyona bir aktivitenin bir tekrarının atanması gerektiği Kısıt (119) ile ifade edilmektedir. Üretilen her bir parçaya ait her aktivite bir defa tekrar edilmeli şartı Kısıt (120) ile; her bir aktivitenin her bir tekrarının yalnızca 1 parçaya ait olması Kısıt (121) ile sağlanmaktadır. (122),(123) ve (124) numaralı kısıtlarla ($z_{i,j,p}^n = x_{i,j}^p \cdot y_{i,j}^n$) ifadesinin doğrusallaştırılması sağlanmaktadır. Ele alınan modelde yapılan varsayımlardan biri de her bir döngünün başlangıç noktasının A_0 aktivitesi olduğu, yani giriş stoğundan yeni bir parça alınarak başladığı varsayılmıştır. Her döngüde bu aktivite en az bir defa yapılacağı için bu varsayım genellikle birşey kaybettirmemektedir. Kısıt (125) bu varsayımı sağlamaktadır. Diğer bir varsayım olarak ele alınan ilk pozisyona ait başlangıç zamanının 0 olması durumu Kısıt (126) ile ifade edilmektedir. Kısıt (127) ile herhangi bir aktivitenin daha önceki bir tekrarının daha sonraki bir tekrarından daha önceki bir pozisyona atanma şartı sağlanır. Yani aktivitelerin tekrarlarının sıralı olması sağlanır. Bir döngünün olurlu olabilmesi için A_i aktivitesinin herhangi iki tekrarı arasında mutlaka bir A_{i-1} ve bir A_{i+1} aktiviteleri yer almalıdır. Bu olurluluk şartı ise Kısıt (128) ve Kısıt (129) yardımı

ile sağlanmaktadır. n tipi parçaya ait A_i aktivitesinden sonra gelen A_{i+1} aktivitesinin de n parçasına ait olması gerekmektedir. Bu olurluluk şartı Kısıt (130) ile sağlanmaktadır.

İki ardışık aktivitenin (A_{i-1} ve A_i) başlangıç zamanları arasında yükleme/boşaltma zamanı (2ϵ), robotun ardışık iki makine arasında dolu yükü katettiği süre ve i makinesine atanan parçanın işlem zamanı kadar sürenin geçmesi gerektiği şartı (131) nolu kısıt ile sağlanmaktadır. Eğer A_{i-1} aktivitesi A_i aktivitesinden daha sonra gerçekleşiyse döngü A_i aktivitesi yapılmadan sona erecektir. Bu durumda başlangıç zamanları arasında bir önceki kısıta ilave olarak döngü zamanı CT kadar daha zaman geçmesi gerekmektedir. Bu durum (132) numaralı kısıt ile sağlanır. Kısıt (133), ardışık pozisyonlara atanan aktivitelerin başlama zamanları arasındaki ilişkiyi ifade etmektedir. Burada eğer ardışık pozisyonlara atanan aktiviteler ardışık değilse robotun boş olarak hareket ettiği süre de (δ_p) hesaba katılmaktadır. Aksi durum için bu değer 0 olmaktadır. Döngü zamanı gerçekleştirilen son aktivitenin tamamlanma zamanından yani son pozisyonun başlangıç zamanından büyük olmalıdır. Tamamlanma zamanına parçayı söz konusu makineye yükledikten sonra giriş stoğuna dönüş zamanı da eklenir ve bu şart da Kısıt (134) ile sağlanmaktadır. Boş olarak geçirilen sürenin (δ_p) belirlenmesinde (135) numaralı kısıt rol almaktadır. Açıklamak gerekirse, eğer ardışık pozisyonlara ait aktiviteler de ardışık robotun bu aktiviteler arasında boş hareket etmesi söz konusu olmamaktadır. Ancak burada 0 yerine iki makine arası boş hareket süresinin alt sınır değerinden daha küçük bir sayı olan G sabitinin kabul edilmesi ileride de bahsedilecek olan boş hıza ait değeri olumsuzluktan kurtarmak amacıyla. Robotun dolu ve boş olarak katettiği mesafe (136), (137) ve (138) numaralı kısıtlarla belirlenmektedir. Boş hıza ait alt ve üst sınır değerleri (139), (140), (141) ve (142) numaralı kısıtlarla ifade edilmektedir. Önceden de bahsedildiği gibi robotun ardışık pozisyonlar arasında boş olarak hareket edebilmesi için bu pozisyonlara atanan aktivite çiftinin ardışık olmaması ile mümkün olmaktadır. Dolayısıyla alt sınır değerinin gerektiğinde 0 değerini alabilmesi de sağlanmış olmaktadır. Kısıt (143) ile robot döngüsünün son pozisyonundaki aktivite yapıldıktan sonra ilgili boş hız değeri hesaplanmaktadır.

Kısıt (144) ise çevrim zamanının verilen üst sınırdan daha küçük veya eşit olmasını sağlamaktadır. Boş hıza ait alt ve üst sınır değeri dolu hız için de geçerlidir. Ancak dikkat edilmesi gereken dolu hıza ait fark alt sınır değerinin hiç bir zaman 0 değeri alamayacağıdır. Ardışık pozisyonlara atanan hangi aktivite çifti olursa olsun robot mutlaka pozitif bir dolu hıza sahip olacaktır. Çünkü aktivitenin tanımı gereği her zaman bir yük boşaltımı gerçekleşmektedir. Bu durumu sağlayan (146) ve (147) numaralı kısıtlardır. (149), (150), (151) ve (152) numaralı kısıtlar ise değişkenlere ait işaret kısıtlarıdır. Ele alınan bu modelin lineer olmaması sebebiyle GAMS arayüzü kullanılmış ve içerisinde doğrusal olmayan modelleri çözmede kullanılan BARON ve DICOPT çözümleri ile testler yapılmıştır. DICOPT çözümleri optimalite garanti etmemekle birlikte hızlı çözümler türeten bir sezgisel gibi çalışmaktadır. BARON çözümleri ise belirli yapıdaki problemler için optimalite garantisi vermekle birlikte çözüm süreleri küçük örnekler için dahi oldukça uzun sürmektedir.

Burada geliştirilen MINLP1'e alternatif olabilecek ikinci bir matematiksel model aşağıda MINLP2 olarak sunulmuştur.

Matematiksel Model 2 (MINLP2)

Kümeler:

N = Parça tipi kümesi $\{1, 2, \dots, n\}$

M = Makineler kümesi $\{0, 1, \dots, (m + 1)\}$

A = Aktiviteler kümesi $\{0, 1, \dots, m\}$

P = Pozisyon kümesi $\{1, 2, \dots, n.(m + 1)\}$

T = Tekrar sayısı $\{1, 2, \dots, n\}$

Parametreler:

ε = Makine yükleme boşaltma süresi

$d_{j,h}$ = Makineler arası mesafe

a, b = Enerji tüketim fonksiyonu hesabı için üssel kuvvet olarak kullanılan sayılar

C_e = Boş hıza ait enerji tüketim fonksiyonu katsayısı

C_f = Dolu hıza ait enerji tüketim fonksiyonu katsayısı

P_m^n = Her bir parçanın her bir makinedeki işlem görme süresi

\overline{CT} = Çevrim zamanı için üst sınır

lb/ub = Robot hızları için alt/üst sınır

G = İki makine arası boş hareket süresinin alt sınır değerinden daha küçük bir sayı

M = Büyük bir sayı.

Karar Değişkenleri:

$$x_{i,j}^p : \begin{cases} 1, & \text{Eğer } p \text{ pozisyonunda } i \text{ aktivitesi } j. \text{ kez yapılıyorsa} & (i \in A, k \in P, j \in T) \\ 0, & \text{diğer durumlarda} \end{cases}$$

$$\alpha_{i,q}^p = x_{i,j}^p x_{q,j}^{p+1} : \begin{cases} 1, & \text{Eğer } i \text{ ve } q \text{ aktiviteleri sırasıyla } p, p + 1 \text{ pozisyonlarında gerçekleştiriliyorsa} \\ 0, & \text{diğer durumlarda} & (i, q \in A, p \in P \setminus n(m + 1), j \in T) \end{cases}$$

$$y_{i,j}^n : \begin{cases} 1, & \text{Eğer } i \text{ aktivitesinin } j. \text{ tekrarı } n \text{ tipi parçaya ait ise} & (i \in A, n \in N, j \in T) \\ 0, & \text{diğer durumlarda} \end{cases}$$

$$z_{i,j,p}^n = x_{i,j}^p y_{i,j}^n : \begin{cases} 1, & \text{Eğer } i \text{ aktivitesinin } n \text{ parçasına ait } j. \text{ tekrarı } p \text{ pozisyonunda ise} \\ 0, & \text{diğer durumlarda} & (i \in A, p \in P, j \in T, n \in N) \end{cases}$$

ϑ_p^f : Robotun p pozisyonunda sahip olduğu dolu hız ($p \in P$)

ϑ_p^e : Robotun p pozisyonunda sahip olduğu boş hız ($p \in P$)

δ_p : p pozisyonundaki alınan boş süre ($p \in P$)

d_p^e : p pozisyonundaki boş hızla alınan mesafe ($p \in P$)

d_p^f : p pozisyonundaki dolu hızla alınan mesafe ($p \in P$)

T_p : p pozisyonundaki aktivitenin başlama süresi ($p \in P$)

CT : Çevrim zamanı

$$\min C_f \sum_{p \in P} d_p^f (v_p^f)^a + C_e \sum_{p \in P} d_p^e (v_p^e)^b$$

St:

$$\sum_{p \in P} x_{i,j}^p = 1 \quad \forall i \in A, \quad \forall j \in T \quad (153)$$

$$\sum_{j \in T} \sum_{i \in A} x_{i,j}^p = 1 \quad \forall p \in P \quad (154)$$

$$\sum_{j \in T} y_{i,j}^n = 1 \quad \forall i \in A, \forall n \in N \quad (155)$$

$$\sum_{n \in N} y_{i,j}^n = 1 \quad \forall i \in A, \forall j \in T \quad (156)$$

$$z_{i,j,p}^n \leq x_{i,j}^p \quad \forall i \in A, \forall p \in P, \forall j \in T, \forall n \in N \quad (157)$$

$$z_{i,j,p}^n \leq y_{i,j}^n \quad \forall i \in A, \forall p \in P, \forall j \in T, \forall n \in N \quad (158)$$

$$z_{i,j,p}^n \geq x_{i,j}^p + y_{i,j}^n - 1 \quad \forall i \in A, \forall p \in P, \forall j \in T, \forall n \in N \quad (159)$$

$$\sum_{n \in N} z_{0,1,1}^n = 1 \quad (160)$$

$$\sum_{i,q \in A} o_{i,q}^p = 1 \quad \forall p \in P \quad (161)$$

$$o_{i,q,p} \geq \sum_{j \in T} x_{i,j}^p + \sum_{j \in T} x_{q,j}^{p+1} - 1 \quad \forall i, q \in A, \forall p \in P \setminus \{n(m+1)\} \quad (162)$$

$$o_{i,0,n(m+1)} = x_{i,T}^{n(m+1)} \quad \forall i \in A \quad (163)$$

$$T_1 = 0 \quad (164)$$

$$\sum_{p \in P} p \cdot x_{i,j}^p \leq \sum_{p \in P} p \cdot x_{i,l}^p \quad \forall i \in A, \forall j, l \in T, j < l \quad (165)$$

$$\sum_{j \in T} \sum_{\substack{p \in P: \\ h < p < s}} x_{i-1,j}^p \geq x_{i,l}^h + x_{i,l+1}^s - 1 \quad \forall k, h \in P : k < h \quad \forall i \in A \setminus \{0\}, \forall l \in T \setminus \{n\} \quad (166)$$

$$\sum_{j \in T} \sum_{\substack{p \in P: \\ h < p < s}} x_{i+1,j}^p \geq x_{i,l}^h + x_{i,l+1}^s - 1 \quad \forall k, h \in P : k < h \quad \forall i \in A \setminus \{m\}, \forall l \in T \setminus \{n\} \quad (167)$$

$$\sum_{j \in T} \sum_{\substack{p \in P: \\ h < p < s}} z_{i+1,j,p}^n \geq \sum_{j \in T} z_{i,j,h}^n + \sum_{\substack{j \in T \\ t \in N: \\ t \neq n}} \sum_{t \in N} z_{i+1,j,s}^t - 1$$

$$\forall i \in A \setminus \{m\}, \forall n \in N, \forall s, h \in P : h < s \quad (168)$$

$$T_k \geq T_p + 2\varepsilon + \frac{d_k^f}{v_k} \cdot \sum_{j \in T} x_{i-1,j}^p + \sum_{j \in T} \sum_{n \in N} z_{i,j,k}^n \cdot P_i^n \quad \forall i \in A \setminus \{0\}, \quad \forall p, k \in P : p < k \quad (169)$$

$$CT + T_k \geq T_p + 2\varepsilon + \frac{d_k^f}{v_k^f} \cdot \sum_{j \in T} x_{i-1,j}^p + \sum_{j \in T} \sum_{n \in N} z_{i,j,k}^n \cdot P_i^n \quad \forall i \in A \setminus \{0\}, \quad \forall p, k \in P : k < p \quad (170)$$

$$T_{p+1} \geq T_p + 2\varepsilon + \frac{d_p^f}{v_p^f} + \delta_p \quad \forall i, q \in A, \forall p \in P \setminus \{n(m+1)\} \quad (171)$$

$$CT \geq T_{n(m+1)} + 2\varepsilon + \frac{d_{n(m+1)}^f}{v_{n(m+1)}^f} + \delta_{n(m+1)} \sum_{j \in T} x_{i,j}^{n(m+1)} \quad \forall i \in A \quad (172)$$

$$\delta_p \leq G + M \sum_{\substack{i,q \in A: \\ q \neq i+1}} o_{i,q}^p \quad \forall p \in P \setminus \{n(m+1)\} \quad (173)$$

$$d_p^f = \sum_{i \in A} d_{i,i+1} \cdot \sum_{j \in T} x_{i,j,p} \quad \forall p \in P \quad (174)$$

$$d_p^e \geq d_{i+1,q} \left(\sum_{j \in T} x_{i,j}^p + \sum_{j \in T} x_{q,j}^{p+1} - 1 \right) \quad \forall q, i \in A \quad \forall p \in P \setminus \{n(m+1)\} \quad (175)$$

$$d_{n(m+1)}^e \geq \sum_{i \in A} d_{i+1,0} \cdot x_{i,T,n(m+1)} \quad (176)$$

$$v_p^e \leq ub \sum_{\substack{i,q \in A: \\ q \neq i+1}} o_{i,q}^p \quad \forall p \in P \quad (177)$$

$$v_p^e \geq lb \sum_{\substack{i,q \in A: \\ q \neq i+1}} o_{i,q}^p \quad \forall p \in P \quad (178)$$

$$v_p^e \geq \frac{d_p^e}{\delta_p} \quad \forall p \in P \quad (179)$$

$$\overline{CT} \geq CT \quad (180)$$

$$\delta_p \geq G \quad \forall p \in P \quad (181)$$

$$v_p^f \geq lb \quad \forall p \in P \quad (182)$$

$$v_p^f \leq ub \quad \forall p \in P \quad (183)$$

$$v_p^e \leq ub \quad \forall p \in P \quad (184)$$

$$v_p^e \geq 0 \quad \forall p \in P \quad (185)$$

$$T_p \geq 0 \quad \forall p \in P \quad (186)$$

$$CT \geq 0 \quad (187)$$

$$x_{i,j}^p, y_{i,j}^n, z_{i,j,p}^n, o_{i,q}^p \in \{0, 1\} \quad \forall i, q \in A, \forall p \in P, \forall n \in N, \forall j \in T \quad (188)$$

Bu modelde amaç fonksiyonu ve bir çok kısıt önceki modelle aynıdır. Modeli zorlayan bazı kısıtlar ise yeni tanımlanan $o_{i,q}^p$ değişkeni kullanılarak yeniden formüle edilmiştir. Bu kapsamda modelde şu değişiklikler yapılmıştır. Önceki modelde yer alan Kısıt (135) yeni modelde Kısıt (173) şeklinde yeniden yazılmıştır. Benzer şekilde önceki modelde yer alan Kısıtlar (139)-(142) yeni modelde Kısıtlar (177) ve (178) olarak yeniden yazılmıştır. Bunun yanında, modele (161), (162) ve (163) numaralı kısıtlar eklenmiştir. Diğer kısıtların tanımı aynı olmakla beraber değiştirilen ve yeni eklenen kısıtların açıklamaları şu şekildedir: (161), (162) ve (163) numaralı kısıtlar ile $o_{i,q}^n$ değişkenin değeri hesaplanmaktadır. Kısıt (161), ardışık pozisyonlar için bir aktivite çiftinin art arda olmasını sağlamaktadır. (162) numaralı kısıt, herhangi iki aktivitenin arda arda atanmış olması durumunda $o_{i,q}^p$ değişkenin değerini 1 yapmaktadır. Ancak bu

kısıtla $\sigma_{i,q}^p$ değişkeninin değeri son pozisyon haricindeki pozisyonlara atanması durumunda belirlenebildiği için kısıt (163) ile son pozisyona ait değerin belirlenmesi sağlanmaktadır. Boş olarak geçirilen sürenin (δ_p) belirlenmesi Kısıt (173) ile sağlanmaktadır. Buna göre, eğer ardışık pozisyonlara ait aktiviteler de ardışıksa robotun bu aktiviteler arasında boş hareket etmesi söz konusu olmamaktadır. Ancak burada 0 yerine iki makine arası boş hareket süresinin alt sınır değerinden daha küçük bir sayı olan G sabitinin kabul edilmesi ileride de bahsedilecek olan boş hıza ait değeri olumsuzluktan kurtarmak amacıyla. Boş hıza ait alt ve üst sınır değerleri kısıt (177), (178) ile ifade edilmektedir. Robotun ardışık pozisyonlar arasında boş olarak hareket edebilmesi için bu pozisyonlara atanan aktivite çiftinin ardışık olmaması ile mümkün olmaktadır. Dolayısıyla alt sınır değerinin gerektiğinde 0 değerini alabilmesi de sağlanmış olmaktadır. Açıklanan son üç kısıt $\sigma_{i,q}^p$ değişkeni kullanılarak yeniden yazılmıştır.

Geliştirilen bu model de doğrusal olmayan bir yapıdadır. Dolayısıyla, matematiksel modelin çözüm süresinin daha da kısaltılabilmesi ve CPLEX çözücüsü kullanılarak optimal çözümlerin elde edilebilmesi için bu model önceki bölümlerde de açıklanan yöntemler kullanılarak ikinci dereceden konik forma dönüştürülmüştür. Bu model aşağıda açıklanmıştır.

Matematiksel Model-3 (MISOCP)

Kümeler:

N = Parça tipi kümesi $\{1, 2, \dots, n\}$

M = Makineler kümesi $\{0, 1, \dots, (m + 1)\}$

A = Aktiviteler kümesi $\{0, 1, \dots, m\}$

P = Pozisyon kümesi $\{1, 2, \dots, n.(m + 1)\}$

T = Tekrar sayısı $\{1, 2, \dots, n\}$

Parametreler:

ε = Makine yükleme boşaltma süresi

$\sqrt{d_{j,h}}$ = Makineler arası mesafenin karekökü

a, b = Enerji tüketim fonksiyonu hesabı için üssel kuvvet olarak kullanılan sayılar

$\sqrt{C_e}$ = Boş hıza ait enerji tüketim fonksiyonu katsayısı karekökü

$\sqrt{C_f}$ = Dolu hıza ait enerji tüketim fonksiyonu katsayısı karekökü

$P_{m,n}$ = Her bir parçanın her bir makinedeki işlem görme süresi

\overline{CT} = Çevrim zamanı için üst sınır

lb/ub = Robot hızları için alt/üst sınır

G = İki makine arası boş hareket süresinin alt sınır değerinden daha küçük bir sayı

$M =$ Büyük bir sayı.

