

**ESNEK ROBOTİK HÜCRELERİN SÜREÇ ODAKLI
PETRİ AĞLARIYLA MODELLENMESİ**

**MODELING OF FLEXIBLE ROBOTIC CELLS USING
PROCESS ORIENTED PETRI NETS**

AYŞEGÜL ŞAHİN

Doktor Öğretim Üyesi Güldal GÜLERYÜZ

Tez Danışmanı

Hacettepe Üniversitesi

Lisansüstü Eğitim-Öğretim ve Sınav Yönetmeliğinin

Endüstri Mühendisliği Anabilim Dalı için Öngördüğü

YÜKSEK LİSANS TEZİ


olarak hazırlanmıştır.

2019

Ayşegül Şahin'in hazırladığı "Esnek Robotik Hücrelerin Süreç Odaklı Petri Ağlarıyla Modellenmesi" adlı bu çalışma aşağıdaki jüri tarafından ENDÜSTRİ MÜHENDİSLİĞİ ANABİLİM DALI'nda YÜKSEK LİSANS TEZİ olarak kabul edilmiştir.

Dr. Öğr. Üyesi Pelin TOKTAŞ

Başkan



Dr. Öğr. Üyesi Güldal Güleriyüz

Danışman



Prof. Dr. Murat Caner TESTİK

Üye



Dr. Öğr. Üyesi Banu Yüksel ÖZKAYA

Üye



Dr. Öğr. Üyesi Barbaros YET

Üye



Bu tez Hacettepe Üniversitesi Fen Bilimleri Enstitüsü tarafından YÜKSEK LİSANS TEZİ olarak tarihinde onaylanmıştır.

Prof. Dr. Menemşe GÜMÜŞDERELİOĞLU

Fen Bilimleri Enstitüsü Müdürü

CANIM AILEME

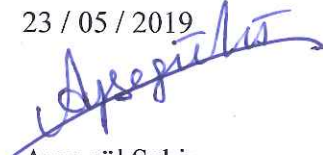
ETİK

Hacettepe Üniversitesi Fen Bilimleri Enstitüsü, tez yazım kurallarına uygun olarak hazırladığım bu tez çalışmada,

- tez içindeki bütün bilgi ve belgeleri akademik kurallar çerçevesinde elde ettiğimi,
- görsel, işitsel ve yazılı tüm bilgi ve sonuçları bilimsel ahlak kurallarına uygun olarak sunduğumu,
- başkalarının eserlerinden yararlanılması durumunda ilgili eserlere bilimsel normlara uygun olarak atıfta bulunduğumu,
- atıfta bulunduğum eserlerin tümünü kaynak olarak gösterdiğimi,
- kullanılan verilerde herhangi bir değişiklik yapmadığımı,
- ve bu tezin herhangi bir bölümünü bu üniversite veya başka bir üniversitede başka bir tez çalışması olarak sunmadığımı

beyan ederim.

23 / 05 / 2019


Ayşegül Şahin

YAYINLANMA FİKRİ MÜLKİYET HAKKLARI BEYANI

Enstitü tarafından onaylanan lisansüstü tezimin/raporumun tamamını veya herhangi bir kısmını, basılı (kâğıt) ve elektronik formatta arşivleme ve aşağıda verilen koşullarla kullanıma açma iznini Hacettepe üniversitesine verdiğimi bildiririm. Bu izinle Üniversiteye verilen kullanım hakları dışındaki tüm fikri mülkiyet haklarım bende kalacak, tezimin tamamının ya da bir bölümünün gelecekteki çalışmalarda (makale, kitap, lisans ve patent vb.) kullanım hakları bana ait olacaktır.

Tezin kendi orijinal çalışmam olduğunu, başkalarının haklarını ihlal etmediğimi ve tezimin tek yetkili sahibi olduğumu beyan ve taahhüt ederim. Tezimde yer alan telif hakkı bulunan ve sahiplerinden yazılı izin alınarak kullanması zorunlu metinlerin yazılı izin alarak kullandığımı ve istenildiğinde suretlerini Üniversiteye teslim etmeyi taahhüt ederim.

Yükseköğretim Kurulu tarafından yayınlanan “*Lisansüstü Tezlerin Elektronik Ortamda Toplanması, Düzenlenmesi ve Erişime Açılmasına İlişkin Yönerge*” kapsamında tezim aşağıda belirtilen koşullar haricince YÖK Ulusal Tez Merkezi / H. Ü. Kütüphaneleri Açık Erişim Sisteminde erişime açılır.

- Enstitü / Fakülte yönetim kurulu kararı ile tezimin erişime açılması mezuniyet tarihimden itibaren 2 yıl ertelenmiştir.
- Enstitü / Fakülte yönetim kurulu gerekçeli kararı ile tezimin erişime açılması mezuniyet tarihimden itibaren ay ertelenmiştir.
- Tezim ile ilgili gizlilik kararı verilmiştir.

24/06/2019
Aysegül Şahin
AYŞEGÜL ŞAHİN

ÖZET

ESNEK ROBOTİK HÜCRELERİN SÜREÇ ODAKLI PETRİ AĞLARIYLA MODELLENMESİ

Ayşegül ŞAHİN

Yüksek Lisans, Endüstri Mühendisliği Bölümü

Tez Danışmanı: Doktor Öğretim Üyesi Güldal GÜLERYÜZ

Mayıs 2019, 66 sayfa

Günümüzde rekabet gittikçe artarken, firmaların talepleri daha kısa zamanda karşılayabilmesi, ürünleri daha kısa sürede ve çeşitlilikte üretebilmesi oldukça önem kazanmıştır. Fabrikalarda, artan çeşitlilikteki müşteri taleplerini zamanında karşılayabilmek için verimlilik ve üretkenliği artırmak amacıyla esnek üretim sistemleri kullanılmaktadır. Esnek üretim sistemleri, yoğun otomasyon ve teknoloji ağırlıklı üretimin yapıldığı, birden fazla tipteki parçayı küçük ya da orta hacimde etkin bir biçimde üretebilmek için tasarlanmış, bilgisayar destekli makinelerden ve bunların işleyişini kontrol eden bilgisayarlardan oluşan sistemlerdir. Müşteri odaklı üretim yaparak ürün çeşidi talebindeki değişimlere hızlı ayak uydurmayı hedefler. Endüstri 4.0 ile tesislerin bireysel esnekliğini ve üretkenliğini artırması daha büyük önem kazanmışken, üretim sistemlerinde esnek robotik hücreler baskın hale gelmeye başlayacaktır. Esnek robotik hücreler, robot ve makinelerden oluşan, bir robotun makinelere yükleme boşaltma işlemlerini gerçekleştirdiği, girdi ve çıktı stoklama alanları barındıran daha esnek ve çevik üretim sistemleridir. Robotik hücrelerde bir parçanın ihtiyaç duyduğu işlemler farklı