Karar Değişkenleri:

$$x_{i,j}^p : \begin{cases} 1, & \text{Eğer } p \text{ pozisyonunda } i \text{ aktivitesi } j. \text{ kez yapılıyorsa} \\ 0, & \text{diğer durumlarda} \end{cases} \quad (i \in A, p \in P, j \in T)$$

$$o_{i,q}^p = x_{i,j}^p x_{q,j}^{p+1} : \begin{cases} 1, & \text{Eğer } i, q \text{ aktiviteleri sırasıyla } p, p+1 \text{ pozisyonlarında gerçekleştiriliyorsa} \\ 0, & \text{diğer durumlarda} \end{cases} \quad (i, q \in A, p \in P \setminus n(m+1), j \in T)$$

$$y_{i,j}^n : \begin{cases} 1, & \text{Eğer } i \text{ aktivitesinin } j. \text{ tekrarı } n \text{ tipi parçaya ait ise} \\ 0, & \text{diğer durumlarda} \end{cases} \quad (i \in A, n \in N, j \in T)$$

$$z_{i,j,p}^n = x_{i,j,p} y_{i,j,n} : \begin{cases} 1, & \text{Eğer } i \text{ aktivitesinin } n \text{ parçasına ait } j. \text{ tekrarı } p \text{ pozisyonunda ise} \\ 0, & \text{diğer durumlarda} \end{cases} \quad (i \in A, p \in P, j \in T, n \in N)$$

$\vartheta_{i,q,p}^e$: Robotun p pozisyonundaki i aktivitesini yaptıktan sonra ardından gelen q aktivitesini gerçekleştirmek için sahip olduğu boş hız ($i, q \in A, p \in P$)

$\vartheta_{i,p}^f$: Robotun p pozisyonunda sahip olduğu dolu hız ($i \in A, p \in P$)

$\delta_{i,q,p}^e$: Robotun p pozisyonundaki i aktivitesini yaptıktan sonra ardından gelen q aktivitesini gerçekleştirmek için sahip olduğu boş süre ($i, q \in A, p \in P$)

$\delta_{i,p}^f$: p pozisyonundaki alınan dolu süre ($i \in A, p \in P$)

T_p : p pozisyonundaki aktivitenin başlama süresi ($p \in P$)

Q : Konik kısıtlar için kullanılan yapay karar değişkeni

F_p^f : Robotun p pozisyonundaki aktivite sırasında yaptığı dolu hareketinden kaynaklanan enerji tüketim fonksiyonu ($p \in P$)

F_p^e : Robotun p pozisyonundaki aktivite sırasında yaptığı boş hareketinden kaynaklanan enerji tüketim fonksiyonu ($p \in P$)

CT : Çevrim zamanı

$v_{i,p}^\alpha$: İkinci dereceden konik programlama için kullanılan karar değişkeni ($i \in A, p \in P, \forall \alpha = 1..3$)

$v_{i,q,p}^\beta$: İkinci dereceden konik programlama için kullanılan karar değişkeni ($i, q \in A, p \in P, \forall \beta = 1..3$)

v_p^γ : İkinci dereceden konik programlama için kullanılan karar değişkeni ($p \in P, \forall \gamma = 1..6$)

$$\min \sum_{p \in P} (F_p^f + F_p^e)$$

St:

$$\sum_{p \in P} x_{i,j}^p = 1 \quad \forall i \in A, \forall j \in T \quad (189)$$

$$\sum_{j \in T} \sum_{i \in A} x_{i,j}^p = 1 \quad \forall p \in P \quad (190)$$

$$\sum_{j \in T} y_{i,j}^n = 1 \quad \forall i \in A, \forall n \in N \quad (191)$$

$$\sum_{n \in N} y_{i,j}^n = 1 \quad \forall i \in A, \forall j \in T \quad (192)$$

$$z_{i,j,p}^n \leq x_{i,j}^p \quad \forall i \in A, \forall p \in P, \forall j \in T, \forall n \in N \quad (193)$$

$$z_{i,j,p}^n \leq y_{i,j}^n \quad \forall i \in A, \forall p \in P, \forall j \in T, \forall n \in N \quad (194)$$

$$z_{i,j,p}^n \geq x_{i,j}^p + y_{i,j}^n - 1 \quad \forall i \in A, \forall p \in P, \forall j \in T, \forall n \in N \quad (195)$$

$$\sum_{n \in N} z_{0,1,1}^n = 1 \quad (196)$$

$$\sum_{i,q \in A} o_{i,q}^p = 1 \quad \forall p \in P \quad (197)$$

$$o_{i,q,p} \geq \sum_{j \in T} x_{i,j}^p + \sum_{j \in T} x_{q,j}^{p+1} - 1 \quad \forall i, q \in A, \forall p \in P \setminus \{n(m+1)\} \quad (198)$$

$$o_{i,0,n(m+1)} = x_{i,T}^{n(m+1)} \quad \forall i \in A \quad (199)$$

$$T_1 = 0 \quad (200)$$

$$\sum_{p \in P} p \cdot x_{i,j}^p \leq \sum_{p \in P} p \cdot x_{i,l}^p \quad \forall i \in A, \forall j, l \in T, j < l \quad (201)$$

$$\sum_{j \in T} \sum_{\substack{p \in P: \\ h < p < s}} x_{i-1,j}^p \geq x_{i,l}^h + x_{i,l+1}^s - 1 \quad \forall k, h \in P : k < h \quad \forall i \in A \setminus \{0\}, \forall l \in T \setminus \{n\} \quad (202)$$

$$\sum_{j \in T} \sum_{\substack{p \in P: \\ h < p < s}} x_{i+1,j}^p \geq x_{i,l}^h + x_{i,l+1}^s - 1 \quad \forall k, h \in P : k < h \quad \forall i \in A \setminus \{m\}, \forall l \in T \setminus \{n\} \quad (203)$$

$$\sum_{j \in T} \sum_{\substack{p \in P: \\ h < p < s}} z_{i+1,j,p}^n \geq \sum_{j \in T} z_{i,j,h}^n + \sum_{\substack{j \in T \\ t \neq n}} \sum_{t \in N} z_{i+1,j,s}^t - 1 \quad \forall i \in A \setminus \{m\}, \forall n \in N, \forall s, h \in P : h < s \quad (204)$$

$$T_k \geq T_p + 2\varepsilon + \sum_{i \in A \setminus \{0\}} \delta_{i-1,p}^f + \sum_{i \in A} \sum_{j \in T} \sum_{n \in N} z_{i,j,k}^n \cdot P_i^n \quad \forall p, k \in P : p < k \quad (205)$$

$$CT + T_k \geq T_p + 2\varepsilon + \sum_{i \in A \setminus \{0\}} \delta_{i-1,p}^f + \sum_{i \in A} \sum_{j \in T} \sum_{n \in N} z_{i,j,k}^n \cdot P_i^n \quad \forall p, k \in P : k < p \quad (206)$$

$$T_{p+1} \geq T_p + 2\varepsilon + \sum_{i \in A} \delta_{i,p}^f + \sum_{i,b \in A} \delta_{i,q,p}^e \quad \forall p \in P \setminus \{n(m+1)\} \quad (207)$$

$$CT \geq T_{n(m+1)} + 2\varepsilon + \sum_{i \in A} \delta_{i,p}^f + \sum_{i \in A} \delta_{i,0,n(m+1)}^e \quad (208)$$

$$\vartheta_{i,p}^f \leq ub \sum_{j \in T} x_{i,j}^p \quad \forall i \in A, \forall p \in P \quad (209)$$

$$\vartheta_{i,p}^f \geq lb \sum_{j \in T} x_{i,j}^p \quad \forall i \in A, \forall p \in P \quad (210)$$

$$\vartheta_{i,q,p}^e \leq ub \cdot o_{i,q}^p \quad \forall i, q \in A : q \neq i+1, \forall p \in P \quad (211)$$

$$\vartheta_{i,q,p}^e \geq lb \cdot o_{i,q}^p \quad \forall i, q \in A : q \neq i+1, \forall p \in P \quad (212)$$

$$\vartheta_{i,i+1,p}^e = 0 \quad \forall i \in A \setminus \{m\}, \forall p \in P \quad (213)$$

$$\delta_{i,i+1,p}^e = 0 \quad \forall i \in A \setminus \{m\}, \forall p \in P \quad (214)$$

$$Q = 1 \quad (215)$$

$$v_{i,p}^1 = \vartheta_{i,p}^f - \delta_{i,p}^f \quad \forall i \in A, \forall p \in P \quad (216)$$

$$v_{i,p}^2 = \vartheta_{i,p}^f + \delta_{i,p}^f \quad \forall i \in A, \forall p \in P \quad (217)$$

$$v_{i,p}^3 = 2 \cdot \sqrt{d_{i,i+1}} \cdot \sum_{j \in T} x_{i,j}^p \quad \forall i \in A, \forall p \in P \quad (218)$$

$$(v_{i,p}^1)^2 + (v_{i,p}^3)^2 \leq (v_{i,p}^2)^2 \quad \forall i \in A, \forall p \in P \quad (219)$$

$$v_{i,q,p}^1 = \vartheta_{i,q,p}^e - \delta_{i,q,p}^e \quad \forall i, q \in A, \forall p \in P \quad (220)$$

$$v_{i,q,p}^2 = \vartheta_{i,q,p}^e + \delta_{i,q,p}^e \quad \forall i, q \in A, \forall p \in P \quad (221)$$

$$v_{i,q,p}^3 = 2 \cdot \sqrt{d_{i+1,q}} \cdot o_{i,q}^p \quad \forall i, q \in A, \forall p \in P \quad (222)$$

$$(v_{i,q,p}^1)^2 + (v_{i,q,p}^3)^2 \leq (v_{i,q,p}^2)^2 \quad \forall i, q \in A, \forall p \in P \quad (223)$$

$$v_p^1 = F_p^f - Q \quad \forall p \in P \quad (224)$$

$$v_p^2 = F_p^f + Q \quad \forall p \in P \quad (225)$$

$$v_p^3 = 2 \cdot \sqrt{C_f} \cdot \sum (i \in A) \sqrt{d_{i,i+1}} \cdot \vartheta_{i,p}^f \quad \forall p \in P \quad (226)$$

$$(v_p^1)^2 + (v_p^3)^2 \leq (v_p^2)^2 \quad \forall p \in P \quad (227)$$

$$v_p^4 = F_p^e - Q \quad \forall p \in P \quad (228)$$

$$v_p^5 = F_p^e + Q \quad \forall p \in P \quad (229)$$

$$v_p^6 = 2 \cdot \sqrt{C_e} \cdot \sum (i, q \in A) \sqrt{d_{i+1,q}} \cdot \vartheta_{i,q,p}^e \quad \forall p \in P \quad (230)$$

$$(v_p^4)^2 + (v_p^6)^2 \leq (v_p^5)^2 \quad \forall p \in P \quad (231)$$

$$\overline{CT} \geq CT \quad (232)$$

$$T_p \geq 0 \quad \forall p \in P \quad (233)$$

$$CT \geq 0 \quad (234)$$

$$x_{i,j}^p, y_{i,j}^n, z_{i,j,p}^n, o_{i,q}^p \in \{0, 1\} \quad \forall i, q \in A, \forall p \in P, \forall n \in N, \forall j \in T \quad (235)$$

$$v_{i,p}^\alpha, v_p^\gamma, v_{i,q,p}^\beta, o_{i,q}^p : urs \quad \forall i, q \in A, \forall p \in P \quad (236)$$

Modelde amaç fonksiyonu için yine bütün döngüler tamamlandığında yani mevcut parça tipleri üretildiğinde robotun harcamış olduğu toplam enerji miktarının en küçüklenmesi hedeflenmektedir. Bir önceki modelden farklı olarak amaç fonksiyonu doğrusal olarak ifade edilmiştir. Açılımı kısıtlarda ifade edilmektedir. Kısıtlar (189)-(204), (232)-(234) Matematiksel Model 2 ile aynıdır. Bu sebeple burada tekrar açıklanmamıştır.

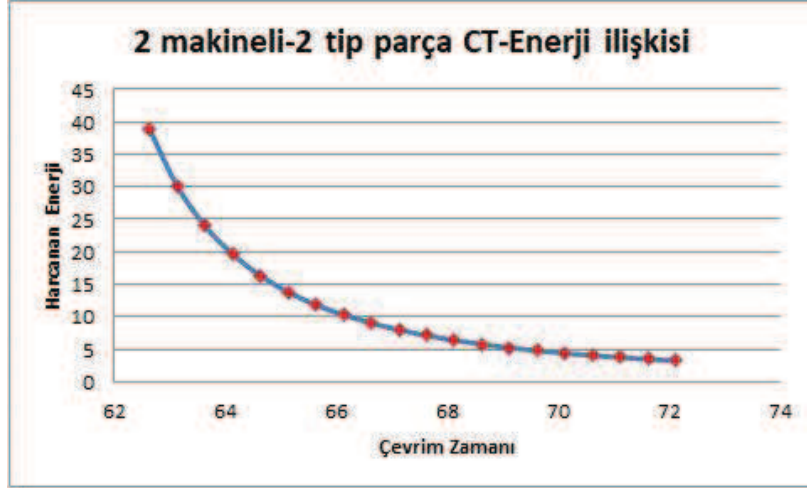
İki ardışık aktivitenin (A_{i-1} ve A_i) başlangıç zamanları arasında yükleme/boşaltma zamanı (2ϵ), robotun ardışık iki makine arasında dolu yükü katettiği süre ve i makinesine atanan parçanın işlem zamanı kadar sürenin geçmesi gerektiği Kısıt (205) ile sağlanmaktadır. Bu kısıt Matematiksel Model 2'de doğrusal olmayan Kısıt (169) yerine yazılmıştır. Benzer şekilde bir önceki modeldeki Kısıtlar (170)-(172) yeni modelde (206)-(208) ile değiştirilmiştir. Bunlardan Kısıt (170), A_{i-1} aktivitesinin A_i aktivitesinden daha sonra gerçekleştiği durumu ele almaktadır. Bu durumda döngü A_i aktivitesi yapılmadan sona erecektir. Dolayısıyla, başlangıç zamanları arasında bir önceki kısıta ilave olarak döngü zamanı (CT) kadar daha zaman geçmesi gerekmektedir. Kısıt (207), ardışık pozisyonlara atanan aktivitelerin başlama zamanları arasındaki ilişkiyi ifade etmektedir. Burada eğer ardışık pozisyonlara atanan aktiviteler ardışık değilse robotun boş olarak hareket ettiği süre de (δ_p) hesaba katılmaktadır. Aksi durum için bu değer 0 olmaktadır. Döngü zamanı gerçekleştirilen son aktivitenin tamamlanma zamanından yani son pozisyonun başlangıç zamanından büyük olmalıdır. Tamamlanma zamanına parçayı söz konusu makineye yükledikten sonra giriş stoğuna dönüş zamanı da eklenir ve bu şart da kısıt (208) ile sağlanmaktadır.

(209) ve (210) numaralı kısıtlar robotun dolu hızının alabileceği alt ve üst sınır değerlerini belirlemektedir. Kısıt (211) ve (212) ile ardışık olmayan aktiviteler arasında robotun boş hızının 0'dan büyük değerler alması gerektiği ifade edilmektedir. (213) ve (214) numaralı kısıtlarda ise önceki belirtilen kısıtların aksi durumları için boş hızın alacağı değer 0 olması sağlanmaktadır.

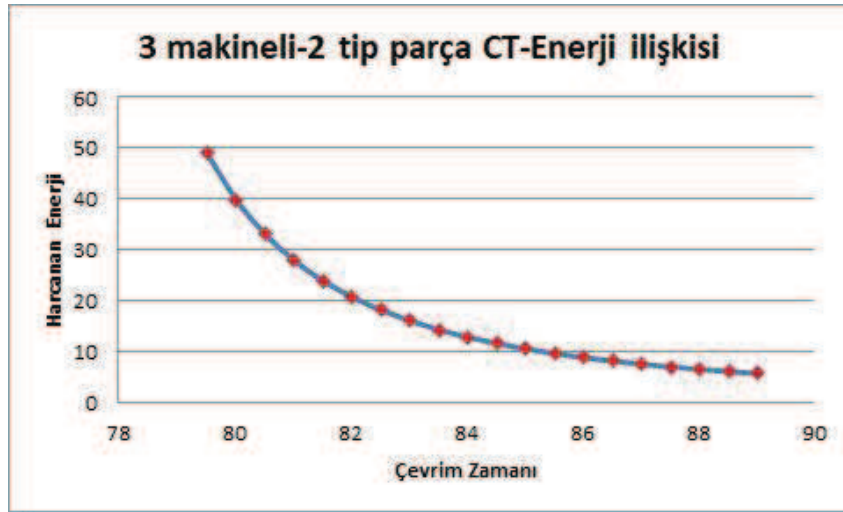
Kısıt (215) ile konik programlama kısıtlarında kullanılacak karar değişkeni Q 'nun değeri 1 olarak atanmıştır. Kısıtlar (216),(217),(218) ve (219) ile dolu hız ve hareket süresi değerleri kullanılarak dolu olarak katedilen toplam mesafe hesaplanmaktadır. Aynı yaklaşım (220),(221),(222) ve (223) numaralı kısıtlarda da kullanılarak boş olarak yapılan harekete ait hız, süre ve mesafe değerleri hesaplanmaktadır. Bu durumlara bağlı olarak tüketilen toplam enerji ise (224)-(231) kısıtları kullanılarak belirlenmektedir. Geriye kalan kısıtlar ise değişkenlere ait işaret kısıtlardır.

Geliştirilen bu model enerji fonksiyonundaki üstel kuvvetin derecesine bağlı olarak önceki bölümde örnekleri verildiği şekilde farklı yöntemler kullanılarak ikinci dereceden konik yapıya dönüştürülebilir.

Geliştirilen bütün modeller çeşitli örnek problemler kullanılarak test edilmiş ve doğrulama ve geçerlemeleri yapılmıştır. Bütün modeller için amaç aslında iki kriterden oluşmaktadır. Bunlardan biri çevrim zamanının diğer ise toplam harcanan enerjinin en küçüklenmesidir. İki aşamalı amaç fonksiyonlarında doğrusal olan amaç fonksiyonu daha kolay kısıtlara aktarılabildiğinden tercih olarak çevrim zamanı kısıtlara bir üst sınır değeri ile sınırlandırılarak yazılır. Bu dönüştürme işlemine epsilon kısıtlı yaklaşım adı verilmektedir. Dolayısıyla elde edilen model belirli bir çevrim zamanı değerini aşmadan sistemde harcanan toplam enerjinin en küçüklendiği bir duruma dönüşmektedir. Burada sistemin daha hızlı şekilde çalışması yani daha kısa sürede tüm işlerin tamamlanması harcanacak enerji miktarını arttırmaktadır. Kısaca çevrim zamanı ve enerji arasında bir ödünleşim söz konusudur. Aradaki ilişkinin daha net anlaşılması için elde edilen bazı sonuçlar Şekil 17 ve 18'de gösterilmektedir.