makinelere tarafından yapılırken, esnek robotik hücrelerde parçanın kalan tüm işlemleri tek bir makine tarafından yapılabilir. Esnek robotik hücrelerde, ara stoklama alanlarının varlığı ve robotun kısıkaç sayısı sistemin üretkenliğini etkileyen önemli faktörlerdendir. Makinelerin ve robotun boş kalma zamanlarını azaltarak en kısa zamanda en büyük çıktı miktarını almak için esnek üretim hücrelerinde robotun hareketlerinin sıralaması büyük önem taşır. Literatürde, bu konuda yapılan matematiksel çalışmalar mevcuttur. Teorik olarak optimum sıralamaları bulmak bazen mümkün olmuşken bazen optimum sonuçlara ancak yaklaşılmıştır. Matematiksel modellerle teorik olarak ulaşılmış çözümler de gerçek hayatta uygulanamayabilir, çalışmayabilir. Bu nedenle teorik olarak elde edilmiş çözümlerin gerçekte çalışıp çalışmadığı petri ağlarından yararlanarak araştırılabilir. Petri ağları sistemdeki faaliyetleri temsil etmek için kullanılacak güçlü, grafiksel bir dildir. Kaynak odaklı, süreç odaklı, renkli petri ağları gibi çeşitli modelleme yöntemleri mevcuttur. Bu çalışmada, süreç odaklı petri ağlarıyla esnek robotik hücrelerin modellenmesi gösterilmektedir. Çalışmanın sonunda ara stok alanı kullanılan bir hücreyle ara stok alanı olmayıp robotun çift kısıkaça sahip olduğu iki hücrenin petri ağı modelleri kurularak üretkenlikleri karşılaştırılmaktadır. Sonuçta, ara stoklama alanlarına sahip hücrenin makine işleme süresi 120 saniye olduğunda verilen sıralama için petri ağı modelinin çalışmadığı, çift kısıkaçlı robotun hücrenin 30 saniye işleme süresi için daha verimli sonuçlar verdiği görülmüştür.

Anahtar Kelimeler: Esnek üretim sistemleri, Esnek robotik hücre, Petri ağı, Ara stok alanı, Üretkenlik, Çift kısıkaçlı robot, Süreç odaklı petri ağı

ABSTRACT

MODELING OF FLEXIBLE ROBOTIC CELLS USING PROCESS ORIENTED PETRI NETS

Ayşegül ŞAHİN

Master's Degree, Department of Industrial Engineering

Supervisor: Doctoral Professor Güldal GÜLERYÜZ

May 2019, 66 pages

Nowadays, as the competition is getting increasing, it is very important for the companies to meet the demands in a shorter time and to produce the products in a shorter time and variety. Flexible manufacturing systems are used in factories to increase productivity and efficiency in order to meet increasing customer demands on time. Flexible manufacturing systems, systems in intensive automation and technology-based production made, which is designed to produce more than one type of part in small or medium volume effectively, is composed of computer-aided machines and computers that control machines' operation. By making customer-oriented production it aims to keep pace with the changes in the demand for product types. With Industry 4.0, while it becomes more important for

plants to increase their individual flexibility and productivity, in production systems flexible robotic cells will become dominant. Flexible robotic cells are a more flexible and agile production systems, consisting of robots and machines, where a robot performs loading and unloading operations on machines, with input and output storage areas. In robotic cells, the processes required by a part are performed by different machines, while in the flexible robotic cells, all the processes of the part can be done by a single machine. In flexible robotic cells, the presence of buffer and the number of grippers of the robot are important factors affecting the productivity of the system. In order to get the maximum output in the shortest time by reducing the idle times of machines and robot, the activity order of the robot in flexible production cells is of great importance. In the literature, there are mathematical studies on this subject. In theory, it is sometimes possible to find the optimum sequences, but sometimes the optimum results are only get closed. Obtained solutions in theory with mathematical models may not be applied in real life, they may not work. Therefore, it can be investigated by taking advantage of petri nets whether the theoretically obtained solutions are working in reality. Petri nets are a powerful, graphical language that can be used to represent activities in the system. Various modeling methods are available, such as resource-oriented, process-oriented, colored petri nets. In this study, modeling of flexible robotic cells with process-oriented petri nets is shown. At the end of the study, by building the petri net models of the cell with the intermediate buffer and the cell without intermediate buffer where the robot has a dual gripper, their productivity is compared. As a result, it has been found that the model of the cell with intermediate buffers does not work for the given sequence when the processing time is 120 seconds, and the cell with the dual gripper robot provides more efficient results for 30 seconds processing time.

Keywords: Flexible manufacturing systems, Flexible robotic cell, Petri net, Intermediate buffer, Productivity, Dual-gripper robot, Process oriented petri net

TEŞEKKÜR

Lisansüstü tez çalışmam sırasında, engin bilgi ve tecrübelerinden yararlandığım, her konuda destek olup yol gösteren, motive eden, her zaman yardımcı olan değerli danışmanım Sayın Dr. Öğr. Üyesi Güldal GÜLERYÜZ'e,

Tez çalışmam sırasında bilgi ve desteğiyle her zaman yardımcı olan, esnek robotik hücreler ve petri ağları konusunda beni yönlendirip tezime katkı sağlayan değerli hocam Sayın Öğr. Gör. Dr. Reza VATANKHAH'a,

Lisans ve lisansüstü eğitimim boyunca tecrübesini ve desteğini esirgemeyen, başta Sayın Prof. Dr. Murat Caner TESTİK olmak üzere tüm Hacettepe Endüstri Mühendisliği Bölümü hocalarıma,

Her zaman yanımda olan, bugünlere gelmemde en büyük pay sahibi çok değerli annem Sayın Fatma ŞAHİN ve babam Sayın Yaşar ŞAHİN'e,

Çok yakında birlikte bir aile olacağımız, her türlü desteğiyle her anımda yanımda olan Sayın Mustafa TUFANER'e,

Teşekkürlerimi sunarım.

Ayşegül ŞAHİN

Mayıs 2019, Ankara

İÇİNDEKİLER

ÖZET	i
ABSTRACT	iii
TEŞEKKÜR.....	v
İÇİNDEKİLER	vi
ŞEKİLLER DİZİNİ	viii
ÇİZELGELER DİZİNİ	ix
SİMGELER VE KISALTMALAR	x
1 GİRİŞ	1
1.1 Problem Tanımı	2
1.2 Motivasyon	3
1.3 Tezin Katkısı.....	4
1.4 Tezin Organizasyonu	4
2 LİTERATÜR ARAŞTIRMASI	6
2.1 Robotik Hücre.....	6
2.2 Esnek Robotik Hücre	7
2.2.1 Robotik Hücrelerde ve Esnek Robotik Hücrelerde çizelgeleme ile ilgili daha önce yapılmış çalışmalar	11
2.3 PETRİ AĞLARI	16
2.3.1 Petri ağlarıyla modellemede kullanılan temel kavramlar.....	20
2.3.2 Petri Ağlarının Özellikleri	23
2.3.3 Petri Ağlarıyla Modelleme	26
2.3.4 Petri ağlarıyla daha önce yapılmış çalışmalar	26
3 YÖNTEM.....	30
3.1 SÜREÇ ODAKLI PETRİ AĞI MODELLEME	30
3.1.1 Malzeme yükleme-boşaltma işlemlerinin POPN ile modellenmesi	31