Şekil 17. 2-Makineli 2-Tip Parça için Çevrim Zamanı ve Enerji Ödünleşimi



Şekil 18. 3-Makineli 2-Tip Parça için Çevrim Zamanı ve Enerji Ödünleşimi

İkinci dereceden konik modelleme yaklaşımının çözüm süresine katkısını anlamak için Model 2 ve MISOCP kullanılarak 2 ve 3 farklı tipte parça üreten 2, 3, ve 4 makineli örnek problemlerde testler yapılmıştır. MISOCP GAMS arayüzü altında CPLEX çözücüsü ile çözdürülmüş, Matematiksel Model 2 ise MINLP yapısında olduğu için BARON ve DICOPT çözücülerine çözülmüştür. Bu çözücülerden BARON belirli şartlar altında optimal çözümü garanti etmekte iken DICOPT çözücüsü hızlı çözümler türeten fakat optimalliği garanti etmeyen bir nevi sezgisel niteliğindedir. Bu testlerden elde edilen sonuçlar Tablo 29 ve Tablo 30'da sunulmuştur. MINLP modelleri 1 saat zaman limiti ile çalıştırılmıştır. Zaman limitine takılarak duran problemler tablolarda ZL ile gösterilmiştir. Bu tablolarda da görülebileceği gibi BARON çözücüsü, çözülen 5 örneğin hiçbirisinde 1 saat içerisinde optimal çözümü bulamamıştır. Bunlardan 4 tanesinde 1 saat içerisinde olurlu biz çözüm dahi bulamamış, olurlu çözüm bulunduğu durumda ise bulabildiği çözümün enerji tüketim değeri 192.4 çıkmıştır. BARON'un

bu durum için belirlediği alt sınır değeri 0.174'tür. Görülebileceği gibi BARON çözücüsü kullanılarak optimal çözüme makul sürelerde ulaşmak, çok küçük problem boyutlarında bile mümkün olamamaktadır. Aynı problem için MISOCP 14.39 gibi bir değer bulmuştur. Diğer taraftan, DICOPT çözücüsü problemler için kısa sürede çözümler türetebilmektedir. Fakat bu çözümlerin enerji tüketim değerleri optimalden sapmaktadır. Örneğin Tablo 29'de sunulan $n = 2$, $m = 2$ durumu için $CT = 65$ verildiğinde MISOCP enerji tüketimini 14.39 bulurken DICOPT çözücüsü 27.90 bulmuştur. %100'e yaklaşan bir sapma söz konusudur. Bu testler MISOCP'un problemin çözüm süresini hızlandırmak anlamında oldukça başarılı olduğunu göstermektedir. Fakat problem boyutu arttıkça bu modelin de çözüm süresi hızla artmaktadır. Dolayısıyla, daha büyük problem durumlarını makul sürelerde çözebilmek için sezgisel yöntemlere ihtiyaç duyulmaktadır. Sonraki bölümde geliştirilen sezgisel algoritma açıklanacaktır.

Tablo 29. $n=2$ için MINLP2 ve MISOCP'e ait Test Sonuçları

m	CT	MISOCP-CPLEX		MINLP2-BARON		MINLP2-DICOPT	
		Enerji	Süre (sn)	Enerji	Süre (sn)	Enerji	Süre (sn)
2	65.00	14.39	0.91	192.4*	ZL	27.90	0.73
3	80.00	40.37	3.68	ÇY	ZL	41.36	1.70
4	103.00	36.17	77.10	ÇY	ZL	57.77	5.74

Tablo 30. $n=3$ için MINLP2 ve MISOCP'e ait Test Sonuçları

m	CT	MISOCP-CPLEX		MINLP2-BARON		MINLP2-DICOPT	
		Enerji	Süre (sn)	Enerji	Süre (sn)	Enerji	Süre (sn)
2	104.00	27.12	6.76	ÇY	ZL	27.36	2.67
3	136.00	53.16	185.36	ÇY	ZL	53.28	12.69

5.4 Hibrit Metasezgisel Algoritma (HMA)

Bir önceki bölümde alınan sonuçlara bakıldığında doğrusal olmayan matematiksel modellerin ikinci dereceden konik programlama kullanılarak yeniden oluşturulmasının daha kısa sürede çözüm alınabilmesi adına bir avantaj olduğu görülmektedir. Ancak karar verilmesi gereken çok sayıda durumun olması küçük boyuttaki problemlerde bile optimal çözüme makul sürede ulaşmayı engellediğinden sezgisel bir algoritma geliştirme ihtiyacı doğmuştur.

Bu aşamada Genetik ve Tabu arama algoritmalarının beraberce kullanıldığı hibrit bir metasezgisel algoritma geliştirilmiştir. Bu yaklaşım Gultekin vd. [117] tarafından hız kontrolü olmayan m -makineli robotlu hücrelerdeki parça sıralaması (PS) ve robot hareket sıralamasını (RHS) belirlemek için geliştirilen sezgisel algoritmaya benzerlik göstermektedir. Bu sebeple öncelikle bu yaklaşımın detaylarından bahsedilecektir. Daha sonra, enerji tüketimini enküçükleme üzere hız belirleme aşamalarının bu sezgisele nasıl entegre edildiği açıklanacaktır.

Gultekin vd. [117] tarafından ele alınan problemde robot hareket hızları sabittir. Dolayısıyla, o problemde hız parametresi bulunmamakta, bunun yerine robotun ardışık makineler arasındaki mesafeyi katetme süresi parametre olarak yer almaktadır. Ayrıca, o problemdeki tek amaç, en düşük çevrim zamanını belirlemektir. Geliştirilen algoritmada PS'nı belirlemek üzere bir GA geliştirilmiştir. Bu GA'da çözümler parça sırasını gösteren kromozomlar şeklinde tutulmuştur. Örnek olarak, 5 parçalı bir sistemde bir kromozom (2-1-5-3-4) şeklinde parça sırasını göstermektedir. GA'nın başlayabilmesi için başlangıç popülasyonu rastgele PS'ler belirlenerek oluşturulmaktadır. Bu problem için uyumluluk (fitness) değeri, çevrim zamanıdır. Bir PS'ye karşılık gelen çevrim zamanının belirlenebilmesi için RHS'nin de bilinmesi gerekmektedir. Bu amaçla, verilen bir PS'ye karşılık gelen en iyi RHS'yi belirlemek üzere bir Tabu Arama (TA) algoritması geliştirilmiştir. Bu algoritmanın detayları aşağıda anlatılacaktır. Popülasyondaki her bir birey için uyumluluk değeri hesaplandıktan sonra, en iyi uyumluluk değerine sahip belirli sayıda elit çözüm doğrudan bir sonraki nesle aktarılmaktadır. Sonraki nesildeki geri kalan çözümleri belirlemek üzere, başlangıç popülasyonundaki bireyler, rulet tekerleği seçim yöntemine göre seçilerek birbirleriyle eşlenmektedir. Bu seçim yönteminde uyumluluk değeri daha iyi olan çözümlere, uyumluluk değeri oranında daha yüksek seçilme olasılığı atanmaktadır. Birbirleriyle eşlenen bireyler çaprazlanarak iki yeni çözüm oluşturulmaktadır. Oluşturulan bu çözümler belirli bir olasılıkla mutasyona uğramakta ve bunun sonucunda sonraki nesle aktarılmaktadır.

Çaprazlama, bireylerin ortadan bölünmesiyle yapılmaktadır. Çocuklarda eksik kalan kısımlar ikinci ebeveyndeki sıralamaya göre tamamlanır. Örneğin, ilk birey {1, 3, 2, 5, 4, 6}, ikinci birey, {1, 4, 3, 2, 6, 5} olsun. İlk bireyin ilk elemanları alınır {1, 3, 2} ve daha sonra burada olmayan elemanlar için ikinci bireydeki eleman sırası dikkate alınarak tamamlanır. Sonuçta, {1, 3, 2, 4, 6, 5} ve {1, 4, 3, 2, 5, 6} bireyleri elde edilmiş olur. Bu bireyler daha sonra belirli bir olasılıkla mutasyona uğratılır. Mutasyon herhangi bir gendeki rastgele değişikliği ifade eder. Eğer bir birey mutasyona uğrayacaksa rastgele belirlenen iki pozisyondaki elemanların yerleri değiştirilir. Örneğin, {1, 3, 2, 4, 6, 5} bireyi mutasyon için belirlenmiş olsun. Rasgele 2 ve 5 numaralı pozisyonlar belirlenmiş olsun. Bu iki pozisyondaki parçaların yerleri değiştirilerek {1, 6, 2, 4, 3, 5} bireyi elde edilir.

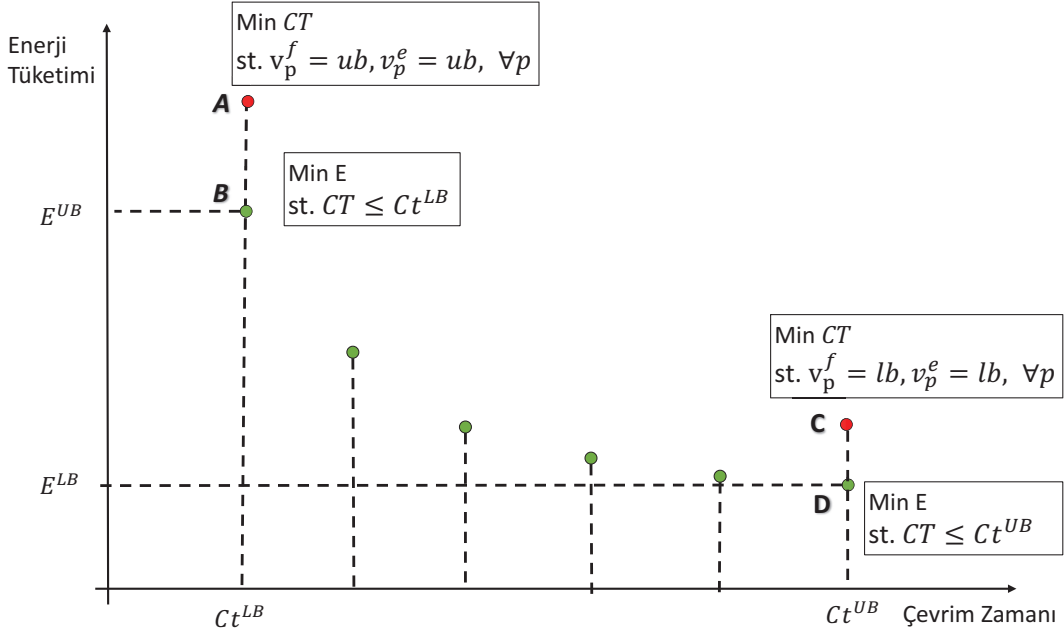
GA, durma kriterine ulaşıncaya kadar çalışmaya devam eder. Durma kriteri, ardışık belirli sayıda iyileşme olmayan iterasyon sayısı olarak belirlenmiştir. GA'nın popülasyon büyüklüğü, elit çözüm yüzdesi, durma kriteri, mutasyon olasılığı gibi parametreleri yapılan hesaplamalı çalışmalarla belirlenmiştir. Gultekin vd. [117]'nin belirlemiş olduğu bu parametreler şu şekildedir: Popülasyon büyüklüğü= $15 \times n$. Burada n problemdeki parça sayısını ifade etmektedir. Elit çözüm sayısı=popülasyon büyüklüğünün yüzde yirmisi. Mutasyon olasılığı=0.05 ve durma kriteri de iyileşme olmayan 75 iterasyon olarak belirlenmiştir.

TA, verilen bir PS'ye karşılık gelen en düşük çevrim zamanını veren RHS'yi belirlemek üzere geliştirilmiştir. TA bir başlangıç çözümüyle başlar ve bu çözümün komşularını tarayarak iyileştirmeye çalışır. Ele alınan problem için bir başlangıç çözümü, olurluluk koşullarını sağlayacak şekilde rasgele bir RHS belirleyerek elde edilmiştir. Verilen herhangi bir RHS'de robotun zaten boş olan bir makineyi boşaltmaya çalışmaması ve zaten dolu olan bir makineyi yüklemeye çalışmaması gerekmektedir. Bu kriterleri sağlamak üzere Crama ve van de Klundert

[21] olurluluk koşullarını şu şekilde belirlemiştir: Herhangi ardışık iki A_i aktivitesi arasında bir tane A_{i-1} ve bir tane A_{i+1} aktivitesi yer almalıdır, $1 \leq i \leq m$. Ardışık A_0 aktiviteleri arasında sadece A_1 ve ardışık A_{m+1} aktiviteleri arasında da sadece A_m aktivitesinin yer alması gerekir. Verilen bir çözümün komşuları türetilirken, her aktivite sırasıyla bulunduğu pozisyonundan kaldırılarak olurluluk şartını bozana kadar tek tek önceki ve arkadaki pozisyonlara yerleştirilmiş ve her yerleştirme yeni bir komşu türetmiştir. Daha sonra türetilen komşuların çevrim zamanı değerleri iteratif bir algoritma kullanılarak hesaplanmış ve içlerinden en iyisi komşu çözüm seçilmiştir. Eğer bu komşu tabu listesinde değilse veya tabu listesinde olmasına rağmen o zamana kadar bulunan en iyi çevrim zamanı değerine sahipse bir sonraki çözüm olarak kabul edilmiştir. Değilse, bir sonraki en iyi komşu seçilerek devam edilmiştir. TA algoritmasının durma kriteri olarak iyileşme olmayan ardışık belirli sayıda iterasyona ulaşılmış olması belirlenmiştir. TA algoritmasının tabu listesi uzunluğu ve durma kriteri yine Gultekin vd. [117] tarafından belirlenmiştir. Buna göre, tabu listesi uzunluğu $= 0.05 \times n \times (m + 1)$ olarak parça sayısı ve makine sayısına bağlı bir şekilde belirlenmiştir. Bu formülün kullanılma sebebi n -birim döngüde $n \times (m + 1)$ robot aktivitesinin bulunmasıdır. Durma kriteri ise iyileşme olmayan ardışık 50 iterasyon olarak belirlenmiştir.

TA içerisinde yer alan ve verilen bir PS ve RHS için çevrim zamanını hesaplayan iteratif algoritma şu şekilde çalışmaktadır: Öncelikle bütün aktivite başlangıç zamanları 0 olarak ayarlanır. Bütün döngülerin A_0 aktivitesi ile başladığı varsayıldığından bu aktiviteden başlanarak robot belirlenmiş olan hareket sıralamasını takip eder. Bu esnada da bütün aktivite başlangıç zamanları, robotun bir önceki aktiviteden bu aktiviteye gelene kadar yapması gereken hareketlerin süreleri nispetinde güncellenir. Bir döngüdeki bütün aktivitelerin başlama zamanları bu şekilde hesaplandığında, başlangıç zamanlarının algoritmanın bir önceki iterasyonundaki değerleriyle, yeni hesaplanan değerleri birbirleriyle karşılaştırılır. Eğer ardışık iki döngüdeki zamanlar birbirleriyle aynı ise algoritma durdurulur. Değilse bu eşitlik sağlanana kadar iterasyonlar devam ettirilir. Algoritma durdurulduğunda, sıralamadaki son aktivitenin başlangıç zamanına, robotun bu hareketi tamamlayarak yükleme stok alanına gelene kadar harcayacağı süre hesaplanıp eklenerek çevrim zamanı hesaplanır.

Proje kapsamında ele alınan problemde amaç pareto etkin çözümler türetmektir. Bu amaçla kullanılacak yöntem Şekil 19'de gösterilmiştir. Bu yöntemde öncelikle verilen robot hareket hızı üst sınır değerlerine göre çevrim zamanının alabileceği en düşük değer (Ct^{LB}) belirlenecektir. Bu değerler belirlenirken, bütün hızlar üst sınıra eşitlenerek Gultekin vd. [117] tarafından geliştirilen yukarıda açıklanan algoritma kullanılacaktır. Bulunan çözüm, en düşük çevrim zamanını verirken, bütün hızlar üst limitlerinde olduğu için, en yüksek enerji tüketimi değerine sahip olacaktır (Şekil 19'deki A noktası). Fakat bu algoritmanın bulacağı çözümün Pareto etkin bir çözüm olma garantisi bulunmamaktadır. Çünkü aynı çevrim zamanını sağlayan alternatif döngüler olabilir ve bu döngüler arasından bazıları diğerlerinden daha az enerji tüketiyor olabilir. Bunun yanında, bütün hızların üst sınıra eşitlenerek elde edilen bir döngüde, robot bazı makineleri boşaltmak için önlerine gittiğinde henüz işlem bitmemiş olabilir. Dolayısıyla, robot hızını düşürerek hala aynı çevrim zamanını elde etmek mümkün olabilmektedir. Bu sebeple, elde edilen bu ilk çözüm, detayları aşağıda açıklanan yöntem kullanılarak aynı çevrim zamanına sahip fakat daha düşük enerji tüketim değerine sahip bir çözüm bulunarak iyileştirilecektir



Şekil 19. Pareto etkin çözümler

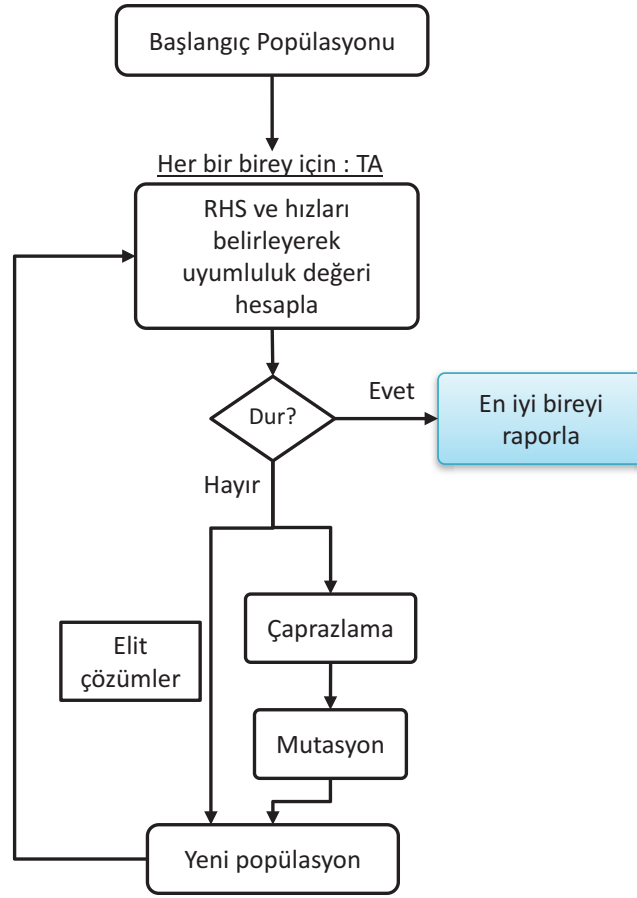
(Şekil 19'deki B noktası). Bu yeni çözümün enerji tüketimi Pareto önyüzde olabilecek en yüksek enerjiyi, yani enerji üst limitini (E^{UB}) temsil etmektedir. Benzer şekilde, bütün hızlar alt limitlerine eşitlenerek Gultekin vd. algoritması kullanılacak ve bu hızlarda olabilecek en düşük çevrim zamanı bulunacaktır (Şekildeki C noktası). Fakat bu çözümde bütün hızlar alt limitlerinde olsa dahi, en düşük toplam enerji değerine sahip olduğunun bir garantisi yoktur. Çünkü aynı çevrim zamanını veren başka döngüler de olabilir ve bu döngülerde robot daha az hareket ediyorsa enerji tüketimi daha düşük olabilecektir. Bu sebeple yine detayları aşağıda verilen yöntem kullanılarak, hızlar alt limitlerindeyken elde edilen en düşük çevrim zamanına sahip çözümler arasından en düşük enerji tüketim değerine sahip olanı bulunacaktır (Şekildeki D noktası). Bütün hızlar alt limitinde olduğu için bu çözüm, Pareto önyüzde en düşük enerji (E^L)-en yüksek çevrim zamanına (Ct^U) sahip çözüm olacaktır.

Bu aşamadan sonra, belirlenen çevrim zamanı alt limit ve üst limiti arası istenen sayıda pareto etkin çözüm çıkartmak üzere eşit aralıklara bölünerek, her bir çevrim zamanına karşılık gelen en düşük enerji tüketim değeri belirlenecektir. Örneğin, toplamda q adet pareto etkin çözüm çıkartılmak isteniyorsa, bunlardan iki tanesi yukarıda bahsedilen çözümler olmak üzere, bunların arasında $q - 2$ adet çözüm türetilenektir. Bunlar arasından h 'inci çözümün çevrim zamanı aşağıdaki şekilde hesaplanır:

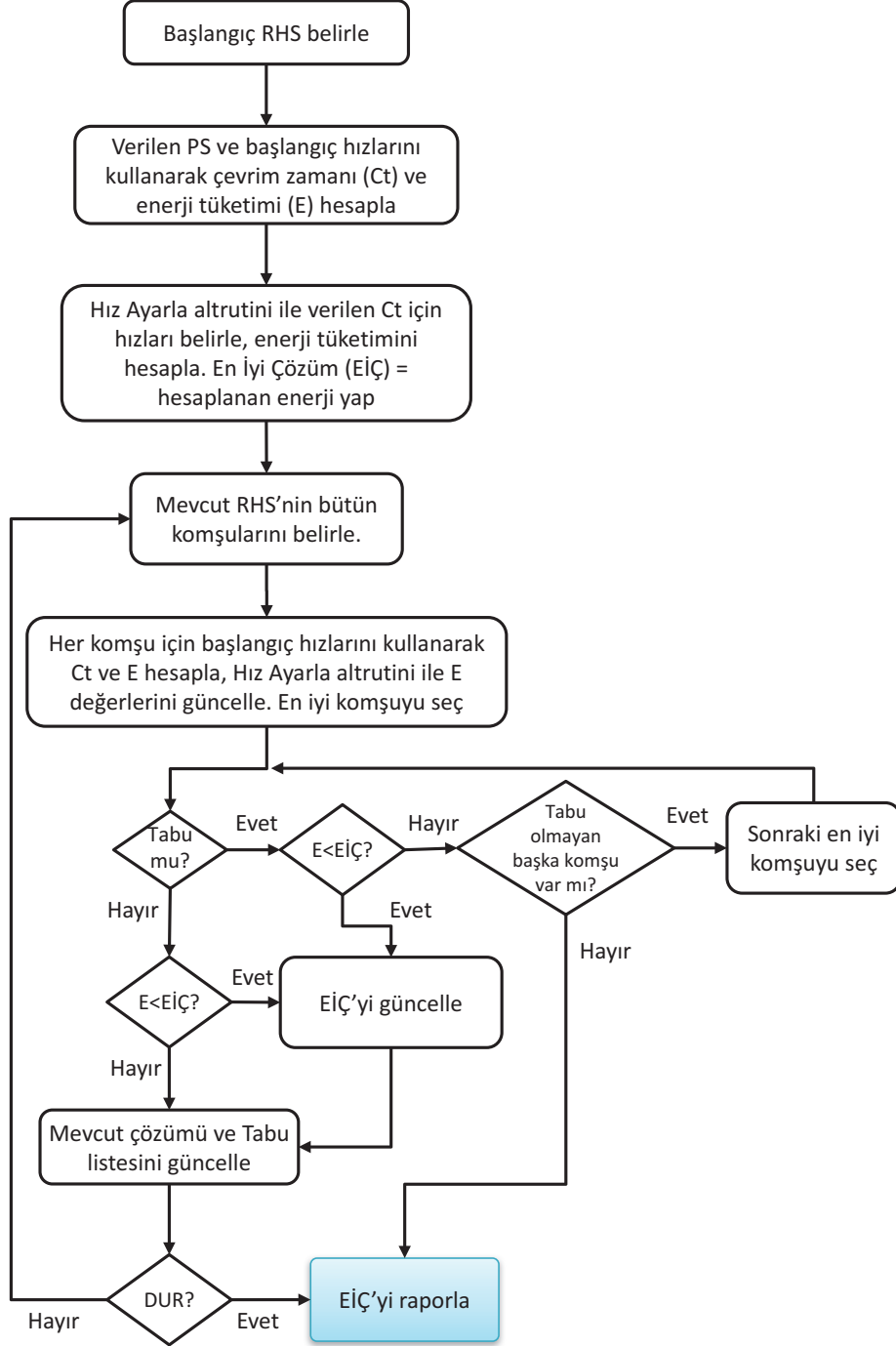
$$Ct^L + \frac{h(Ct^U - Ct^L)}{(q - 1)}, \quad 1 \leq h \leq q \quad (237)$$

Dolayısıyla, problem, belirlenen bu çevrim zamanına sahip çözümler arasından en düşük enerji tüketim değerine sahip olanı bulmaktır. Yani, bu çevrim zamanını sağlayan en düşük enerjili PS, RHS ve hız değerleri bulunmalıdır.

Bahsedilen bu problemin çözümü için Gultekin vd. [117] tarafından geliştirilen hibrit metasezgisel algoritma benzeri bir algoritma geliştirilmiştir. Bu yöntemde yine GA ile PS



Şekil 20. Genetik Algoritma akış şeması



Şekil 21. Tabu Arama algoritması akış şeması

tutulurken, TA, verilen parça sıralaması ve verilen çevrim zamanına karşılık gelen en düşük enerjiye sahip RHS ve hareket hızlarını belirlemektedir. GA'nın akış şeması Şekil 20'de sunulmuştur. Bu algoritmada her bir bireyin uyumluluk fonksiyonunun hesaplanabilmesi için RHS'nin ve parça sıralamalarının belirlenmesi gerekmektedir. Bu da TA ile yapılmaktadır. TA'nın akış şeması da Şekil 20'de sunulmuştur. TA içerisindeki önemli adımlardan bir tanesi, istenen çevrim zamanı için, PS ve RHS sabitlendiğinde en düşük enerjiye sahip hızların belirlenmesidir. Bu da "Hız Ayarla" adı verilen bir alt rutinle sağlanmaktadır. Bu algoritmanın adımları aşağıda sunulmuştur. Bu algoritma, robot hareketleri için bir başlangıç hız değeri ile başlar. Bu değer bütün hızlar için ub veya lb olabileceği gibi, çaprazlamada eşlenen iki bireyden birisinin o anki hızı da olabilir. Algoritmanın adımları anlatılırken, daha önce verilen matematiksel modelden farklı olarak hız notasyonu V_i , $1 \leq i \leq 2n(m+1)$ olarak değiştirilmiştir. Buna göre, eğer $i = 2p - 1$ ise, v_i robotun RHS'de p ($1 \leq p \leq n(m+1)$) pozisyonundaki dolu hareketi yapma hızını göstermektedir. Sonuçta, her aktivite aslında robotun bir makineden bir sonraki makineye yapacağı bir dolu hareketi ifade etmektedir. Diğer taraftan, eğer $i = 2p$ ise, v_i robotun RHS'de p pozisyonundaki dolu hareketi yaptıktan sonra, bir sonraki hareketi yapabilmek için yapması gereken boş hareketin hızını göstermektedir. Robot bir öndeki dolu hareketi tamamladıktan sonra, bulunduğu makinenin önünden, bir sonraki hareketin yapılacağı makinenin önüne boş olarak hareket etmektedir. Eğer bir sonraki hareket de aynı makinenin önünde yapılacaksa, boş hareket 0 olarak gösterilecektir. Örneğin, RHS $A_0A_3A_4 \dots$ şeklinde verildiyse, v_1 , robotun A_0 hareketini yaparken, giriş stoğundan Makine 1'e parça taşıma hızını göstermektedir. v_2 ise, robotun Makine 1 (M1)'i yükledikten sonra, bir sonraki A_3 hareketini yapabilmek için M1'den M3'e yapacağı boş hareketin hızını göstermektedir. Benzer şekilde v_3 , M3'den M4'e parça taşıma hızını gösterirken, v_4 de, M4'ten M4'e boş hareket hızıdır. Robot aynı makinenin önünde olacağı için $v_4 = 0$ olacaktır. Dolayısıyla, eğer v_i gösterimindeki i indisi bir tek sayıysa, dolu bir hareketi, çift sayıysa da boş bir hareketi ifade etmektedir.

v_i gösterimine benzer şekilde, mesafeler için de benzer bir notasyon kullanılmıştır. d_i , robotun v_i hızına karşılık gelen kat ettiği mesafeyi göstermektedir. Yukarıda verilen örnekte, d_0 , M0'dan M1'e olan mesafeyi, d_1 ise M1'den M3'e olan mesafeyi göstermektedir.

Bu notasyon kullanılarak verilen makineler arası mesafe, hareket hızları, ϵ , C_e , C_f , a ve b değerleri için toplam enerji tüketimi şu şekilde hesaplanacaktır:

$$C_f \sum_{p=1}^{n(m+1)} d_{(2p-1)} v_{(2p-1)}^a + C_e \sum_{p=1}^{n(m+1)} d_{(2p)} v_{(2p)}^b \quad (238)$$

Algoritma, istenen çevrim zamanını sağlayacak şekilde hızları tek tek belirli bir adım uzunluğunda artırır veya azaltır. Mevcut CT istenenden daha küçükse (aşağıdaki algoritmada Adım 1), hızlar azaltılabilir. Bunu yaparken, hangi hızın azaltılacağına karar vermek için, mevcut durumda her bir hızın enerji tüketimine etkisi hesaplanır ve en yüksek etkiye sahip olan ilk azaltılacak şekilde bir yol izlenir. Hızların enerji tüketimine katkısı, boş veya dolu bir hız olmasına göre, enerji tüketim fonksiyonunda mesafe çarpanı dikkate alınmadan kalan kısmın türevi alınarak şu şekilde hesaplanır:

$$C_f \cdot a \cdot v_{(i)}^{a-1} \quad (239)$$

$$C_e \cdot b \cdot v_i^{b-1} \quad (240)$$

Algoritmada, enerji etkisi en yüksek olan hız belirli bir adım uzunluğu kadar azaltılır (Adım 1.2). Çevrim zamanı yeniden hesaplanır. Hesaplanan çevrim zamanı isteneni aşarsa (Adım 1.2.1) değişiklik geri alınır ve bu hız bir daha ele alınmamak için işaretlenir. Eğer bu azaltma neticesinde hız kendi alt sınırına ulaştıysa, daha fazla azaltılamayacağı için yine işaretlenir (Adım 1.2.2). Değişiklik neticesi enerji tüketim etkisi yeniden hesaplanır ve başa dönülür.

Diğer taraftan, mevcut CT istenenden daha büyükse (Adım 2), bu defa hız değerlerini artırmak gerekecektir. Bunu yaparken de enerji tüketim etkisi en düşük olan hız ilk artırılarak devam edilir. bir önceki durumdan bir farklılık da, hızda yapılan değişiklik çevrim zamanını değiştirmiyorsa, önceki durum açısından iyi birşeyken, yeni durumda enerji tüketimini artırdığı için kötü birşeydir. Bu sebeple Adım 1.2.1'den farklı olarak, Adım 2.2.1'de eşitlik durumunda da hız değeri eski değerine geri alınır.

Algoritmadaki Adım 3 ile başlayan kısım, daha fazla iyileşme sağlamak için yapılmaktadır. Bu kısımda, istenen çevrim zamanını sağlayan bir çözüm ele alınarak, bu çözümdeki enerji etkisi fazla olan bir hızın adım uzunluğu kadar düşürülüp yerine enerji etkisi düşük bir hızın artırılması, bu sayede de istenen çevrim zamanının korunarak toplam enerji tüketiminin düşürülmesi sağlanmaktadır.

Hız Ayarla

Girdi: $n, m, PS, RHS, \overline{CT}, \epsilon, C_e, C_f, a, b, d_i, v_i, lb, ub$, işlem süreleri ve adım uzunluğu (α)

Çıktı: Güncellenmiş v değerleri

Adım 0: CT ve E hesapla.

Adım 1: Eğer $CT < \overline{CT}$ ise;

Adım 1.1: Bütün hızlar işaretlendiyse Adım 3'e git. Henüz işaretlenmemiş hızlar varsa, hızları enerji tüketim etkisine göre büyükten küçüğe sırala.

Adım 1.2: Sıradaki işaretlenmemiş ilk hızı seç (v_i) ve $v_i \leftarrow v_i - \alpha$ yap. Eğer $v_i < lb$ olursa, $v_i \leftarrow lb$ yap. CT hesapla.

Adım 1.2.1: Eğer $CT > \overline{CT}$ olursa, v_i 'yi eski değerine getir ve işaretle.

Adım 1.2.2: Değilse, v_i 'nin enerji tüketim etkisini hesapla. Eğer $v_i = lb$ olduysa, v_i 'yi işaretle ve Adım 1.1'e dön.

Adım 2: Eğer $CT > \overline{CT}$ ise;

Adım 2.1: Bütün hızlar işaretlendiyse Adım 3'e git. İşaretlenmemiş hızlar varsa, hızları enerji tüketim etkisine göre küçükten büyüğe sırala.

Adım 2.2: Sıradaki işaretlenmemiş ilk hızı seç (v_i) ve $v_i \leftarrow v_i + \alpha$ yap. Eğer $v_i > ub$ olursa, $v_i \leftarrow ub$ yap. CT hesapla.

Adım 2.2.1: Eğer $CT \leq \overline{CT}$ olursa, v_i 'yi eski değerine getir ve işaretle.

Adım 2.2.2: Değilse, v_i 'nin enerji tüketim etkisini hesapla. Eğer $v_i = ub$ ise, v_i 'yi işaretle. Adım 2.1'e dön.

- Adım 3:** Mevcut hızlarla E hesapla ve $E^* \leftarrow E$ yap. Bütün hızların işaretini temizle.
- Adım 4:** Bütün hızlar işaretlendiyse DUR. İşaretlenmemiş hızlar varsa, hızları enerji tüketim etkisine göre büyükten küçüğe sırala.
- Adım 5:** Sıradaki işaretlenmemiş ilk hızı seç (v_i) ve $v_i \leftarrow v_i - \alpha$ yap. Eğer $v_i < lb$ olursa, $v_i \leftarrow lb$ yap ve v_i 'yi işaretle. CT hesapla. Eğer CT 'de bir değişiklik olmadıysa v_i 'yi eski değerine getir ve işaretle ve Adım 4'e dön. Değilse, sonraki adıma devam et.
- Adım 6:** Hız vektörünün kopyasını oluştur (\bar{v}) ve bu kopyadaki bütün işaretleri temizle.
- Adım 7:** \bar{v} 'yi enerji etkisine göre küçükten büyüğe sırala.

Adım 7.1: Eğer \bar{v} 'deki bütün hızlar işaretlendiyse ve $CT > \overline{CT}$ ise v_i 'yi eski değerine getir ve işaretle, Adım 4'e git.

Adım 7.2: Değilse, Eğer \bar{v} 'deki bütün hızlar işaretlendiyse ve $E^* < E$ ise, v_i 'yi eski değerine getir ve işaretle, Adım 4'e git.

Adım 7.3: Değilse, Eğer \bar{v} 'deki bütün hızlar işaretlendiyse ve $E < E^*$ ise, $E^* \leftarrow E$ yap.

Adım 7.4: Değilse, henüz işaretlenmemiş hızlar varsa, \bar{v} 'yi enerji tüketim etkisine göre küçükten büyüğe sırala.

Adım 7.4.1: Sıradaki işaretlenmemiş ilk hızı seç (v_j) ve $v_j \leftarrow v_j + \alpha$ yap. Eğer $v_j > ub$ olursa, $v_j \leftarrow ub$ yap ve v_j 'yi \bar{v} 'de işaretle. CT ve E hesapla.

Adım 7.4.1.1: Eğer $CT < \overline{CT}$ olursa, v_j 'yi eski değerine getir ve \bar{v} 'de işaretle.

Adım 7.4.1.2: Değilse, E hesapla ve Adım 7'ye git.

Sonuç olarak, geliştirilen HMA, Pareto etkin çözümler türetmek için kullanılmaktadır. Algoritmanın iki kullanım şekli mümkündür. Bunlardan ilki, çıkartılması istenen pareto etkin çözümleri için çevrim zamanı değerlerinin dışarıdan algoritmaya verildiği versiyondur. Bu şekilde bir çalışma, algoritmanın performansını test etmek amacıyla geliştirilen matematiksel modelle karşılaştırma aşamasından kullanılmıştır. Matematiksel modelle türetilen pareto etkin çözümlere ait çevrim zamanı değerleri sezgisel girdi olarak verilmiş ve sezgiselin de aynı çevrim zamanına karşılık gelen çözümler türetmesi sağlanmıştır. İkinci kullanım şekli ise, algoritmaya herhangi bir şekilde çevrim zamanı değeri vermeden, sadece kaç adet çözüm türetilmek istendiğinin verilmesi şeklindedir. Bu durumda, algoritma Denklem 237 formülünü kullanarak, Pareto önyüzdeki çevrim zamanı değerlerini kendisi belirler. Aşağıda bütün algoritmanın adımları bu ikinci çalışma şekli esas alınarak sunulmuştur:

Pareto Etkin Çözüm Türetme Algoritması

Girdi: $n, m, \epsilon, C_e, C_f, a, b, d_i, v_i, lb, ub$, işlem süreleri, istenen çözüm sayısı (q), ve adım uzunluğu (α)

Çıktı: İstenen sayıda eşit aralıklı Pareto etkin çözüm ve bunlara karşılık gelen PS, RHS, hareket hızları, CT ve E değerleri.

Adım 1: $v_i \leftarrow ub$ yap ve Gultekin vd. [117] algoritmasını kullanarak CT^{LB} değerini bul.

Adım 2: Bulunan CT^{LB} değerini HMA algoritmasına girerek E^{UB} değerini bul ve kaydet. Bulunan bu çözüm ilk Pareto etkin çözümdür.

Adım 3: $v_i \leftarrow lb$ yap ve Gultekin vd. [117] algoritmasını kullanarak CT^{UB} değerini bul.

Adım 4: Bulunan CT^{UB} değerini HMA algoritmasına girerek E^{LB} değerini bul ve kaydet.

Adım 5: $h = 1$ 'den $(q - 2)$ 'ye kadar tekrarla;

Adım 5.1: $\overline{CT} \leftarrow Ct^{LB} + \frac{h(Ct^{UB} - Ct^{LB})}{(q - 1)}$ yap.

Adım 5.2: \overline{CT} değerini HMA'ya girerek yeni Pareto etkin çözüm bul ve kaydet.