3.2	ARA STOKSUZ, TEK KISKAÇLI, ÜÇ MAKİNELİ FRC İÇİN PETRİ AĞI MODELİ OLUŞTURULMASI	33
3.2.1	Üretim süreçlerinin sıralanmasının modellenmesi	34
3.2.2	Kaynakların modellenmesi.....	36
3.2.3	Operasyonlar için kaynak ihtiyaçlarının modellenmesi.....	36
4	ANALİZLER VE TARTIŞMALAR	42
4.1	Robotu tek kıskaçlı, ara stok alanlarına sahip iki makineli FRC.....	42
4.2	Robotu çift kıskaçlı, ara stok alanı olmayan iki makineli FRC.....	44
5	SONUÇ	60
6	KAYNAKLAR.....	62
	EKLER	65

ŞEKİLLER DİZİNİ

Şekil 2.1.	Bir robotik hücre örneğinin şematik gösterimi	7
Şekil 2.2.	Bir robot ve üç makineden oluşan esnek robotik hücre.....	8
Şekil 2.3.	CNC makinelere parçaları yükleyen ve boşaltan robotun hareket ettiği raylı hat, bir FRC örneği.....	9
Şekil 2.4.	Robotun CNC makineyi beslemesi	10
Şekil 2.5.	Basit bir petri ağı modeli	18
Şekil 2.6.	Geçiş t1 in ateşlemesi	20
Şekil 2.7.	Sistem özelliklerini göstermek için kullanılan petri net temellerinden bazıları	22
Şekil 2.8.	Tutucu bir petri ağı örneği	25
Şekil 3.1.	Süreç odaklı petri ağı (POPN) modeli örneği	31
Şekil 3.2.	Robot ve makinelerden oluşan bir üretim sisteminin POPN modeli örneği	33
Şekil 3.3.	Ara stok alanı olmayan, robotun tek kısıkaçlı olduğu model	34
Şekil 3.4.	M1 makinesinde işlenen parçalar için süreç odaklı petri ağı modeli	35
Şekil 3.5.	Tek kısıkaçlı, ara stoksuz sistemin petri net modeli.....	38
Şekil 4.1.	Robotu tek kısıkaçlı, ara stok alanları olan model	43
Şekil 4.2.	Esnek robotik hücrelerde kullanılan çift kısıkaç örneği	45
Şekil 4.3.	Ara stok alanları olmayan, robotun iki kısıkaçlı olduğu model	46
Şekil 4.4.	Ara stok alanları bulunan esnek robotik hücrenin süreç odaklı petri ağı modeli	48
Şekil 4.5.	Çift kısıkaçlı robotun olduğu esnek robotik hücrenin süreç odaklı petri ağı modeli	53
Şekil 4.6.	30 saniye işleme süresiyle çalıştırıldığında ilk konfigürasyon	57
Şekil 4.7.	120 saniye işleme süresiyle çalıştırıldığında ilk konfigürasyon	57
Şekil 4.8.	30 saniye işleme süresiyle çalıştırıldığında ikinci konfigürasyon.....	58
Şekil 4.9.	120 saniye işleme süresiyle çalıştırıldığında ikinci konfigürasyon.....	58

ÇİZELGELER DİZİNİ

Çizelge 3.1. Şekil 3.2'deki geçişlerin ve yerleşimlerin tanımları	31
Çizelge 3.2. Şekil 3.4'deki yerleşim ve geçiş düğümlerinin tanımları	35
Çizelge 3.3. Şekil 3.5'deki yerleşim ve geçiş düğümlerinin tanımları	39
Çizelge 4.1. Birinci ve ikinci konfigürasyon için robot aktivite sıralamaları	46
Çizelge 4.2. Şekil 4.4'deki petri ağı modelinin yerleşim ve geçiş düğümlerinin tanımları	49
Çizelge 4.3. Şekil 4.5'deki yerleşim ve geçiş düğümlerinin tanımları	54

SİMGELER VE KISALTMALAR

Simgeler

P	Yerleşimler kümesi
T	Geçişler kümesi
I	Giriş fonksiyonu
O	Çıkış Fonksiyonu
M ₀	Başlangıç İşaretleme

Kısaltmalar

CNC	Bilgisayar Nümerik Kontrollü
FRC	Esnek Robotik Hücre
RC	Robotik Hücre
POP _N	Süreç Odaklı Petri Ağı
TSP	Gezgin Satıcı Problemi
NP	Belirleyici Olmayan Polinom Zamanlı
NSG _{AI}	Domine Edilemeyen Sıralama Genetik Algoritması
TMG	Zamanlanmış İşaretli Grafik
TPN	Zamanlı Petri Ağları

1 GİRİŞ

Müşteri davranışlarındaki değişim ve özelleştirilmiş ürünlere olan talebin artmasıyla firmaların başarısı, özelleştirilmiş ve farklılaştırılmış ürünlerin aynı üretim sürecinde üretilmesine bağlı hale gelmiştir. Firmalar, geniş çeşitlilikteki kaliteli ürünleri kısıtlı zamanlarda üretmek zorunda kalmaktadırlar. Bunun için, üretim sistemlerine esneklik kazandırma çabasındadırlar [1]. Aynı üretim hattında, ürünleri daha kısa sürede ve çeşitlilikte üretebilmek için makinelerin yetenekleri artırılmış, robotlar üretim sistemlerinde daha fazla kullanılmaya başlanmıştır. Önce robotik hücrelerde bir parça tüm makinelerde işlenirken, makinelerin birden fazla işlem yapabilmesiyle, esnek robotik hücrelerde bir parça tek bir makinede işlenmeye başlamıştır. Parçaların üretim süresini düşürüp, aynı zamanda daha fazla çıktı elde ederek verimliliği artırma çalışmaları yoğun olarak devam etmektedir. Makinelerin dizilim şekli, makinelerin işlem süreleri ve kapasiteleri, robotun kısıtlı sayısı, ara stokların mevcudiyeti ve kapasitesi gibi girdiler değiştirilerek elde edilen çeşitli üretim konfigürasyonlarının üretkenliği denenip en verimli seçeneğe yatırım yapılmak istenmektedir.

Verimliliği artırmada, parçaların makinelerde işleme sırası ve robot hareketlerinin sıralaması da önemli bir araştırma konusu olmuştur. Literatürde, siparişe göre üretim sistemlerinde, esnek robotik hücrelerde çeşitli üretim konfigürasyonları için robotun aktivitelerinin en iyi sıralamasını arayan birçok çalışma mevcuttur. En iyi çözüme yakın sonuçlar bulmak için genellikle sezgisel algoritmalar kullanılmıştır [2]. Bulunan sonuçların birçoğu gerçek hayatta doğrulanamamıştır.