5.4.1 Parametre Kalibrasyonu

Bu bölümde geliştirilen HMA algoritmada kullanılan parametreler için en iyi değerleri belirlemek üzere yapılan hesaplamalı çalışmalardan bahsedilecektir. Geliştirilen HMA C++ dilinde kodlanmış ve Microsoft Visual Studio ile Intel Xeon E5645 2.40 GHz CPU (26 cores) işlemcili ve 18 GB RAM içeren 64bit işletim sisteminde çalıştırılmıştır.

Parametre kalibrasyonu için $n = 3$ parça ve $m = 3$ makine içeren bir örnek problem daha önce Carlier [122] tarafından akış atölyesi problemleri için önerilmiş veriler kullanılarak türetilmiştir. Carlier tarafından önerilen verilerde sadece makine işlem süreleri bulunmaktadır. Bu verilerde ϵ , lb , ub , C_e , C_f , a , b ve makineler arası mesafe değerleri bulunmamaktadır. Parametre kalibrasyonu için, sonuçlar üzerinden ciddi etkileri olmadığı anlaşıldığından bazı parametreler sabitlenirken, diğer bazıları için ikişer farklı değer kullanılarak farklı test problemleri türetilmiştir. Buna göre Carlier dosyasından alınan işlem süreleri şu şekildedir:

$$P = \begin{bmatrix} 243 & 91 & 483 \\ 263 & 285 & 293 \\ 48 & 227 & 581 \end{bmatrix}$$

Bunun yanında $\epsilon = 1$, $lb = 1$, $ub = 3$, $a = b = 2$ sabitlenerek kullanılmıştır. (C_e, C_f) ikilisi olarak $(1,1)$ ve $(1,2)$ olmak üzere iki farklı değer kullanılmıştır. Makineler arası mesafe için ise aşağıdaki 4 farklı matris kullanılmıştır. Sonuçta toplamda $2 \times 4 = 8$ farklı problem verisi oluşturulmuştur. Makineler arası mesafeler için kullanılan değerler şu şekildedir:

$$D1 = \begin{bmatrix} 0 & 1 & 2 & 3 & 4 \\ 1 & 0 & 1 & 2 & 3 \\ 2 & 1 & 0 & 1 & 2 \\ 3 & 2 & 1 & 0 & 1 \\ 4 & 3 & 2 & 1 & 0 \end{bmatrix} \quad D2 = \begin{bmatrix} 0 & 12 & 24 & 36 & 48 \\ 12 & 0 & 12 & 24 & 36 \\ 24 & 12 & 0 & 12 & 24 \\ 36 & 24 & 12 & 0 & 12 \\ 48 & 36 & 24 & 12 & 0 \end{bmatrix}$$
$$D3 = \begin{bmatrix} 0 & 2 & 5 & 7 & 11 \\ 2 & 0 & 3 & 5 & 9 \\ 5 & 3 & 0 & 2 & 6 \\ 7 & 5 & 2 & 0 & 4 \\ 11 & 9 & 6 & 4 & 0 \end{bmatrix} \quad D4 = \begin{bmatrix} 0 & 6 & 13 & 21 & 26 \\ 6 & 0 & 7 & 15 & 20 \\ 13 & 7 & 0 & 8 & 13 \\ 21 & 15 & 8 & 0 & 5 \\ 26 & 20 & 13 & 5 & 0 \end{bmatrix}$$

Tablo 31. HMA parametre kalibrasyonu için kullanılan değerler

Parametre	Küçük Değer	Büyük Değer
GA Popülasyon Büyüklüğü	5	10
GA Durma Kriteri	15	30
GA Elit Çözüm Oranı	0.1	0.2
GA Mutasyon Olasılığı	0.1	0.15
TA Tabu Listesi Uzunluğu	0.1	0.15
TA Durma Kriteri	15	30

Hız ayarlama alt rutinindeki adım uzunluğu parametresi için ön denemeler yapılmış ve $ub/10$, $ub/100$ ve $ub/1000$ değerleri denenmiştir. Çözüm zamanı açısından kısa olsa da $ub/10$ değerinin çözüm kalitesi açısından daha kötü olduğu tespit edilmiştir. Diğer iki değer ise çözüm süresi ve kalitesi açısından birbirlerine oldukça yakın olduğu belirlenmiştir. Bu sebeple de adım uzunluğu olarak $ub/100$ 'ün kullanılmasına karar verilmiştir.

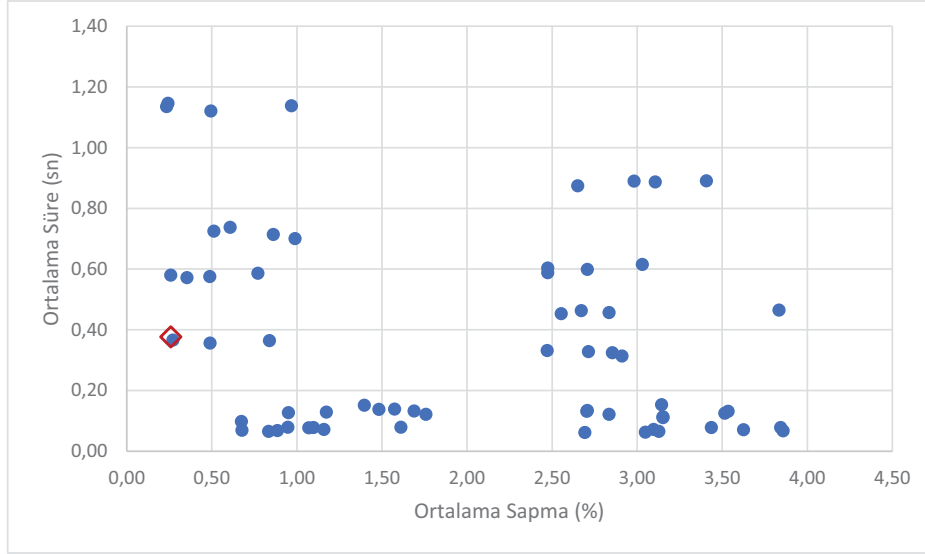
GA için popülasyon büyüklüğü, durma kriteri, elit çözüm yüzdesi, mutasyon olasılığı parametreleri, TA'nın tabu listesi uzunluğu ve durma kriteri parametrelerinin tamamı tam faktöryel deney tasarımında ikişer farklı değerle ele alınmıştır. Ele alınan parametreler ve kullanılan değerler Tablo 31'de sunulmuştur. Bu durumda toplamda 2^6 farklı kombinasyon ortaya çıkmaktadır. 8 farklı test problemi ve her bir problem verisi için pareto önyüz üzerinde 5+1 (Pareto önyüzdeki 5 noktaya ilave olarak bir de bütün hızlar üst limitteyken çözdürülen hız minimizasyonu probleminden elde edilen sonuç) farklı nokta türetilmiştir. Sonuçta $8 \times 6 \times 2^6 = 3072$ adet problem çözdürülmüştür.

Çözdürülen problemlerden elde edilen sonuçlar Tablo 32'da özetlenmiştir. Bu tabloda, parametreler için denenmiş olan 64 kombinasyonla çözdürülmüş olan toplamda $8 \times 6 = 48$ farklı probleme ait veriler yer almaktadır. Her problemin pareto önyüzündeki her nokta için herhangi bir kombinasyonla elde edilen enerji tüketim değeri (E_i), 64 kombinasyondan elde edilen en iyi enerji tüketim değeriyle (E^*) kıyaslanarak yüzde sapma değerleri (S_i) aşağıdaki şekilde hesaplanmıştır.

$$S_i = \frac{100(E_i - E^*)}{E^*}$$

Herhangi bir kombinasyonla çözdürülen 48 problemde elde edilen en yüksek sapma değeri de tabloda maksimum sapma olarak yer almıştır. Tablodaki ortalama çözüm süresi de bir kombinasyonla çözdürülen bu 48 problemin ortalama çözüm süresini göstermektedir.

Tablo, elde edilen ortalama sapma değerine göre küçükten büyüğe sıralanmıştır. Tablodaki değerlerin görselleştirilmesi için Şekil 22'da sunulan grafik çizdirilmiştir. Burada amaç ortalama ve maksimum saptması ve çözüm süresi düşük olan kombinasyonu belirlemektir. Değerler incelendiğinde, ortalama sapma değeri en düşük (%0.24) olmasa da ona çok yakın olan (%0.26), çözüm süresi 0.38 saniye gibi oldukça düşük olan ve en kötü performans sergilediği örnekteki sapma değeri en düşük (%4.53) olan 41 numaralı kombinasyonun seçilmesine karar verilmiştir. Bu kombinasyonla elde edilen çözüm Şekil 22'da işaretlenmiştir. Bu kombinasyonda, popülasyon büyüklüğü= $10 \times n$, GA durma kriteri=15, elit çözüm oranı=0.1, mutasyon



Şekil 22. HMA ortalama sapma ve ortalama çözüm süreleri

olasılığı=0.1, Tabu listesi uzunluğu=0.1 ve TA durma kriteri=30'dur.

5.5 Hesaplamalı Deney Sonuçları

Bu bölümde geliştirilmiş olan HMA algoritmasının çözüm süresi ve çözüm kalitesi açısından yapılan performans testlerinin sonuçları sunulacaktır. C++ dilinde kodlanmış olan HMA algoritmasının sonuçları GAMS dilinde uygulanmış ve CPLEX 12.6.2 çözdürücüsü ile çözdürülen MISOCP'nin sonuçları ile karşılaştırılacaktır. Literatürde, ele alınan bu problem için test problemleri bulunmamaktadır. Fakat, parçalar sistemdeki makinelerde akış atölyesi (flowshop) mantığıyla hep aynı makine sırasıyla işlendiği için, akış atölyesi çizelgeleme problemleri için kullanılan test verileri baz alınarak kendi problemimiz için test problemleri oluşturulmuştur. Bu amaçla Carlier [122] tarafından önerilmiş olan test problemlerinden bazıları kullanılmıştır. Bu test verilerinde farklı iş sayısı ve makine sayısına sahip sistemler için, işlerin makinelerdeki işlem süreleri verilmiştir. Fakat, parça yükleme boşaltma süresi, (ϵ), makineler arası mesafe, enerji tüketim fonksiyonu parametreler (C_e , C_f , a , b), hareket hızı alt ve üst limitleri (lb , ub) gibi kendi problemimize özgü parametreler bu test problemlerinde bulunmamaktadır. Bu sebeple, öncelikle MISOCP'un makul sürelerde çözebileceği problem büyüklüklerinde üç tanesi seçilmiştir. ($n-m$) ikilisi sırasıyla iş ve makine büyüklüğünü göstermek üzere (3-3), (3-4), (4-3) büyüklüklerinde veriler kullanılmıştır. Ayrıca, her ne kadar MISOCP'la çözüm alınamasa da, sezgisel algoritmanın çözüm süresini görmek açısından, test problemleri arasındaki en büyük iş-makine sayısına sahip (8-9)'luk veri de kullanılmıştır.

Öncelikle hangi parametrelerin çözüm kalitesine ve çözüm süresine etkisinin olabileceği araştırılmıştır. Buna göre, C_e ve C_f parametrelerinin birbirlerine göre büyüklüğü, makineler arası mesafenin işlem süreleri ve hız alt ve üst limitlerine göre büyüklüğü ve farklı makineler arası mesafelerin birbirlerine göre büyüklüklerinin etkisi olabileceği değerlendirilmiştir. Her döngüde her makinenin eşit sayıda yüklenip boşaltıldığı bu sebeple de makine yükleme

Tablo 32. HMA parametre kombinasyonları sapma değerleri ve çözüm süreleri

Kombinasyon	Ortalama Sapma (%)	Ortalama Süre (sn)	Maksimum Sapma (%)
48	0,24	1,13	7,45
64	0,24	1,15	4,53
62	0,26	0,58	4,97
41	0,26	0,38	4,53
57	0,27	0,37	5,19
42	0,35	0,57	5,19
58	0,49	0,58	11,00
45	0,49	0,36	8,98
44	0,50	1,12	6,00
63	0,51	0,72	4,53
43	0,61	0,74	15,16
36	0,67	0,10	7,14
38	0,68	0,07	5,19
46	0,77	0,59	12,55
54	0,83	0,06	6,42
61	0,84	0,36	8,98
47	0,86	0,71	8,98
34	0,89	0,07	9,64
37	0,95	0,08	9,64
35	0,95	0,13	7,14
60	0,97	1,14	15,16
59	0,99	0,70	8,98
49	1,07	0,08	6,00
50	1,10	0,08	8,98
33	1,16	0,07	9,64
51	1,17	0,13	6,32
55	1,40	0,15	7,14
39	1,48	0,14	9,64
40	1,57	0,14	8,98
53	1,61	0,08	8,98
52	1,69	0,13	6,42
56	1,76	0,12	9,64
25	2,47	0,33	7,14
27	2,48	0,59	7,14
31	2,48	0,60	7,14
30	2,55	0,45	8,59
28	2,65	0,87	7,14
26	2,67	0,46	7,14
18	2,69	0,06	7,14
19	2,70	0,13	7,14
3	2,71	0,13	7,14
11	2,71	0,60	7,14
29	2,71	0,33	7,14
4	2,83	0,12	7,14
14	2,83	0,46	7,14
9	2,85	0,32	7,14
13	2,91	0,31	7,14
12	2,98	0,89	15,16
15	3,03	0,62	7,14
2	3,05	0,06	7,14
21	3,10	0,07	7,14
16	3,11	0,89	7,14
6	3,13	0,07	9,64
20	3,14	0,15	7,14
8	3,15	0,11	7,14
24	3,15	0,11	7,14
32	3,41	0,89	15,16
22	3,44	0,08	7,14
23	3,52	0,12	15,16
7	3,54	0,13	8,98
17	3,63	0,07	9,64
10	3,83	0,46	15,16
5	3,84	0,08	8,98
1	3,86	0,07	9,64

Tablo 33. Hesaplamalı çalışmada kullanılan parametre değerleri ve özellikleri

Parametre	Özellik / Değer			
$(n-m)$	(3-3)	(3-4)	(4-3)	(8-9)
(C_e, C_f)	(1, 1)	(1, 2)		
Mesafe Büyüklüğü	K	B		
Mesafe Yapısı	Ö	F		

boşaltma zamanının çözüme fazla bir etkisi olmadığı değerlendirilmiştir. Bu sebeple bu parametre $\epsilon = 1$ olarak sabitlenmiştir. Ayrıca, makineler arası mesafenin kendi başına büyüklüğü değil, işlem süreleri ve hız limitlerine göre oranının etkili olduğu değerlendirilmiştir. Bu sebeple, $lb = 1$, $ub = 3$ değerlerine sabitlenmiş, Carrier'in test problemlerinde verilen işlem süresi aynı şekilde alınmış ve buna karşılık gelen küçük (K) - büyük (B) değerler ve ardışık makineler arası mesafenin özdeş (Ö) veya farklı (F) olduğu değerler olmak üzere dört farklı mesafe değeri türetilmiştir. Son olarak, enerji tüketimi için $a = b = 2$ değerinin hem sistemdeki doğrusal olmayan yapıyı temsil ettiği ve hem de yaygın kullanıldığı için sabitlenmiş, $(C_e$ ve $C_f)$ ikilisi için de iki farklı değer kullanılmıştır. Sonuçta kullanılan problemlere ait özellikler Tablo 33'de özetlenmiştir. Sonuç olarak her problem boyutu için 8 farklı veri seti ve her veri seti için Pareto Önyüz üzerinden 5+1 nokta için çözüm alınmıştır. MISOCP (8-9)'luk veri setinin çözümünde kullanılamamıştır. HMA algoritması içerisindeki rassallığın etkisini kaldırmak için, her veriye ait her nokta toplamda 5 defa tekrar tekrar çözdürülmüştür. Dolayısıyla, MISOCP ile 144, HMA ile ise 960 defa çözüm alınmıştır. MISOCP, her bir nokta için 3600 sn (1 saat) zaman limiti ile çalıştırılmıştır.

Karşılaştırma yapılırken öncelikle MISOCP çözdürülmüş, CT^{LB} ve CT^{UB} değerleri belirlenmiş, sonra da daha önce açıklandığı şekilde Denklem 237 kullanılarak çevrim zamanı limitleri belirlenmiştir. Sonrasından her bir limit için MISOCP çözdürülerek enerji değerleri elde edilmiştir. (3-3), (3-4), (4-3) problem verilerinin hem MISOCP ve hem de HMA ile çözülmesinden elde edilen sonuçlar sırasıyla Tablolar 34, 37 ve 38'de sunulmuştur. Bu tablolarda, M.Y.; Mesafe Yapısını, M.B; Mesafe Büyüklüğünü, P.N.; Pareto Noktayı, A.S.; Alt Sınırı ifade eden kısaltmalardır. P.N. sütununda 0 numaralı nokta olarak sunulan çözüm, bütün hızların üst limitte olduğu, çevrim zamanını minimize eden problemde elde edilen çözümdür. 1 numaralı nokta ise aynı çevrim zamanına karşılık gelen enerji tüketiminin minimize edildiği çözüme karşılık gelen noktadır. 5 numaralı nokta ise bütün hızların alt limitte olduğu, çevrim zamanı üst limitini veren noktalar arsından enerji yüketimini minimize eden noktadır. 2, 3 ve 4 numaralı noktalar ise bu uç noktalar arasında yer alan noktalardır.

MISOCP bu şekilde çözdürüldükten sonra ara noktalar için belirlenen çevrim zamanı değerleri HMA algoritmasına girdi olarak verilmiştir. HMA algoritması, 0, 1 ve 5 numaralı noktaları kendisi belirlemede, MISOCP'tan gelen limitleri kullanmamaktadır. Ara noktalar içinse MISOCP sonucunda belirlenen üst limit değerlerine göre çözüm almaktadır. Bunun yapılma nedeni MISOCP ve HMA'nın iki kriter açısından karşılaştırmasının anlamlı olmasıdır. Dolayısıyla, her iki model de aynı çevrim zamanına karşılık gelen enerji tüketim değerleri üzerinden karşılaştırılmıştır. Bu karşılaştırma, HMA çözümünün MISOCP çözümünden yüzde

sapması aşağıdaki şekilde hesaplanarak yapılmıştır.

$$\text{Sapma} = \frac{100 \times (\text{HMA-MISOCP})}{\text{MISOCP}}$$

Her probleme ait her Pareto nokta için HMA algoritması 5 defa tekrar çalıştırıldığı için, tablolarda bu algoritma ile elde edilen çözümlere ait sapmaların ortalama, minimum ve maksimum değerleri sunulmuştur.

MISOCP ile bütün veriler sadece (3-3) problemi için çözdürülebilmektedir. Tablolarda ZL ile gösterilen satırlar MISOCP'un optimal çözüme ulaşmadan zaman limiti ile durduğunu belirtmektedir. Tablolar 37 ve 38'de de görülebileceği gibi, bazı problem verileri zaman limiti ile durduğunda elde olan en iyi tamsayı çözümü E sütununda raporlanmıştır. Bu çözümler için, CPLEX'in verdiği alt sınır değerler de A.S. sütununda sunulmuştur. Örneğin, Tablo 37'de üçüncü satırda böyle bir durum söz konusudur. Zaman limiti ile duran MISOCP, 1 saat sonunda 81.8 gibi bir enerji tüketim değerine sahip olurlu bir çözüm bulabilmiştir. Model durduğu andaki alt sınır değeri ise 25.8'dir. Bu tablolarda görüleceği üzere, bu durumlarda HMA algoritması MISOCP'tan daha iyi çözümlere ulaşabilmiştir. Örneğin 5 numaralı verinin 3 numaralı pareto noktasında MISOCP en iyi 197.5 gibi bir enerji tüketim değeri bulmuşken, HMA algoritması 84.3 gibi bir çözüm de bulabilmiştir. Diğer taraftan, bazı problemler için MISOCP ile zaman limiti içerisinde herhangi bir olurlu çözüme ulaşamamıştır. Örneğin, 1 numaralı verinin 1 ve 3 numaralı pareto noktalarında herhangi bir olurlu çözüme ulaşamamıştır. Diğer taraftan, 4 numaralı veri için 0 ve 5 numaralı pareto noktaları için olurlu çözüm bulunamadığı için ara noktalardaki CT limitleri belirlenememiştir. Bu tip durumlarda HMA algoritması limit değerlerini girdi olarak almadan, kendisi belirleyecek şekilde çalıştırılmıştır.

(3-3) probleminden elde edilen sonuçlar Tablo 34'de incelendiğinde, her problem verisi için HMA algoritmasının 0 numaralı pareto noktasına karşılık gelen minimum çevrim zamanını her tekrarında bulabildiği görülmüştür. Bütün çözümlerin ortalama sapması %2'dir. HMA için alınan beş tekrardan elde edilen en iyi çözümler üzerinden hesaplanırsa ortalama sapma %1.4 olmaktadır. Her tekrarın en kötüsü alındığında ise %2.6 olmaktadır. Bütün çözümler arasından elde edilne en yüksek sapma değeri %13.6 olmuştur. (3-3) problemi için HMA ile alınan toplam 240 çözümden sadece bir tanesi %'10'dan fazla saparken, %5'ten fazla sapan durumların sayısı sadece 34'tür. Bu da HMA algoritmasının çözüm kalitesinin oldukça iyi olduğunu göstermektedir.

HMA algoritmasının ortalama çözüm süresi 0.4 sn gibi oldukça kısa bir süredir. MISOCP'un (3-3) problemleri için ortalama çözüm süresi 40.2 sn'dir. HMA algoritmasının çözüm süresi makine sayısı artırıldığında (3-4) problemi için 0.56 sn'ye çıkarken iş sayısı bir atıldığında (4-3) problemi için 0.3 saniye olmaktadır. Fakat, tablolarda da görüldüğü üzere, MISOCP'un çözüm süresi iş sayısının veya makine sayısının bir artırılmasıyla çok hızlı bir şekilde artmakta ve bazı noktalar için zaman limiti içerisinde çözüm alınamamaktadır. Çözdürülen en büyük problem verisi olan (8-9) için HMA algoritmasının ortalama çözüm süresi 132.01 sn olmaktadır. Bu tip sistemler için gerçek hayatta kullanılabilecek problemlerde genelde makine sayıları daha düşük değerlerde olmaktadır. Bu da düşünüldüğünde, bu karmaşık problem için elde edilen çözüm süresinin makul olduğu değerlendirilmiştir.

(3x3) veri kümesi için elde edilen sonuçlara parametrelerin etkisini görmek için Tablo

Tablo 34. 3x3 veri kümesi için MISOCP HMA karşılaştırma sonuçları

Veri	(C_e, C_f)	M.Y.	M.B.	P.N.	MISOCP			HMA			
					CT	E	Süre (sn)	Ort. (%)	Min. (%)	Mak. (%)	Süre (sn)
1	(1, 1)	Ö	K	0	1372,99	288,00	49,64	0,0	0,0	0,0	0,3
				1	1372,99	127,63	34,45	0,3	0,3	0,3	0,5
				2	1374,98	67,87	31,25	0,7	0,7	0,7	0,2
				3	1376,97	46,92	30,89	0,8	0,8	0,8	0,3
				4	1378,96	37,22	29,17	0,6	0,6	0,6	0,3
				5	1380,95	32,00	27,02	1,3	0,0	6,3	0,5
2	(1, 1)	Ö	B	0	1416,95	3456,00	56,91	0,0	0,0	0,0	0,3
				1	1416,95	1558,60	38,89	0,1	0,1	0,1	0,6
				2	1440,93	839,24	37,89	0,3	0,3	0,3	0,3
				3	1464,90	571,88	38,13	3,8	3,0	7,2	0,3
				4	1488,88	446,92	36,63	9,3	8,2	13,6	0,4
				5	1512,86	384,00	35,25	2,5	0,0	6,3	0,6
3	(1, 1)	F	K	0	1380,99	756,00	45,98	0,0	0,0	0,0	0,3
				1	1380,99	371,76	43,49	0,1	0,1	0,1	0,5
				2	1386,99	191,91	41,64	0,3	0,3	0,3	0,3
				3	1392,98	128,96	44,67	3,0	0,2	4,8	0,3
				4	1398,98	99,82	36,72	3,8	0,2	6,2	0,2
				5	1404,97	84,00	39,14	0,0	0,0	0,0	0,5
4	(1, 1)	F	B	0	1395,06	1962,00	56,61	0,0	0,0	0,0	0,3
				1	1395,06	843,61	49,02	1,5	1,5	1,5	0,6
				2	1408,09	451,03	50,94	1,2	0,0	3,1	0,3
				3	1421,11	315,94	52,25	5,9	4,2	8,6	0,3
				4	1434,14	250,05	52,50	1,9	0,8	3,6	0,3
				5	1447,17	218,00	46,80	0,0	0,0	0,0	0,5
5	(1, 2)	Ö	K	0	1372,99	396,00	52,08	0,0	0,0	0,0	0,3
				1	1372,99	186,89	29,38	0,8	0,6	1,7	0,5
				2	1374,98	94,85	33,55	5,0	5,0	5,0	0,3
				3	1376,98	64,70	32,56	4,0	4,0	4,0	0,3
				4	1378,97	50,74	28,94	1,4	1,4	1,4	0,3
				5	1380,96	44,00	29,22	0,0	0,0	0,0	0,5
6	(1, 2)	Ö	B	0	1416,95	4752,00	51,88	0,0	0,0	0,0	0,3
				1	1416,95	2268,31	31,88	0,5	0,5	0,5	0,5
				2	1440,93	1161,88	31,84	4,4	4,4	4,4	0,3
				3	1464,90	788,25	35,89	8,1	8,1	8,1	0,4
				4	1488,88	608,52	36,39	3,1	1,8	5,1	0,3
				5	1512,86	526,00	37,58	0,4	0,4	0,4	0,6
7	(1, 2)	F	K	0	1380,99	1053,00	56,81	0,0	0,0	0,0	0,3
				1	1380,99	546,25	34,95	0,9	0,5	1,6	0,5
				2	1386,99	269,60	35,72	4,6	4,6	4,6	0,3
				3	1392,98	179,12	33,05	7,8	7,8	7,8	0,3
				4	1398,98	137,24	35,14	1,0	1,0	1,0	0,3
				5	1404,97	117,00	36,42	0,0	0,0	0,0	0,5
8	(1, 2)	F	B	0	1395,06	2664,00	32,30	0,0	0,0	0,0	0,3
				1	1395,06	1229,75	45,92	1,3	1,3	1,3	0,6
				2	1408,09	628,29	48,78	5,1	3,3	5,6	0,3
				3	1421,11	431,58	47,45	5,1	2,5	5,8	0,3
				4	1434,14	340,52	46,63	3,6	0,4	4,5	0,3
				5	1447,17	295,00	37,39	0,3	0,3	0,3	0,5

Tablo 35. 3x3 veri kümesi için parametre bazında elde edilen sonuçlar

Parametre	Değer	MISOCP		HMA		
		Süre (sn)	Ort. (%)	Min. (%)	Max. (%)	Süre (sn)
(C_e, C_f)	(1, 1)	41,9	1,6	0,9	2,7	0,4
	(1, 2)	38,4	2,4	2,0	2,6	0,4
M.Y.	Ö	36,6	2,0	1,7	2,8	0,4
	F	43,8	2,0	1,2	2,5	0,4
M.B.	K	37,2	1,5	1,2	2,0	0,3
	B	43,2	2,4	1,7	3,3	0,4

Tablo 36. 3x3 veri kümesi için pareto nokta bazında yüzde sapma ve çözüm süreleri

Pareto Nokta	Ort. Sapma (%)	Ort. Süre (sn)
0	0,0	0,3
1	0,7	0,5
2	2,7	0,3
3	4,8	0,3
4	3,1	0,3
5	0,6	0,5

35 oluşturulmuştur. Bu tabloda, Tablo 34'da sunulan sonuçların parametre değeri bazında ortalaması alınmıştır. Tablo incelendiğinde, MISOCP çözüm sürelerine göre $C_e = C_f$ durumunun $C_f > C_e$ durumuna göre, mesafelerin farklı olduğu durumun, özdeş olduğu duruma göre ve mesafelerin büyük olduğu durumun küçük olduğu duruma göre daha uzun sürdüğü görülebilir. HMA'da ise çözüm süreleri arasındaki tek fark, mesafelerin büyük olduğu durumun küçük olduğu duruma göre daha uzun sürmesidir. HMA algoritmasının sapması incelediğinde hem ortalama hem de minimum sapma açısından $C_f > C_e$ durumunda diğer duruma göre daha kötü sonuçlar verdiği görülebilir. Mesafelerin özdeş veya farklı olmalarının HMA algoritması sonuçlarını fazla etkilemediği, fakat mesafelerin büyük olduğu durumda ortalama, minimum ve maksimum sapma değerlerinin tamamının daha yüksek olduğu görülebilir.

(3-3) verisi için çözdürülen sekiz problem verisinin Pareto nokta bazında ortalamaları alındığında Tablo 36'da sunulan sonuçlar elde edilmiştir. Bu tablodan, ortalama sapma değerlerinin Pareto önyüzün orta kısımlarında daha yüksek olduğu, iç noktalarda daha iyi sonuçlar elde ettiği görülmektedir. Noktalar arasında ortalama çözüm sürelerindeki farklılık da aynı tablosa sunulmuştur. Buna göre, en düşük çevrim zamanına karşılık gelen en düşük enerji tüketimine sahip çözümün arandığı nokta 1 ve çevrim zamanı üst limitine karşılık gelen en düşük enerjili çözümün arandığı nokta 5 için çözüm süreleri diğer noktalara göre daha uzundur.

(3-4) ve (4-3) problemlerinde MISOCP ile her probleme zaman limiti içerisinde çözüm üretilmediğinden Tablolar 37 ve 38'de sapma değerleri değil, doğrudan CT ve E değerleri sunulmuştur. Yine HMA ile alınan 5 değerlerin ortalama, minimum ve maksimum değerleri tablolarda yer almaktadır.

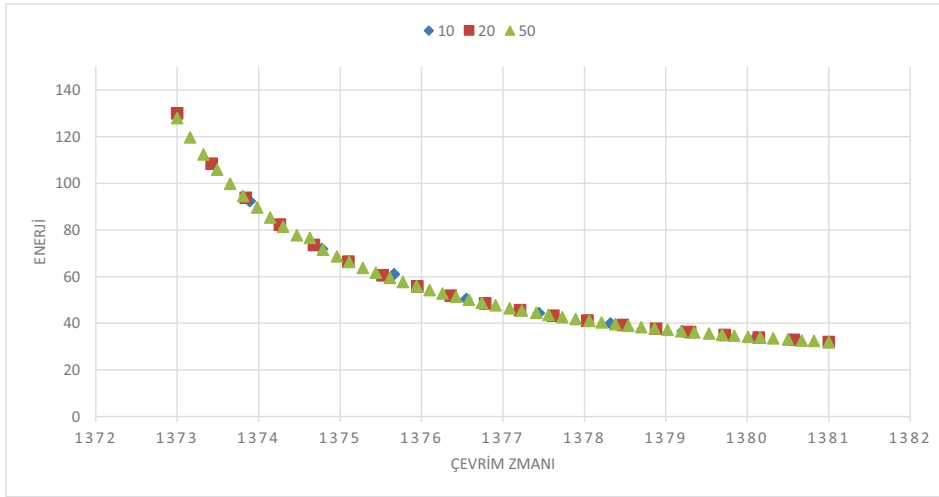
Bu testlerin yanında, HMA algoritmasına girdi olarak CT limitlerini vermeden, Pareto

Tablo 37. 3x4 veri kümesi için MISOCP HMA karşılaştırma sonuçları

Veri	(C_e, C_f)	M.Y.	M.B.	P.N.	MISOCP				HMA				
					CT	E	A.S.	Süre (sn)	CT	Ort.	Min.	Mak.	Süre (sn)
1	(1, 1)	Ö	K	0	1252,0	435,0	252,3	ZL	1252,0	414,0	414,0	414,0	0,5
				1	1252,0	-	-	ZL	1252,0	142,0	142,0	142,0	0,8
				2	1254,0	81,8	25,8	ZL	1254,0	82,4	82,4	82,4	0,3
				3	1256,0	-	-	ZL	1256,0	61,2	61,2	61,2	0,3
				4	1258,0	55,2	25,2	ZL	1258,0	51,4	51,4	51,4	0,3
				5	1260,0	46,0	34,0	ZL	1260,0	48,0	48,0	48,0	0,7
2	(1, 1)	Ö	B	0	1334,0	5400,0	1863,0	ZL	1296,0	5184,0	5184,0	5184,0	0,4
				1	1334,0	-	-	ZL	1296,0	1728,0	1728,0	1728,0	0,7
				2	1348,4	1264,3	237,0	ZL	1348,4	728,5	728,5	728,5	0,3
				3	1362,9	-	-	ZL	1362,8	658,8	658,8	658,8	0,2
				4	1377,4	-	-	ZL	1377,2	611,0	611,0	611,0	0,2
				5	1391,9	576,0	385,0	ZL	1392,0	576,0	576,0	576,0	0,7
3	(1, 1)	F	K	0	1260,0	1170,0	765,0	ZL	1260,0	1134,0	1134,0	1134,0	0,4
				1	1260,0	407,7	83,6	ZL	1260,0	414,0	414,0	414,0	0,6
				2	1284,7	161,8	58,9	ZL	1266,0	234,2	234,2	234,2	0,2
				3	1309,5	120,0	87,9	ZL	1272,0	171,0	171,0	171,0	0,1
				4	1334,2	120,0	87,2	ZL	1278,0	141,8	141,8	141,8	0,2
				5	1359,0	124,0	53,0	ZL	1284,0	126,0	126,0	126,0	0,7
4	(1, 1)	F	B	0	-	-	-	ZL	1274,0	3024,0	3024,0	3024,0	0,5
				1	-	-	-	ZL	1274,0	960,0	960,0	960,0	0,9
				2	-	-	-	ZL	1287,0	570,6	570,6	570,6	0,4
				3	-	-	-	ZL	1300,0	433,5	433,5	433,5	0,4
				4	-	-	-	ZL	1313,0	370,3	370,3	370,3	0,4
				5	-	-	-	ZL	1326,0	336,0	336,0	336,0	0,5
5	(1, 2)	Ö	K	0	1252,0	-	-	ZL	1252,0	556,2	549,0	567,0	0,3
				1	1252,0	-	-	ZL	1252,0	205,8	205,0	207,0	0,5
				2	1254,0	111,8	42,9	ZL	1254,0	117,4	116,6	118,6	0,2
				3	1256,0	197,5	41,4	ZL	1256,0	85,1	84,3	86,3	0,2
				4	1258,0	128,6	43,3	ZL	1258,0	69,1	68,3	70,3	0,2
				5	1260,0	-	-	ZL	1260,0	63,0	63,0	63,0	0,7
6	(1, 2)	Ö	B	0	1296,0	-	-	ZL	1296,0	6804,0	6804,0	6804,0	0,4
				1	1296,0	2471,6	2471,6	3270,2	1296,0	2484,0	2484,0	2484,0	0,7
				2	1319,9	-	-	ZL	1319,9	1416,8	1416,8	1416,8	0,3
				3	1343,9	1003,4	1003,4	1736,5	1343,9	1046,6	1033,8	1055,0	0,3
				4	1367,9	2895,8	420,7	ZL	1367,6	843,6	843,6	843,6	0,2
				5	1391,9	-	-	ZL	1392,0	756,0	756,0	756,0	0,6
7	(1, 2)	F	K	0	1265,3	1440,0	729,0	ZL	1260,0	1458,0	1458,0	1458,0	0,5
				1	1265,3	324,6	121,2	ZL	1260,0	594,0	594,0	594,0	0,8
				2	1270,0	246,2	96,9	ZL	1270,0	254,4	254,4	254,4	0,3
				3	1274,7	196,0	128,0	ZL	1274,7	207,2	207,2	207,2	0,3
				4	1279,3	-	-	ZL	1279,3	177,1	177,1	177,1	0,3
				5	1284,0	162,0	87,0	ZL	1284,0	162,0	162,0	162,0	0,5
8	(1, 2)	F	B	0	1274,1	-	-	ZL	1274,0	3996,0	3996,0	3996,0	0,5
				1	1274,1	1444,5	258,2	ZL	1274,0	1380,0	1380,0	1380,0	1,0
				2	1287,1	810,6	315,2	ZL	1287,1	797,3	797,3	797,3	0,4
				3	1300,1	998,4	259,2	ZL	1300,1	590,5	590,5	590,5	0,4
				4	1313,1	-	-	ZL	1313,0	489,7	489,7	489,7	0,4
				5	1326,2	430,0	261,0	ZL	1326,0	444,0	444,0	444,0	0,5