Petri ağları, esnek üretim sistemlerinin modellenmesi, analizi ve kontrolünde yaygın olarak kabul edilmiş grafiksel ve matematiksel bir araçtır. Bir matematiksel araç olarak petri ağları, sistem davranışlarını tanımlayan eşitlikler elde etmeyi, cebirsel sonuçlar bulmayı, başka matematiksel modeller geliştirmeyi sağlar. Blok şeması, mantıksal ağaç gibi diğer grafiksel sistem gösterim tekniklerine göre petri ağları, sistemdeki parçalar ve aktiviteler arasındaki mantıksal bağlantıları doğal bir yolla göstermeye daha uygundur [3]. Petri ağlarının esnek üretim sistemlerinde kullanılan süreç odaklı, kaynak odaklı, nesne odaklı, renkli petri ağları gibi çeşitli modelleme yöntemleri mevcuttur. Süreç odaklı petri ağlarında aktiviteler, stoklar ve kaynaklar yerleşim düğümleriyle, gerçekleşen olaylar geçiş düğümleriyle ifade edilir. Parçaların üretilmesi, hangi aşamalardan geçtiği, net ve ayrıntılı bir şekilde modellenabilir.

1.1 Problem Tanımı

Esnek robotik hücre, bir ürünün tüm işlemlerinin CNC makineler tarafından yapıldığı ve robotun parçaları makinelere yükleme ve makinelerden boşaltma görevini gerçekleştirdiği üretim birimidir [4]. Giriş ve çıkış stokları mevcuttur. Robot giriş stokundan aldığı hammaddeleri makinelere getirir ve yükler. Daha sonra işlenmiş parçaları makinelerden boşaltarak götürüp çıkış stokuna bırakır. Bu hücrelerde, tüm makineler üzerinde aynı işlemler gerçekleştirilir. Makineler özdeştir ve makinelerdeki işlemler birbirine paralel yürür, makinelerin işlem süreleri aynıdır. Parçanın üretilmesi için gereken tüm işlemler tek bir makine tarafından yapılır. Esnek robotik hücreler, sadece tek kısıkaçlı bir robot ve birbirine eş CNC makinelerden oluşabileceği gibi, ara stoklama alanlarının olduğu veya robotun çift kısıkaçlı olduğu hücreler de mevcuttur.

Robotun makinelere malzeme yükleme, giriş stokuna gitme, stoktan malzeme alma, makinelerden ürün boşaltma gibi aktivitelerinin sıralaması sistemin performansı için oldukça önemlidir. Literatürde, esnek robotik hücrelerde verimlilik oranlarını en üst düzeylere çıkarmak için robotun aktivitelerini sıralama çalışmaları mevcuttur [2]. Çeşitli eniyileme metodlarıyla elde edilmiş birçok sonuç bulunmaktadır, bu sonuçların çoğunda en iyi çözümlere ancak yaklaşılmıştır. İki-üç makineli FRC'lerde, tek kısıkaçlı robotun olduğu, ara stok alanlarının olduğu, iki kısıkaçlı robotun olduğu varyantlarda da robotun hareketlerinin en iyi sıralaması bulunmaya çalışılmıştır [5][4]. Matematiksel olarak en iyi çözümler bulunsa dahi gerçek hayatta bu çözümlerin çalışıp çalışmadığı, uygulanabilirliği, tıkanmaların, çıkmazların olup olmadığı araştırılmalıdır. Böyle bir araştırma burada, robotun çift kısıkaçlı olduğu ve ara stok alanları olan iki farklı hücre için petri ağlarından yararlanılarak yapılmaktadır.

Petri ağları, sistemi grafiksel dille tanımlayarak kurulan matematiksel modelin gerçek hayatta kullanılabilirliğinin kontrolünü sağlar [6]. Sistemdeki bilgi, malzeme, vb. akışını göstererek akıştaki mantık hatalarının bulunmasını, darboğazların tespitini kolaylaştırır. Sistemin performansını artırmak için atılacak adımların kolayca denenmesini mümkün kılar. Petri ağlarının kaynak odaklı ve süreç odaklı modelleme gibi çeşitli modelleme yöntemleri mevcuttur. Çalışmada kullanılacak olan süreç odaklı petri ağı modelleme yönteminde; sistemdeki aktiviteler, kaynaklar ve kaynak ihtiyaçları belirlenir. Tüm bunlar modellenerek birbirleriyle bağlantısı sağlanır ve aktiviteler belirlenen sırayla çalıştırılır.

Bu tez çalışmasında esnek robotik hücrelerin süreç odaklı petri ağı yöntemiyle modellenmesi anlatılmaktadır. Daha önceki çalışmalarda esnek robotik hücrelerin süreç odaklı petri ağlarıyla modellenmesi bulunmamaktadır. İlk olarak, ara stok alanı olmayan ve tek kısaçlı robotun olduğu sistemin petri ağı modelinin kurulması bir örnekle gösterilmiştir. Daha sonra ara stok alanı olan ve tek kısaçlı robotun olduğu sistem ile ara stok alanı olmayan iki kısaçlı robotun olduğu sistemlerin de petri ağı modelleri kurularak bu iki sistemin üretkenliği karşılaştırılmıştır. İki sistem için de literatürde daha önce matematiksel olarak bulunmuş robot aktivitelerinin en iyi sıralama senaryolarının canlılığı, gerçek hayatta uygulanabilirliği kontrol edilmiş ve çıktıları analiz edilip performansları karşılaştırılmıştır. Senaryolara ait kaynak kullanım oranları, çevrim süreleri, çıktı miktarları gibi performans göstergeleri incelenerek verimlilik analizleri yapılmıştır.

1.2 Motivasyon

Esnek robotik hücreler, son yıllarda otomasyonun fabrikalarda yaygınlaşmaya başlamasıyla daha fazla kullanılmaya başlanmıştır. Hücrelerdeki robot ve makine gibi araçlar, oldukça pahalı yatırım kalemleridir. Bu araçları en verimli şekilde kullanabilmek için, makinelerin kullanım sırası, dizilim şekli, ne kadar sürede parçayı işleyeceği, ara stok alanlarının olup olmaması ve robot sayısı, robotun sahip olduğu kısaç sayısı, robotun aktivitelerinin sıralaması gibi faktörler yatırımcılar ve yöneticiler için oldukça önemlidir. Araştırmacılar için de geniş bir çalışma konusu oluşturmuşlardır. Literatürde, sistemdeki araçların çeşitli kombinasyonları ve farklı kullanım şekilleriyle yapılmış birçok çalışma mevcuttur. Çevrim süresini düşürüp çıktı miktarını artırmak için robot hareketlerinin en iyi çizelgelemesi konusunda problemler matematiksel olarak çözülmeye çalışılmıştır. Problemler karmaşık ve çözümü zor olduğundan en iyi çözümlere genellikle sezgisel algoritmalarla ulaşmayı denemişlerdir [7]. Teoride, matematiksel olarak uygun bulunan çözümlerin çoğu gerçek hayatta kontrol edilip doğrulanmamıştır [5]. Pahalı yatırımlar yapılmadan önce bu çözümlerin uygulanabilirliği test edilmelidir. Ayrık olay (discrete event) sistemlerin kontrolü için petri ağı modellerini kullanmak uygun bir seçenektir. Sistem, ayrıntılarıyla incelenmek istendiğinde süreç odaklı petri ağları oldukça iyi bir yöntemdir. Tüm aktiviteler, kaynaklar ve bu aktiviteler arasında akan parçalar bu yöntemle elde edilen modellerle rahatlıkla görülebilir; sistemin darboğaz noktaları, kaynak ihtiyaçları ve kapasite yetersizlikleri gibi durumlar kolayca tespit edilebilir.