Tablo 38. 4x3 veri kümesi için MISOCP HMA karşılaştırma sonuçları

Veri	(C_e, C_f)	M.Y.	M.B.	P.N.	MISOCP				HMA				
					CT	E	A.S.	Süre (sn)	CT	Ort.	Min.	Mak.	Süre (sn)
1	(1, 1)	Ö	K	0	1737,3	378,0	378,0	1416,8	1737,3	396,0	396,0	396,0	0,2
				1	1737,3	169,4	30,4	ZL	1737,3	172,0	172,0	172,0	0,4
				2	1740,0	89,7	29,5	ZL	1740,0	92,4	92,4	92,4	0,1
				3	1742,6	61,8	36,6	ZL	1742,6	64,3	64,3	64,3	0,1
				4	1745,3	86,3	31,4	ZL	1745,3	51,3	51,3	51,3	0,1
				5	1748,0	42,0	42,0	1235,4	1748,0	44,0	44,0	44,0	0,4
2	(1, 1)	Ö	B	0	1795,4	4536,0	4536,0	1824,8	1796,0	4752,0	4752,0	4752,0	0,2
				1	1795,4	2062,1	2062,1	2160,1	1796,0	2068,8	2064,0	2088,0	0,4
				2	1827,9	1103,2	418,2	ZL	1827,9	1107,8	1106,0	1115,1	0,1
				3	1859,9	751,6	421,0	ZL	1859,8	769,1	769,1	769,1	0,1
				4	1891,8	588,0	403,4	ZL	1891,6	656,0	646,4	670,4	0,1
				5	1923,8	504,0	504,0	2243,0	1924,0	528,0	528,0	528,0	0,5
3	(1, 1)	F	K	0	-	-	-	ZL	1748,0	1011,6	990,0	1044,0	0,2
				1	-	-	-	ZL	1748,0	500,0	500,0	500,0	0,3
				2	-	-	-	ZL	1756,0	256,6	254,2	260,2	0,1
				3	-	-	-	ZL	1764,0	172,4	170,0	176,0	0,0
				4	-	-	-	ZL	1772,0	133,5	131,1	137,1	0,0
				5	-	-	-	ZL	1780,0	116,0	116,0	116,0	0,4
4	(1, 1)	F	B	0	1766,7	-	-	ZL	1766,7	2649,6	2574,0	2700,0	0,2
				1	1766,7	1134,2	246,3	ZL	1766,7	1133,4	1132,0	1138,9	0,4
				2	1784,1	-	-	ZL	1784,1	604,9	596,5	610,5	0,1
				3	1801,5	-	-	ZL	1801,4	423,7	415,3	429,3	0,1
				4	1818,9	476,8	235,8	ZL	1818,8	342,6	331,4	359,4	0,1
				5	1836,2	287,0	287,0	1457,1	1836,0	300,0	300,0	300,0	0,4
5	(1, 2)	Ö	K	0	1737,3	522,0	522,0	2016,5	1737,3	540,0	540,0	540,0	0,2
				1	1737,3	250,4	44,3	ZL	1737,3	252,0	252,0	252,0	0,4
				2	1740,0	125,7	125,7	1315,0	1740,0	135,3	135,3	135,3	0,3
				3	1742,6	-	-	ZL	1742,6	91,6	91,6	91,7	0,2
				4	1745,3	68,9	48,1	ZL	1745,3	69,8	69,8	69,8	0,2
				5	1748,0	58,0	58,0	1467,4	1748,0	60,0	60,0	60,0	0,4
6	(1, 2)	Ö	B	0	1795,9	6232,0	6232,0	1413,5	1796,0	6480,0	6480,0	6480,0	0,2
				1	1795,9	3008,2	3008,2	3259,2	1796,0	3024,0	3024,0	3024,0	0,3
				2	1827,9	1522,3	1522,3	2541,5	1827,8	1618,9	1618,9	1618,9	0,2
				3	1859,9	-	-	ZL	1859,7	1171,4	1110,7	1211,9	0,1
				4	1891,8	803,5	609,3	ZL	1891,5	896,3	890,4	915,6	0,1
				5	1923,8	693,0	693,0	1938,6	1924,0	720,0	720,0	720,0	0,6
7	(1, 2)	F	K	0	-	-	-	ZL	1748,0	1396,8	1386,0	1440,0	0,2
				1	-	-	-	ZL	1748,0	736,0	736,0	736,0	0,3
				2	-	-	-	ZL	1756,0	379,6	378,4	384,4	0,1
				3	-	-	-	ZL	1764,0	250,7	249,5	255,5	0,1
				4	-	-	-	ZL	1771,9	184,1	182,9	188,9	0,0
				5	-	-	-	ZL	1780,0	160,0	160,0	160,0	0,3
8	(1, 2)	F	B	0	1777,4	3636,0	2943,0	ZL	1766,7	3585,6	3510,0	3636,0	0,3
				1	1777,4	1023,1	347,7	ZL	1766,7	1655,9	1652,0	1666,0	0,5
				2	1792,1	700,0	252,6	ZL	1792,1	721,4	718,6	723,3	0,3
				3	1806,8	538,4	331,8	ZL	1806,7	591,1	583,0	596,6	0,3
				4	1821,5	632,0	303,7	ZL	1821,3	456,3	437,7	473,1	0,2
				5	1836,2	-	-	ZL	1836,0	404,0	404,0	404,0	0,3



Şekil 23. HMA ile türetilen 10, 20 ve 50 noktalı Pareto önyüzler

önyüzde kaç tane çözüm istendiği bilgisini vererek denemeler yapılmıştır. Bu amaçla 10, 20 ve 50 çözüm çıkartılacak şekilde denemeler yapılmıştır. Yukarıda 5 çözüm için verilen tabloda, (3-3) verisi için bir çözüm için ortalama süre 0.4 sn iken, 10, 20 ve 50 için bu süre sırasıyla 0.32, 0.29 ve 0.25 sn'dir. Bu durumda, bir pareto önyüzdeki bütün noktaları türetmek için gerekli süreler sırasıyla 2, 3.2, 5.8 ve 12.5 sn'dir. Ardışık çözümler arası CT değerleri arasındaki fark azaldıkça, HMA algoritması daha hızlı çözüm üretmektedir. Ayrıca, (3-3) verilerinden ilki için bu şekilde 10, 20 ve 50 nokta türetilerek oluşturulan Pareto önyüzler Şekil 23'de gösterilmiştir. Bu şekilden de görülebileceği üzere, türetilmek istenen nokta sayısı ne olursa olsun, HMA algoritmasının türettiği çözümler Pareto önyüzü iyi bir şekilde temsil etmektedir.

5.6 Sonuç ve Değerlendirme

Bu bölümde değişik tip parça üreten tek tutuculu robotlu hücreler ele alınmış ve bu sistemlerde ortaya çıkan parça sıralaması, robot hareket sıralaması ve robot hareket hızlarının belirlenmesi problemlerinin eş zamanlı çözümü için yöntemler geliştirilmiştir. Bu problem, parça sıralamasının da belirlenmesi gerekliliği yüzünden önceki iş paketlerinde ele alınan tek tip parça üreten tek ve çift tutuculu robotlu hücrelerdeki problemlere göre daha karmaşık bir yapıdadır. Bu problem için alternatif matematiksel programlama formülasyonları geliştirilmiştir. Bunlardan iki tanesi (MINLP1, MINLP2) Karma Tamsayı Doğrusal Olmayan yapıdayken sonuncusu (MISOCP) İkinci Dereceden Konik Programlama formülasyonu şeklindedir.

Geliştirilen bütün formülasyonların türetilen örnek problemlerle doğrulamaları yapılmıştır. Ayrıca doğrusal olmayan formülasyonların ikinci dereceden konik programlama formülasyonu ile karşılaştırmaları türetilen örnek problemler üzerinden yapılmıştır. Bu testlerde MISOCP'un çözüm süresini ciddi oranda düşürdüğü görülmüştür. Doğrusal olmayan modellerle BARON çözücüsü kullanılarak çok küçük problem boyutlarında bile optimal çözümleri elde etmek mümkün olmamıştır.

MISOCP her ne kadar çözüm süresini önemli ölçüde azaltsa da problemin karmaşıklığından dolayı, problem boyutu arttıkça çözüm süresi de hızlı bir şekilde artmaktadır. Bu sebeple makul

sürelerde kaliteli çözümler elde edebilmek için sezgisel bir algoritma geliştirilmiştir. Geliştirilen bu algorithmada parça sıralamalarını belirlemek için Genetik Algoritma kullanılırken, herhangi bir parça sıralamasına karşılık gelen robot hareket sıralaması ve hareket hızlarının belirlenmesi aşamasında Tabu Arama algoritması kullanılmıştır. Tabu arama içerisinde hız belirleme aşaması için aç gözlü bir inşa ve iyileştirme algoritması kullanılmıştır. Geliştirilen bu hibrit metasezgisel algoritmanın kalibre edilmesi gereken çok sayıda parametresi bulunmaktadır. Bu amaçla, literatürde akış atölyeleri için verilmiş test verileri eksik parametreler tamamlanarak mevcut probleme uygun hale getirilmiş ve tam faktöryel deney tasarımı ile çok sayıda problem çözdürülerek parametre değerleri belirlenmiştir.

Nihai şeklini alan HMA algoritmasının performansı, MISOCP ile karşılaştırılarak test edilmiştir. Bu amaçla farklı boyutlarda test verileri kullanılmıştır. Fakat MISOCP, iş ve makine sayısının 3'ten büyük olduğu durumlarda 1 saat zaman limiti içinde çözüm verememektedir. HMA ise, saniyeler içerisinde Pareto önyüz üzerinden çok sayıda çözüm belirleyebilmektedir. HMA'nın optimal çözümde sapması ise oldukça düşük değerlerdedir ve belirlenmiş olan başarı kriterlerine uygundur. Sonuç olarak, HMA'nın problemin çözümü için etkin bir çözüm yöntemi olduğu söylenebilir.

KAYNAKLAR

- [1] Sebastian Panek, Olaf Stursberg, and Sebastian Engell. Optimization of timed automata models using mixed-integer programming. In *Formal Modeling And Analysis of Timed Systems*, pages 73–87. Springer, 2003.
- [2] Avenir Kobetski and Martin Fabian. Time-optimal coordination of flexible manufacturing systems using deterministic finite automata and mixed integer linear programming. *Discrete Event Dynamic Systems*, 19(3):287–315, 2009.
- [3] Annual survey of manufacturers. 2005.
- [4] Manufuture-strategic research agenda. *ManuFuture Platform*, 2006.
- [5] Iain MacLeay. *Digest of United Kingdom energy statistics 2010*. The Stationery Office, 2010.
- [6] Energy Efficiency. Tracking industrial energy efficiency and co2 emissions. *International Energy Agency*, 34(2):1–12, 2007.
- [7] <http://www.eia.gov/consumption/manufacturing>. visited on Dec. 2014.
- [8] A Fysikopoulos, D Anagnostakis, K Salonitis, and G Chryssolouris. An empirical study of the energy consumption in automotive assembly. *Procedia CIRP*, 3:477–482, 2012.
- [9] T Chettibi, HE Lehtihet, M Haddad, and S Hanchi. Minimum cost trajectory planning for industrial robots. *European Journal of Mechanics-A/Solids*, 23(4):703–715, 2004.
- [10] J Franke, S Kreitlein, F Risch, and S Guenther. Energy-efficient production strategies and technologies for electric drives. In *Industrial Technology (ICIT), 2013 IEEE International Conference on*, pages 1898–1903. IEEE, 2013.
- [11] Sebastian Thiede. *Energy efficiency in manufacturing systems*. Springer Science & Business Media, 2012.
- [12] M Bornschlegl, M Drechsel, S Kreitlein, and J Franke. Holistic approach to reducing co2 emissions along the energy-chain (e-chain). In *Sustainable Automotive Technologies 2013*, pages 227–234. Springer, 2014.
- [13] Fridtjof Unander. Decomposition of manufacturing energy-use in iea countries: How do recent developments compare with historical long-term trends? *Applied Energy*, 84(7): 771–780, 2007.
- [14] Jörg Engelmann. *Methoden und Werkzeuge zur Planung und Gestaltung energieeffizienter Fabriken*. IBF, 2009.
- [15] Davis Meike and Leonids Ribickis. Energy efficient use of robotics in the automobile industry. In *Advanced Robotics (ICAR), 2011 15th International Conference on*, pages 507–511. IEEE, 2011.

- [16] Anton Rassolkin, Hardi Hoimoja, and Raivo Teemets. Energy saving possibilities in the industrial robot irb 1600 control. In *2011 7th International Conference-Workshop Compatibility and Power Electronics (CPE)*, 2011.
- [17] Zdenek Kolibal and Anna Smetanova. Experimental implementation of energy optimization by robot movement. In *19th International Workshop on Robotics in Alpe-Adria-Danube Region (RAAD 2010)*, 2010.
- [18] Rolf Isermann. *Mechatronic systems: fundamentals*. Springer Science & Business Media, 2007.
- [19] Marcello Pellicciari, Angelo O Andrisano, Francesco Leali, and Alberto Vergnano. Engineering method for adaptive manufacturing systems design. *International Journal on Interactive Design and Manufacturing (IJIDeM)*, 3(2):81–91, 2009.
- [20] Suresh P Sethi, Chelliah Sriskandarajah, Gerhard Sorger, Jacek Blazewicz, and Wieslaw Kubiak. Sequencing of parts and robot moves in a robotic cell. *International Journal of Flexible Manufacturing Systems*, 4(3 – 4):331–358, 1992.
- [21] Yves Crama and Joris Van De Klundert. Cyclic scheduling of identical parts in a robotic cell. *Operations Research*, 45(6):952–965, 1997.
- [22] Milind Dawande, Chelliah Sriskandarajah, and Suresh Sethi. On throughput maximization in constant travel-time robotic cells. *Manufacturing & Service Operations Management*, 4(4):296–312, 2002.
- [23] Leon van Os. The power of cyclic production. <https://www.involvation.nl/en/articles/power-cyclic-production-2/>. Accessed: 2016-08-18.
- [24] IG Drobouchevitch, S Sethi, J Sidney, and Chelliah Sriskandarajah. 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, 2004.
- [25] R Neugebauer, M Putz, J Böhme, M Todtermuschke, and M Pfeifer. New aspects of energy consumption analysis in assembly processes and equipment. In *Sustainable Manufacturing*, pages 197–201. Springer, 2012.
- [26] A Barili, M Ceresa, and C Parisi. Energy-saving motion control for an autonomous mobile robot. In *Industrial Electronics, 1995. ISIE'95., Proceedings of the IEEE International Symposium on*, volume 2, pages 674–676. IEEE, 1995.
- [27] Milind W Dawande, H Neil Geismar, Suresh P Sethi, and Chelliah Sriskandarajah. *Throughput optimization in robotic cells*, volume 101. Springer Science & Business Media, 2007.

- [28] Alessandro Agnetis. Scheduling no-wait robotic cells with two and three machines. *European Journal of Operational Research*, 123(2):303–314, 2000.
- [29] Alessandro Agnetis and Dario Pacciarelli. Part sequencing in three-machine no-wait robotic cells. *Operations Research Letters*, 27(4):185–192, 2000.
- [30] Ada Che, Chengbin Chu, and Eugene Levner. A polynomial algorithm for 2-degree cyclic robot scheduling. *European Journal of Operational Research*, 145(1):31–44, 2003.
- [31] Nicholas G Hall and Chelliah Sriskandarajah. A survey of machine scheduling problems with blocking and no-wait in process. *Operations research*, 44(3):510–525, 1996.
- [32] Vladimir Kats and Eugene Levner. Cyclic scheduling in a robotic production line. *Journal of Scheduling*, 5(1):23–41, 2002.
- [33] Eugene Levner, Vladimir Kats, and Vadim E Levit. An improved algorithm for cyclic flowshop scheduling in a robotic cell. *European Journal of Operational Research*, 97(3):500–508, 1997.
- [34] William D Hitz and Cecil R Stewart. Oxygen and carbon dioxide effects on the pool size of some photosynthetic and photorespiratory intermediates in soybean (glycine max [L.] merr.). *Plant physiology*, 65(3):442–446, 1980.
- [35] Yves Crama, Vladimir Kats, J Van de Klundert, and Eugene Levner. Cyclic scheduling in robotic flowshops. *Annals of operations Research*, 96(1-4):97–124, 2000.
- [36] Milind Dawande, H Neil Geismar, Suresh P Sethi, and Chelliah Sriskandarajah. Sequencing and scheduling in robotic cells: Recent developments. *Journal of Scheduling*, 8(5):387–426, 2005.
- [37] W Baumann, R Birner, J Haeusler, and R-P Hartmann. Operating and idle times for cyclic multi-machine servicing. *Industrial Robot: An International Journal*, 8(1):44–49, 1981.
- [38] DJ Medeiros, EF Watson, JS Carson, and MS Manivannan. Operational modeling & simulation in semiconductor manufacturing. In *Proceedings of the 1998 Winter Simulation Conference*, 1998.
- [39] SY Nof and D Hanna. Operational characteristics of multi-robot systems with cooperation. *THE INTERNATIONAL JOURNAL OF PRODUCTION RESEARCH*, 27(3):477–492, 1989.
- [40] Anthony S Kondoleon. *Cycle time analysis of robot assembly systems*. Society of Manufacturing Engineers, 1979.
- [41] BH Claybourn. Scheduling robots in flexible manufacturing cells. *Chartered Mechanical Engineer*, 30:36–40, 1983.
- [42] Jacek Blazewicz, Suresh P Sethi, and Chelliah Sriskandarajah. *Scheduling of robot moves and parts in a robotic cell*. École des hautes études commerciales, 1989.

- [43] R Logendran and C Sriskandarajah. Sequencing of robot activities and parts in two-machine robotic cells. *International Journal of production research*, 34(12):3447–3463, 1996.
- [44] N Brauner and G Finke. Final results on the one-cycle conjecture in robotic cells. *Internal note, Laboratoire LEIBNIZ, Institut IMAG, Grenoble, France*, 1997.
- [45] Nadia Brauner and Gerd Finke. On a conjecture about robotic cells: new simplified proof for the three-machine case. *Infor-Information Systems and Operational Research*, 37(1): 20–36, 1999.
- [46] N Brauner and G Finke. Optimal moves of the material handling system in a robotic flow-shop. In *Proceedings IEPM*, volume 99, pages 409–417, 1999.
- [47] Nadia Brauner and Gerd Finke. Cycles and permutations in robotic cells. *Mathematical and Computer Modelling*, 34(5):565–591, 2001.
- [48] Nadia Brauner, Gerd Finke, and Wieslaw Kubiak. Complexity of one-cycle robotic flow-shops. *Journal of Scheduling*, 6(4):355–372, 2003.
- [49] Nicholas G Hall, Hichem Kamoun, and Chelliah Sriskandarajah. Scheduling in robotic cells: Classification, two and three machine cells. *Operations Research*, 45(3):421–439, 1997.
- [50] Nicholas G Hall, Hichem Kamoun, and Chelliah Sriskandarajah. Scheduling in robotic cells: Complexity and steady state analysis. *European Journal of Operational Research*, 109(1):43–65, 1998.
- [51] Chelliah Sriskandarajah, Nicholas G Hall, and Hichem Kamoun. Scheduling large robotic cells without buffers. *Annals of Operations Research*, 76:287–321, 1998.
- [52] M Selim Akturk, Hakan Gultekin, and Oya Ekin Karasan. Robotic cell scheduling with operational flexibility. *Discrete Applied Mathematics*, 145(3):334–348, 2005.
- [53] Hakan Gultekin, M Selim Akturk, and Oya Ekin Karasan. Cyclic scheduling of a 2-machine robotic cell with tooling constraints. *European Journal of Operational Research*, 174(2):777–796, 2006.
- [54] Lei Lei and Tzyh-Jong Wang. Determining optimal cyclic hoist schedules in a single-hoist electroplating line. *IIE transactions*, 26(2):25–33, 1994.
- [55] Haoxun Chen, Chengbin Chu, and Jean-Marie Proth. Cyclic scheduling of a hoist with time window constraints. *Robotics and Automation, IEEE Transactions on*, 14(1):144–152, 1998.
- [56] Ada Che, Chengbin Chu, and Feng Chu. Multicyclic hoist scheduling with constant processing times. *Robotics and Automation, IEEE Transactions on*, 18(1):69–80, 2002.