Ayrıca, kaynak kullanım oranları, çevrim süreleri gibi verilere de ulaşılarak sistemin üretkenliği incelenebilir, sistemler arası üretkenlik karşılaştırılması yapılabilir.

1.3 Tezin Katkısı

Bu çalışmada esnek robotik hücrelerin süreç odaklı petri ağlarıyla modellenmesi anlatılmaktadır. Esnek robotik hücrelerin yapısı, çalışma şekli anlatılarak, bunların petri ağlarıyla nasıl ifade edildiği gösterilmektedir. Ara stok alanları olan hücrelerle, çift kısıklı robotun olduğu hücrelerin de çalışma yapıları anlatılmış, petri ağlarıyla nasıl modellendiği gösterilmiştir. Ayrıca, çalışmanın sonunda bir vaka analiziyle, literatürde daha önce matematiksel olarak bulunmuş robotun en iyi aktivite sıralamalarının gerçekte çalışıp çalışmadığı kontrol edilmiş, iki farklı hücre yapısının üretkenlikleri karşılaştırılmıştır. Literatürde var olan iki farklı çalışmada matematiksel sonuçları verilen hücre yapıları kullanılmış ve bu hücrelerin petri ağı modelleri kurulmuştur. Çalışmalardan elde edilen matematiksel çevrim sürelerinin petri ağlarıyla elde edilen çevrim süreleriyle aynı olduğu görülmüş, böylece yöntemin doğru olduğu ispatlanmıştır. Bu çalışmayla, esnek robotik hücrelerin süreç odaklı petri ağlarıyla modellenebileceği, çeşitli hücre yapılarının kaynak kullanım oranlarının ve çevrim sürelerinin karşılaştırılabileceği gösterilmiştir. Bu hücrelerde matematiksel olarak bulunmuş robot aktivite sıralamalarının gerçekte çalışıp çalışmadığının kontrol edilmesi gerektiği vurgulanmıştır.

1.4 Tezin Organizasyonu

Tez çalışması, literatür araştırmasıyla devam etmektedir. Literatür araştırmasında ilk olarak, esnek üretim sistemleri, robotik sistemler ve esnek robotik sistemlerle çalışma yapıları anlatılarak bu konularda daha önce yapılmış çalışmalar örneklendirilmiştir. Daha sonra, petri ağları, yapısı, modellerin kurulması gösterilmiş ve petri ağlarıyla daha önce yapılmış çalışma örnekleri verilmiştir. Literatür araştırması bölümünden sonra yöntem bölümüne geçilmektedir. Yöntemde, süreç odaklı petri ağlarıyla modellemenin nasıl yapıldığı ve esnek robotik hücrelere uygulanması açıklanmıştır. Son olarak sonuçlar ve tartışma bölümünde, ara stok alanları olan tek kısıklı robotun olduğu esnek robotik hücrelerle ara stok alanı olmayıp çift kısıklı robotun olduğu hücrelerin petri ağı modelleri kurulmuş, literatürde daha önce bulunmuş robot hareketi sıralamalarının

gerçekte çalışıp çalışmadığı kontrol edilmiş, hücrelerin üretkenlikleri karşılaştırılmıştır. Yorum bölümünde de, elde edilen sonuçlar ve bulgular tartışılmıştır.

2 LİTERATÜR ARAŞTIRMASI

Esnek üretim sistemleri, bilgisayar nümerik kontrollü makineler tarafından desteklenen bütünleşmiş üretim sistemleridir. Bu sistemler, fabrikaların makinelerini hızlı bir şekilde yeniden yapılandırmasını sağlayan hatlarda, yüksek teknoloji ve otomasyon kullanarak, aynı kaynaklarla daha fazla çeşitte ürünün üretilmesine olanak sağlar [1]. Esnek üretim sistemleri yüksek sermaye yatırımları gerektirir. Yatırımlardan kısa zamanda karlı dönüşler bekleniyorsa, en iyi kaynak kullanımı sağlanmalıdır. Optimum kaynak kullanımı, üretim zamanını ve maliyetlerini minimize ederken verimliliği maksimize eder. Gelecekteki üretim, giderek müşteriye göre üretime doğru kaymaktadır. Gittikçe daha fazla ürün kişiselleştirilmektedir [8].

Endüstri 4.0, son derece esnek üretim şartları altında, ürünlerin güçlü bireyselleşmesiyle nitelendirilir. Müşteri odaklı üretim, doğrudan karmaşık esnek üretim sistemlerine götürür. Bu esnek üretim sistemlerinin modellenmesi ve optimize edilmesi gerekir. Endüstri 4.0 içindeki esnek üretim sistemlerinin kullanılabilirliğinin analizi, beklenmedik makine arızaları ve buna bağlı olarak ortaya çıkan kayıpların riskini azaltmayı amaçlarken üretim sistemlerinin optimizasyonunu da hedefler [9]. Esnek üretim sistemlerinde otomasyonun artmasıyla önce robotik hücreler daha sonra da esnek robotik hücreler kullanılmaya başlanmıştır.