- [57] Raza Ur-Rehman, Stéphane Caro, Damien Chablat, and Philippe Wenger. Path placement optimization of manipulators based on energy consumption: application to the orthoglide 3-axis. *arXiv preprint arXiv:0910.4000*, 2009.
- [58] Cory Bryan, Mitch Grenwalt, and Adam Stienecker. Energy consumption reduction in industrial robots. In *Proceedings ASEE North Central Sectional Conference*, 2010.
- [59] A Smetanová. Optimization of energy by robot movement. *Modern Machinery Science Journal*, 3(1):172–176, 2010.
- [60] Davis Meike and Leonids Ribickis. Analysis of the energy efficient usage methods of medium and high payload industrial robots in the automobile industry. In *10th International Symposium "Topical Problems in the Field of Electrical and Power Engineering" Pärnu, Estonia*, 2011.
- [61] M Pellicciari, G Berselli, F Leali, and A Vergnano. A method for reducing the energy consumption of pick-and-place industrial robots. *Mechatronics*, 23(3):326–334, 2013.
- [62] Alberto Vergnano, Carl Thorstensson, Bengt Lennartson, Petter Falkman, Marcello Pellicciari, Francesco Leali, and Stephan Biller. Modeling and optimization of energy consumption in cooperative multi-robot systems. *Automation Science and Engineering, IEEE Transactions on*, 9(2):423–428, 2012.
- [63] Avenir Kobetski and Martin Fabian. Velocity balancing in flexible manufacturing systems. In *Discrete Event Systems, 2008. WODES 2008. 9th International Workshop on*, pages 358–363. IEEE, 2008.
- [64] Kai Li, Xun Zhang, Joseph Y-T Leung, and Shan-Lin Yang. Parallel machine scheduling problems in green manufacturing industry. *Journal of Manufacturing Systems*, 38:98–106, 2016.
- [65] Yi-Chi Wang, Ming-Jun Wang, and Sung-Chi Lin. Selection of cutting conditions for power constrained parallel machine scheduling. *Robotics and Computer-Integrated Manufacturing*, 2015.
- [66] Min Ji, Jen-Ya Wang, and Wen-Chiung Lee. Minimizing resource consumption on uniform parallel machines with a bound on makespan. *Computers & Operations Research*, 40(12):2970–2974, 2013.
- [67] Dvir Shabtay and Moshe Kaspi. Parallel machine scheduling with a convex resource consumption function. *European Journal of Operational Research*, 173(1):92–107, 2006.
- [68] S Afshin Mansouri, Emel Aktas, and Umut Besikci. Green scheduling of a two-machine flowshop: Trade-off between makespan and energy consumption. *European Journal of Operational Research*, 248(3):772–788, 2016.
- [69] Jian-Ya Ding, Shiji Song, and Cheng Wu. Carbon-efficient scheduling of flow shops by multi-objective optimization. *European Journal of Operational Research*, 248(3):758–771, 2016.

- [70] Wenwen Lin, DY Yu, Chaoyong Zhang, Xun Liu, Sanqiang Zhang, Yuhui Tian, Shengqiang Liu, and Zhanpeng Xie. A multi-objective teaching- learning-based optimization algorithm to scheduling in turning processes for minimizing makespan and carbon footprint. *Journal of Cleaner Production*, 101:337–347, 2015.
- [71] Dunbing Tang, Min Dai, Miguel A Salido, and Adriana Giret. Energy-efficient dynamic scheduling for a flexible flow shop using an improved particle swarm optimization. *Computers in Industry*, 81:82–95, 2016.
- [72] Min Dai, Dunbing Tang, Adriana Giret, Miguel A Salido, and Wei Dong Li. Energy-efficient scheduling for a flexible flow shop using an improved genetic-simulated annealing algorithm. *Robotics and Computer-Integrated Manufacturing*, 29(5):418–429, 2013.
- [73] Hao Luo, Bing Du, George Q Huang, Huaping Chen, and Xiaolin Li. Hybrid flow shop scheduling considering machine electricity consumption cost. *International Journal of Production Economics*, 146(2):423–439, 2013.
- [74] AAG Bruzzone, D Anghinolfi, M Paolucci, and F Tonelli. Energy-aware scheduling for improving manufacturing process sustainability: a mathematical model for flexible flow shops. *CIRP Annals-Manufacturing Technology*, 61(1):459–462, 2012.
- [75] Christian Gahm, Florian Denz, Martin Dirr, and Axel Tuma. Energy-efficient scheduling in manufacturing companies: a review and research framework. *European Journal of Operational Research*, 248(3):744–757, 2016.
- [76] Adriana Giret, Damien Trentesaux, and Vittal Prabhu. Sustainability in manufacturing operations scheduling: a state of the art review. *Journal of Manufacturing Systems*, 37: 126–140, 2015.
- [77] Kuei-Tang Fang and Bertrand MT Lin. Parallel-machine scheduling to minimize tardiness penalty and power cost. *Computers & Industrial Engineering*, 64(1):224–234, 2013.
- [78] Bing Du, Huaping Chen, George Q Huang, and HD Yang. Preference vector ant colony system for minimising make-span and energy consumption in a hybrid flow shop. In *Multi-objective Evolutionary Optimisation for Product Design and Manufacturing*, pages 279–304. Springer, 2011.
- [79] Min Dai, Dunbing Tang, Yuchun Xu, and Weidong Li. Energy-aware integrated process planning and scheduling for job shops. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, 229(1 suppl):13–26, 2015.
- [80] Corinne Subai, Pierre Baptiste, and Eric Niel. Scheduling issues for environmentally responsible manufacturing: The case of hoist scheduling in an electroplating line. *International Journal of Production Economics*, 99(1):74–87, 2006.
- [81] Liping Zhang, Xinyu Li, Liang Gao, Guohui Zhang, and Xiaoyu Wen. Dynamic scheduling model in fms by considering energy consumption and schedule efficiency. In *Computer*

Supported Cooperative Work in Design (CSCWD), 2012 IEEE 16th International Conference on, pages 719–724. IEEE, 2012.

- [82] Gilles Mouzon and Mehmet B Yildirim. A framework to minimise total energy consumption and total tardiness on a single machine. *International Journal of Sustainable Engineering*, 1(2):105–116, 2008.
- [83] Mehmet Bayram Yildirim and Gilles Mouzon. Single-machine sustainable production planning to minimize total energy consumption and total completion time using a multiple objective genetic algorithm. *IEEE transactions on engineering management*, 59(4):585–597, 2012.
- [84] ChenGuang Liu, Jing Yang, Jie Lian, WenJuan Li, Steve Evans, and Yong Yin. Sustainable performance oriented operational decision-making of single machine systems with deterministic product arrival time. *Journal of Cleaner Production*, 85:318–330, 2014.
- [85] Hakan Gultekin, M Selim Akturk, and Oya Ekin Karasan. Bicriteria robotic cell scheduling. *Journal of Scheduling*, 11(6):457–473, 2008.
- [86] Hakan Gultekin, M Selim Akturk, and Oya Ekin Karasan. Bicriteria robotic operation allocation in a flexible manufacturing cell. *Computers & operations research*, 37(4):779–789, 2010.
- [87] M Selim Akturk and Taylan Ilhan. Single cnc machine scheduling with controllable processing times to minimize total weighted tardiness. *Computers & Operations Research*, 38(4):771–781, 2011.
- [88] Zeynep Uruk, Hakan Gultekin, and M Selim Akturk. Two-machine flowshop scheduling with flexible operations and controllable processing times. *Computers & Operations Research*, 40(2):639–653, 2013.
- [89] Yongguo Mei, Yung-Hsiang Lu, Y Charlie Hu, and CS George Lee. Energy-efficient motion planning for mobile robots. In *Robotics and Automation, 2004. Proceedings. ICRA'04. 2004 IEEE International Conference on*, volume 5, pages 4344–4349. IEEE, 2004.
- [90] John A Broderick. *Energy and Mobility Management of a Ground Robot to Increase Operational Capacity*. PhD thesis, University of Michigan, 2015.
- [91] M. Chemnitz, G. Schreck, and J. Krüger. Analyzing energy consumption of industrial robots. In *ETFA2011*, pages 1–4, 2011. doi: 10.1109/ETFA.2011.6059221.
- [92] Yves Crama and Joris Van de Klundert. Cyclic scheduling in 3-machine robotic flow shops. *Journal of Scheduling*, 2(1):35–54, 1999. doi: 10.1002/(SICI)1099-1425(199901/02)2:1<35::AID-JOS15>3.0.CO;2-J.

- [93] S. P. Sethi, J. B. Sidney, and C. Sriskandarajah. Scheduling in dual gripper robotic cells for productivity gains. *IEEE Transactions on Robotics and Automation*, 17(3):324–341, 2001.
- [94] IG Drobouchevitch, S Sethi, J Sidney, and Chelliah Sriskandarajah. Scheduling dual gripper robotic cell: One-unit cycles. *European Journal of Operational Research*, 171(2): 598–631, 2006.
- [95] H Geismar, M Pinedo, and Chelliah Sriskandarajah. Robotic cells with parallel machines and multiple dual gripper robots: a comparative overview. *IIE Transactions*, 40(12):297–314, 2008.
- [96] Chelliah Sriskandarajah, IG Drobouchevitch, S Sethi, and R Chandrasekaran. Scheduling multiple parts in a robotic cell served by a dual-gripper robot. *Operations Research*, 52(1):65–82, 2004.
- [97] H Geismar, Dawande M, and C Sriskandarajah. Throughput optimization in constant travel-time dual gripper robotic cells with parallel machines. *Production and Operations Management*, 14(2):311–328, 2006.
- [98] M Foumani and K Jenab. Cycle time analysis in reentrant robotic cells with swap ability. *International Journal of Production Research*, 50(22):6372–6387, 2012.
- [99] TE Lee. A review of scheduling theory and methods for semiconductor manufacturing cluster tools. In *Proceedings of the 40th Conference on Winter Simulation, Miami, Florida*, 2008.
- [100] H Geismar, U.V Manoj, A Sethi, C Sriskandarajah, and N Ramanan. Scheduling robotic cells served by a dual-arm robot. *IIE Transactions*, 44(3):230–248, 2012.
- [101] N Hall, C Potts, and Chelliah Sriskandarajah. Parallel machine scheduling with a common server. *Discrete Applied Mathematics*, 102(3):223–243, 2000.
- [102] H Geismar, C Sriskandarajah, and N Ramanan. Increasing throughput for robotic cells with parallel machines and multiple robots. *IEEE Transactions on Automation Science and Engineering*, 1(1):84–89, 2004.
- [103] E Gundogdu and H Gultekin. Scheduling in two-machine robotic cells with a self-buffered robot. *IEEE Transactions*, 48(2):170–191, 2015.
- [104] IG Drobouchevitch, N Geismar, and Chelliah Sriskandarajah. 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, 2010.
- [105] V Kats and Levner. A strongly polynomial algorithm for no-wait cyclic robotic flowshop scheduling. *Operations Research Letters*, 21(4):171–179, 1997.
- [106] V Kats and Levner. Minimizing the number of vehicles in periodic scheduling: The non-euclidean case. *European Journal of Operational Research*, 107(2):371–377, 1998.

- [107] Umit Bilge and Gunduz Ulusoy. A time window approach to simultaneous scheduling of machines and material handling system in an fms. *Operations Research*, 43(6):1058 – 1070, 1995.
- [108] Y.P. Aneja and H. Kamoun. Scheduling of parts and robot activities in a two machine robotic cell. *Computers & Operations Research*, 26(4):297 – 312, 1999.
- [109] Johann Hurink and Sigrid Knust. A tabu search algorithm for scheduling a single robot in a job-shop environment. *Discrete Applied Mathematics*, 119(1):181 – 203, 2002.
- [110] A. Soukhal and P. Martineau. Resolution of a scheduling problem in a flowshop robotic cell. *European Journal of Operational Research*, 161(1):62 – 72, 2005.
- [111] Jacques Carlier, Mohamed Haouari, Mohamed Kharbeche, and Aziz Moukrim. An optimization-based heuristic for the robotic cell problem. *European Journal of Operational Research*, 202(3):636 – 645, 2010.
- [112] Wassim Zahrouni and Hichem Kamoun. Sequencing and scheduling in a three-machine robotic cell. *International Journal of Production Research*, 50(10):2823 – 2835, 2012.
- [113] Muhammad Nawaz, E Emory Enscore, and Inyong Ham. Sequencing and scheduling in a three-machine robotic cell. *Omega*, 11(1):91 – 95, 1983.
- [114] G. Didem Batur, Oya Ekin Karasan, and M. Selim Akturk. Multiple part-type scheduling in flexible robotic cells. *International Journal of Production Economics*, pages 726 –740, 2012.
- [115] M.H. Fazel Zarandi, H. Mosadegh, and M. Fattahi. Two-machine robotic cell scheduling problem with sequence-dependent setup times. *Computers & Operations Research*, 40(5):1420 – 1434, 2013.
- [116] Atabak Elmi and Seyda Topaloglu. Scheduling multiple parts in hybrid flow shop robotic cells served by a single robot. *International Journal of Computer Integrated Manufacturing*, 27(12):1144 – 1159, 2014.
- [117] Hakan Gultekin, Betul Coban, and Vahid Eghbal Akhlaghi. Cyclic scheduling of parts and robot moves in m-machine robotic cells. *Computers & Operations Research*, 90:161 – 172, 2018. doi: <https://doi.org/10.1016/j.cor.2017.09.018>.
- [118] Hiroshi Kise. On an automated two-machine flowshop scheduling problem with infinite buffer. *Journal of the Operations Research Society of Japan*, 34(3):354 –361, 1991.
- [119] Helman I. Stern and Gad Vitner. Scheduling parts in a combined production-transportation work cell. *The Journal of the Operational Research Society*, 41(7):625 – 632, 1990.
- [120] I. N. Kamalabadi, S. Gholami, and A. H. Mirzaei. Considering a cyclic multiple-part type three-machine robotic cell problem. In *2007 IEEE International Conference on Industrial Engineering and Engineering Management*, pages 704 – 708, 2007.

- [121] M. Fathian, I. N. Kamalabad, M. Heydari, and H. Farughi. A petri net model for part sequencing and robot moves sequence in a 2-machine robotic cell. *The Journal of Software Engineering and Applications*, 04(11):603 – 608, 2011.
- [122] Jacques Carlier. Ordonnements à contraintes disjonctives. *RAIRO - Operations Research - Recherche Opérationnelle*, 12(4):333–350, 1978.

TÜBİTAK
PROJE ÖZET BİLGİ FORMU

Proje Yürütücüsü:	Doç. Dr. HAKAN GÜLTEKİN
Proje No:	215M845
Proje Başlığı:	Yeşil Robotik Hücre Çizelgeleme
Proje Türü:	1001 - Araştırma
Proje Süresi:	24
Araştırmacılar:	SİNAN GÜREL
Danışmanlar:	
Projenin Yürütüldüğü Kuruluş ve Adresi:	TOBB EKONOMİ VE TEKNOLOJİ Ü. MÜHENDİSLİK F. ENDÜSTRİ MÜHENDİSLİĞİ B.
Projenin Başlangıç ve Bitiş Tarihleri:	15/04/2016 - 15/04/2018
Onaylanan Bütçe:	243750.0
Harcanan Bütçe:	162908.33
Öz:	<p>Bu çalışmada endüstride kullanımı hızlı bir şekilde yaygınlaşan, buna paralel olarak da üzerine yapılan akademik çalışma sayısı aynı hızda artan robotik hücreler konu alınmıştır. Belirli sayıda makine ve bu makinelere hizmet veren bir malzeme elleçleme robotundan oluşan seri üretim sistemlerine robotik hücre adı verilmektedir. Sistemdeki robot üretilecek parçalar üzerinde herhangi bir işlem yapmamakta, makine yükleme/boşaltma ve makineler arası parça transferini gerçekleştirmektedir.</p> <p>Literatürdeki çalışmaların neredeyse tamamı üretim hızının maksimize edilmesini tek amaç olarak ele almaktadır. Bu alandaki çalışmaların tamamında robotun yaptığı bütün hareketleri mümkün olan en yüksek hızda gerçekleştirdiği varsayılmaktadır. Dolayısıyla robot hareketleriyle ilgili süreler sabit birer problem parametresidir. Halbuki robotların enerji tüketimleri hareket hızlarına bağlıdır ve yüksek hızda yapılan hareketler yüksek enerji tüketimine sebep olmaktadır. Diğer taraftan, hareketlerin en yüksek hızda yapılması bazı anlarda robotun bir sonraki hareket başlayana kadar boşta beklemesine sebep olmaktadır.</p> <p>Bu projede parça sıralaması, robot hareket sıralaması ve robot hareket hızlarının belirlenmesi problemleri beraberce ele alınmıştır. Proje kapsamında, i) Tek tip parça üreten tek tutuculu sistemler; ii) Tek tip parça üreten çift tutuculu sistemler ve iii) Farklı tip parça üreten tek tutuculu sistemler ayrı ayrı ele alınmıştır. Ele alınan sistemler için üretim hızı maksimizasyonu ve enerji tüketimi minimizasyonu beraberce ele alınmıştır. Dolayısıyla, her problem için 2-kriterli bir optimizasyon modeli kurulmuş ve başatlanmayan (etkin, nondominated) çözümler kümesinin belirlenmesi hedeflenmiştir.</p> <p>Tek tutuculu tek tip parça üreten 2 makineli sistemler için analitik bazı sonuçlar elde edilebilmiştir. Fakat daha çok makineli sistemler veya diğer hücre kombinasyonlarında karmaşıklık hızlı bir şekilde arttığı için analitik çözüm bulunması mümkün olmamıştır. Bu problemler için matematiksel modeller geliştirilmiştir. Karma Tamsayılı Doğrusal Olmayan yapıdaki bu modeller, çözüm süresini iyileştirmek için ikinci dereceden konik programlama formülasyonları olarak yeniden modellenmiştir. Geliştirilen bütün modeller çözdürülen örnek problemlerle doğrulanmıştır.</p> <p>İkinci dereceden konik modellerin çözüm sürelerini kısalttığı gözlenmekle beraber, problem büyüklüğü arttıkça bu modellerde de makul sürelerde çözümlere ulaşmak mümkün olmamıştır. Bu sebeple, ele alınan her problem tipi için makul sürelerde kaliteli çözümler veren sezgisel/metasezgisel yöntemler geliştirilmiştir. Geliştirilen sezgiseller C++ ve Java dillerinden kodlanmış ve yapılan denemelerle doğrulanmış ve kapsamlı deneysel çalışmalarla performans testleri yapılmıştır. Ayrıca, daha maliyetli olan fakat üretim hızını artırma kapasitesi daha yüksek olan çift tutuculu robotlarla tek tutuculu robotlar enerji tüketimleri ve üretim hızları açılarından birbirleriyle karşılaştırılmıştır.</p>
Anahtar Kelimeler:	Robotik hücre, çizelgeleme, üretimde enerji optimizasyonu, matematiksel programlama, sezgiseller

Fikri Ürün Bildirim Formu Sunuldu Mu?:	Hayır
Projeden Yapılan Yayınlar:	<p>1- Energy Conscious Robot Scheduling in Robotic Cell Manufacturing (Bildiri - Uluslararası Bildiri - Sözlü Sunum),</p> <p>2- Robotic Cell Scheduling Considering Energy Consumption of Robot Moves (Bildiri - Uluslararası Bildiri - Sözlü Sunum),</p> <p>3- Energy Aware Scheduling of the Material Handling Robot in M-Machine Robotic Cells (Bildiri - Uluslararası Bildiri - Sözlü Sunum),</p> <p>4- Energy conscious scheduling of robotmoves in dual-gripper robotic cells (Bildiri - Uluslararası Bildiri - Sözlü Sunum),</p> <p>5- Tek Tutuculu Robotik Hücrelerde Enerji Tüketimi Üretim Hızı Ödünleşimi (Bildiri - Ulusal Bildiri - Sözlü Sunum),</p> <p>6- Çift Tutuculu Robotlu Hücrelerde Robot Hareket Hızlarının Ve Sıralamalarının Belirlenmesi (Bildiri - Ulusal Bildiri - Sözlü Sunum),</p> <p>7- Green Scheduling in a Two-Machine Robotic Cell (Bildiri - Ulusal Bildiri - Sözlü Sunum),</p> <p>8- ENERGY CONSCIOUS SCHEDULING IN A TWO-MACHINE ROBOTIC CELL (Tez (Araştırmacı Yetiştirilmesi) - Yüksek Lisans Tezi),</p>

TÜBİTAK