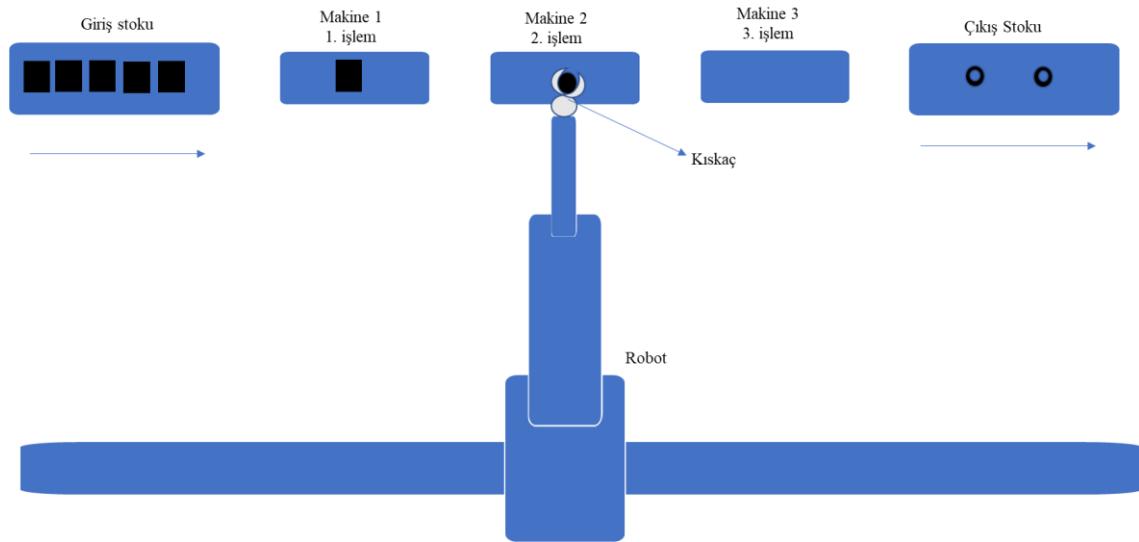
2.1 Robotik Hücre

Otomasyon donanımlarının verimliliğini artıran teknolojik gelişmelerle ve yeni otomasyon teknolojilerinin kurulmasıyla üretim endüstrisinde otomasyon uygulamalarının önemi artmıştır. Robotlar, endüstride kullanılan en genel otomasyon donanımlarındandır ve çoğunlukla malzeme taşıma aracı olarak kullanılırlar [10]. Üretim hücrelerinde robot ve bilgisayar nümerik kontrollü makinelerden oluşan, robotik hücreler mevcuttur. Robotik hücre, her makinede farklı bir işlem yapıp parçanın ihtiyaç duyduğu işlemlerin bir grup makine tarafından tamamlandığı üretim sistemidir. Frezeleme, döndürme, delme gibi işlemlerin farklı makinelerde yapıldığı robotik hücrelerde, makineler birbirine bir malzeme taşıma sistemiyle bağlıdır [11]. Bir parça ayırım olmaksızın tüm makineler tarafından işlenir. Dolayısıyla bir parçanın üretimi için gerekli işlemlerin sırasına göre, parça makineler arasında hareket eder. Aynı parçalar aynı işlem sırasıyla üretileceğinden, parçalar makineler arasında benzer sıralamayla gider. Her bir

makine tek bir işlem gerçekleştirir ve malzeme taşıma sistemiyle parça sıradaki makineye geçer. Bir robotik hücrede, tek bir makine tüm operasyonları tamamlayamaz.

Sıralı robotik hücreler, robot merkezli hücreler, seyyar robotlu hücreler gibi farklı robotik hücre düzenleri mevcuttur. Sıralı robotik hücrelerde makineler lineer sıralı bir şekilde dizilmiştir ve robot makinelerin üzerinde düz bir çizgide malzeme taşımaktadır. Robot merkezli hücrelerde, robot hücrenin merkezine yerleşmiştir ve makineler robotun etrafında dairesel bir şekilde dizilmiştir [10]. Seyyar robotlu hücrelerde robot, makineler arasında sabit bir hatta değil, özerk bir şekilde gezinir.

Aşağıda Şekil 2.1’de bir sıralı robotik hücre örneği verilmiştir. Giriş stokundan alınan parçalara, makinelerde sırasıyla farklı işlemler yapılmaktadır ve işlemleri tamamlanan parçalar çıkış stoku bırakılmaktadır.



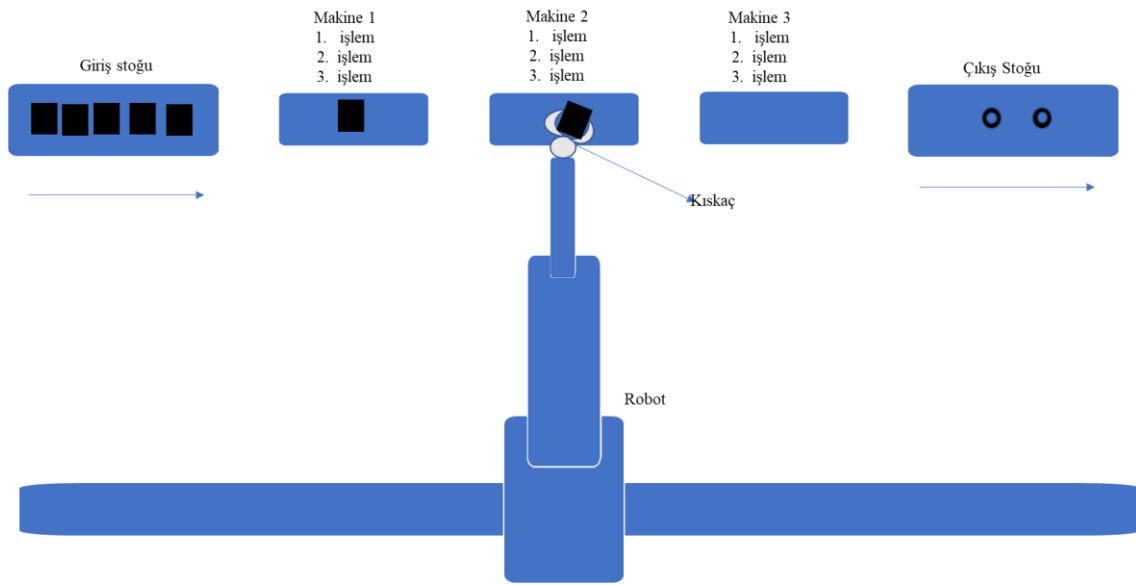
Şekil 2.1. Bir robotik hücre örneğinin şematik gösterimi

CNC makinelerin teknik özelliklerinin geliştirilmesiyle, bir CNC makinede birden fazla işlem yapılabilir hale gelmiş ve bu da robotik hücrelerden esnek robotik hücrelere doğru bir geçişe neden olmuştur. Bir esnek robotik hücrede, aynı parçaların tüm işlemleri tek bir makine üzerinde, aynı işlem süresiyle tamamlanır [12]. Bir parça makineye yüklendiğinde, sadece bir operasyonu değil, kalan tüm operasyonları gerçekleştirilir.

2.2 Esnek Robotik Hücre

Esnek robotik hücre, içinde bir dizi makineyle bu makineleri yükleyen, boşaltan ve makinelerle giriş/çıkış stokları arasında parçaları taşıyan bir robotun olduğu üretim sistemidir. Esnek robotik hücreler, elektronik üretimi, motor üretimi, otomobil krank mili

üretimi ve metal kesimi gibi birçok endüstride yoğun olarak kullanılır [4]. Giriş ve çıkış stokları mevcuttur. Giriş stokundan robot tarafından alınan malzemeler makinelere yüklenir ve makinelerden yine robot tarafından alınan malzemeler çıkış stokuna getirilerek bırakılır. Esnek robotik hücrelerde, bir parça yalnızca bir makine üzerinde işlenir, parçanın işlemlerinin tamamlanması için tek bir makine yeterlidir. Üç makine ve tek robottan oluşan bir esnek robotik hücre örneği Şekil 2.2’de verilmiştir. Makineler birbirine seri halde dizilmiştir. Robot, bir hat üzerinde gidip gelmektedir ve kısıpçaları yardımıyla; çıkış stokuna ve makinelere yükleme, giriş stokundan ve makinelerden boşaltma işlemlerini gerçekleştirmektedir.



Şekil 2.2. Bir robot ve üç makineden oluşan esnek robotik hücre

Esnek robotik hücrelerde çoğunlukla CNC makineler kullanılır, bu da üretim sistemlerine çok büyük esneklik kazandırır. Bir CNC makine, gerekli araçlar yüklü olduğu sürece, geniş çeşitlilikte işleme gerçekleştirebilir. Makinelerin kabiliyetlerinin yanı sıra, esnek robotik hücredeki robotun yetenekleri de sistemin performansını etkileyen önemli bir faktördür. Özellikle, robotun sahip olduğu kısıpçaları sayısı robotun yeteneklerini önemli ölçüde etkileyebilir.

Esnek robotik hücrelerde kullanılan robotlar, geniş çeşitlilikteki tekrar eden görevleri otomatikleştirmeye yarar. Sık tekrar eden işler, bir robot tarafından yapıldığında, iş gücündeki boşluklar azalarak saatlik çalışma maliyeti azalır. İstenirse günün her saati çalıştırılabilecek robotlar, çalışma süresini maksimum düzeye çıkarır. Daha fazla işlem daha doğru bir şekilde yapılabilir. FRC'lerde kullanılan robotlar istenilen rotada hareket edebilir ve birden fazla makinenin çalıştırılması için programlanabilir [13]. Robotun

taşıyacağı malzemenin çeşidine göre, robot koluna bağlı kısıkaçlar hızlı bir şekilde değiştirilerek uygun kısıkaçın kullanılması sağlanabilir.

Şekil 2.3’de raylı bir hat üzerinde giriş stokundan çıkış stokuna hareket eden, aynı işlemleri yapan CNC makineleri yükleyip boşaltan bir robotun olduğu, esnek robotik bir hücre örneği görülmektedir.



Şekil 2.3. CNC makinelere parçaları yükleyen ve boşaltan robotun hareket ettiği raylı hat, bir FRC örneği¹

FRC’ler, kitlesel bireyselleştirme üretim modeli altında çalışan firmalar tarafından büyük bir memnuniyetle karşılanmıştır; çünkü siparişe göre üretim sistemlerine göre daha esnek ve çevik olmasının yanı sıra, daha büyük miktardaki ürünleri daha iyi bir kaliteyle üretebilmeyi sağlamaktadır [14]. Esnek robotik hücrelerde, makineler birbirleriyle eşir. Üretilen parçaların her biri aynıdır ve genellikle üretilmeleri için makineler tarafından yapılan birkaç işlem vardır. Makineler, parçaların üretilmesi için ihtiyaç duyulan işlemleri yapabilmek için yeterince esnektir. Bir parça bir makineye yüklendiğinde, makine ürünün üretilmesi için gerekli tüm işlemlerini tamamlayarak nihai ürünü ortaya çıkarabilir [2]. Örneğin, bir CNC makine üzerinde tek bir yükleme işlemiyle, frezeleme, delme, delik genişletme, diş açma işlemleri yapılabilir. İşlenecek parçaların işlem programları, makinelerin kontrol ünitesi hafızasında saklanabilir; programda bir değişiklik olursa kolayca yapılarak yine kontrol ünitesi hafızasında tutulabilir.

Bir esnek robotik hücrede, tüm makineler üzerinde aynı işlem grupları gerçekleştirilebilir. Dolayısıyla, bir parça, sadece bir makine üzerinde işlenir. Şekil 2.4’te bir CNC makinenin

¹ <https://www.hillaryinc.com> web adresinden 07.04.2019 tarihinde alınmıştır.

robot tarafından yüklenmesi gösterilmiştir. Robot kısıncıyla tuttuđu parçayı, makineye parçanın kalan işlemlerini gerçekleştirmesi için yerleştirmektedir. Parçanın işlemleri tamamlandıktan sonra, robot aynı şekilde gelerek kısıncıyla parçayı tutup makineyi boşaltacaktır.



Şekil 2.4. Robotun CNC makineyi beslemesi²

Giriş parçalarının sırasına bağılı olarak, bir hücrenin durumu (örneğin, makineler ve robotların durumu) tersinir olabilir. Tersinir olmak, hücrenin bileşenlerinin (makineler, robot, vs.) başlangıç durumuna ya da başlangıç durumu olarak kabul edilmiş duruma daima geri dönebileceğini ifade eder. Bu durumda çizelgeleme problemi döngüsel, tersinir döngünün süresi, çevrim süresi olarak adlandırılır [15]. Örneğin, tüm makineler ve robotun boş olduğu ve robotun çıkış stokunun önünde beklediği durumu, başlangıç durumu olarak kabul edelim. Sistem çalıştırıldığında robot, çıkış stokundan giriş stokuna gelerek işlenmemiş bir parça alacak, bu parçayı makinelerden birine götürüp yükleyecek ve daha sonra işlenmiş ürünü makineden alarak çıkış stokuna götürüp bırakacaktır. Bu durumda yine makineler boş ve robot çıkış stokunun önünde bekliyor olduğundan, sistem başlangıç durumuna dönmüş olur. Böylece, döngü tamamlanmış olacaktır. Bir döngüde, her bir makine bir parçayı işler, kaç makine varsa o kadar parça üretilir ve çevrim süresinin kısılması, üretim oranının artması anlamına gelir. Çevrim süresi, sistemdeki

² <http://demrobotikotomasyon.com> web adresinden 07.04.2019 tarihinde alınmıştır.

robotun hareketlerinin sırasına bağlıdır [16]. Çevrim süresini minimize etmek için hareketlerin sırasını belirlemek (çizelgeleme) bir eniyileme problemidir.

Bir esnek robotik hücrenin çizelgeleme problemi, robotun hareketlerinin döngüsel çizelgelenmesi ve operasyon dağıtımı olarak genelleştirilebilir [5]. Robot hareketlerinin döngüsel çizelgelenmesinde, robotun hareketlerinin sıralamasının en iyi döngüsü istenen tüm denge durumları arasından düşünülür. Operasyon dağıtımında ise, her bir makine için makine üzerinde yapılacak işlemlerin en iyi sıralaması bulunur. Makine ve operasyon sayısı arttığında çizelgeleme problemi giderek zorlaşır. Bu nedenle literatürde çoğunlukla iki makineden oluşan ve tek bir parça tipi üreten basit hücrelerde robotun hareketlerinin sıralaması (çizelgeleme) çalışmaları yapılmıştır. Robot ve makineyle ilgili kısıtlar (robotun kısaç sayısının değişmesi ya da makinelerin ara stoklama alanlarının olması gibi) ve hücrenin yerleşimi de problemi zorlaştırabilir.

2.2.1 Robotik Hücrelerde ve Esnek Robotik Hücrelerde çizelgeleme ile ilgili daha önce yapılmış çalışmalar

Bazı araştırmacılar robotik hücrelerde ve esnek robotik hücrelerde çizelgeleme problemlerinin çözümlerine katkıda bulunmuşlardır. Crama ve Klundert [17] çalışmalarında siparişe göre üretim yapan üç makineli, makinelerin sıralı dizildiği, bir robotik hücre modeli düşünmüşlerdir. Tek bir robot makineler arasında malzeme taşımaktadır. Her bir makine farklı bir işlem yapmakta ve parça her döngüde üç makinede hep aynı sırayla işlenip üretimi tamamlanmaktadır. Üç makinede aynı parçanın işlenmesi sonucu, tek bir birim ürün elde edilmektedir. Böyle döngülere, bir-birim döngüler denir. Robot sabit sıralı aktiviteleri yapmakta ve birbirine eş parçalar üretilmektedir. Üç makineli, ara stok alanı olmayan robotik hücrelerde bir-birim döngülerin en kısa çevrim süreleriyle en iyi üretim oranları verdiği varsayıp doğrulanmıştır. Sonuçta, robot hareketlerinin en kısa süreli döngüsel çizelgesi bulunmuştur.

Gültekin ve diğerleri [18] aynı parçaların üretildiği bir robotik hücrede çizelgeleme problemlerini incelemişlerdir. Süreçlerin ve işlemlerin esnek olduğu yaklaşımıyla elde edilen robot hareketlerinin döngüsünün, iki makineli hücrelerde daha önceki yapılan çalışmalardan baskın olduğu görülmüştür. Ayrıca, makine yerleşimi sıralı yerleşimden robot merkezli yerleşime değiştirildiğinde, çevrim süresinin düştüğü kanıtlanmıştır.

Gültekin, Karaşan ve Aktürk [19] çalışmalarında m makineli esnek robotik hücrede Gezgin Satıcı Problemi yöntemiyle en iyi döngüyü bulmaya çalışmışlardır. İki makineli

bir hücre için ilk durumda belirlenen iki döngüden birinin diğer döngüler arasında en uygun olduğunu bulmuşlardır. Ayrıca çalışmalarında, literatürdeki siparişe göre üretim tipindeki robotik hücrelerin döngülerine göre esnek robotik hücrelerin döngülerinin daha iyi olduğunu kanıtlamışlardır. Esnek üretim hücrelerinin çevrim sürelerinin daha kısa, çıktı miktarlarının daha fazla olduğunu ispatlamışlardır. İşleme zamanı, robot ulaşım süreleri, makinelerin yükleme/boşaltma süreleri gibi verilen hücre parametreleri için çevrim süresini minimize eden makine sayısını bulmuşlardır.

NP (Belirleyici Olmayan Polinom Zamanlı), karar problemlerinin karmaşıklık sınıfıdır. NP-hard problemler, polinomsal zamanda bir çözümü olduğu ispatlanamamış problemlerdir. Esnek robotik hücrelerde çizelgeleme problemleri NP-hard olduğundan, makine sayısı arttığında çözüm süresi hızlıca artar ve kesin çözüm yöntemleri bu problemleri çözmekte yetersiz kalır. Bu durumda en etkili yaklaşımlardan biri metasezgisel algoritmaları kullanmaktır [2]. Genetik algoritmalar, literatürde en sık kullanılan algoritmalar arasındadır.

Nejad ve diğerleri [2] eş olmayan ürünlerin üretildiği, m paralel makineli bir esnek robotik hücrede, çevrim zamanını minimize etmek için robotun aktivitelerinin sıralamasını bulmaya çalışmışlardır. Problem için evrensel bir çizelgeleme modeli (MTZ-tabanlı TSP) kurup bekleme zamanı değişkenlerini çıkararak problemin küçültülmüş sürümünü elde etmişlerdir. MTZ (Miller-Tucker-Zemlin), TSP problemlerini tam sayılı lineer programlamada formüle etme metodudur. TSP ziyaret edilecek noktalar arasında, her noktayı ziyaret edecek şekilde mümkün olan en kısa yolla başlangıç noktasına geri gelmeyi sağlayan rotayı arar. TSP modelleme yaklaşımları iyi sonuçları kısa sürede verse de esnek üretim sistemlerinin çizelgeleme problemleri için hesaplama süresi çok uzundur; altı makineli bir hücrede çözüm 223 saatte bulunabilmektedir. Bu nedenle, bekleme zamanları eniyileme problemi çözülüp sonuçlar bulunduktan sonra da hesaplanabileceğinden, problemin çözüm süresini kısaltmak için bekleme zamanı değişkenleri çıkarılabilir diye düşünmüşlerdir. Geliştirilen model altı makineye kadar sayısal olarak sınanmıştır. En iyi sonuçların veya alt sınırların elde edilmesi için atama problemi yöntemi kullanılmıştır. Literatürdeki modelle karşılaştırıldığında modelin küçültülmüş sürümünün çok daha etkili olduğu görülmüştür. İki-altı makineli hücreler için robotun hareketlerinin sıralamasının en uygun çözümü genelleştirilmiştir.

Nejad ve diğerleri [7] çalışmalarında makinelerin birbiriyle aynı ve paralel olduğu bir esnek robotik hücrede çizelgeleme problemi incelemiştir. Çevrim süresi, robotun

hareketlerine bağılı olduğundan, çevrim süresini minimize eden robot hareketlerini bulmaya çalışmışlardır. Problemi çözmek için, TSP'nin ağ akış modelleme yaklaşımı kullanılarak yeni bir matematiksel model sunulmuştur. Büyük boyutlu problemler için alternatif olarak tavlama benzetimi yöntemi geliştirilmiştir. Önerilen yöntemlerle birkaç sayısal örnek çözümlenerek performansları değerlendirilmiştir. Önerilen yeni modelin literatürdeki modele göre daha iyi olduğu ve tavlama benzetimi yönteminin kabul edilebilir sürelerde neredeyse en iyi sonuçları bulabildiği görülmüştür.

Nejad ve diğerleri [14] esnek bir robot hücrede, hücredeki işlemlerin çizelgelemesi ve üretim maliyetleri arasında denge kurulmasını amaçlayan iki amaçlı bir çizelgeleme problemi incelemiştir. Hücrede makineler özdeş ve paralel sıralıdır. Bir robot, giriş stokundan makinelere ve makineden çıkış stokuna kadar parçaların yüklenmesinden ve boşaltılmasından sorumludur. Sistem döngüsel, bu da her döngüde aynı işlemlerin tekrar ettiği anlamına gelir. Her bir makinenin her döngüde bir parça işlediği varsayılmaktadır. Problemi çözmek için iki-amaçlı (üretim maliyetlerini ve çevrim süresini minimize etmeyi amaçlayan) bir matematiksel model sunulmuştur ve bir alternatif olarak büyük boyutlu problemler için bir Domine Edilemeyen Sıralama Genetik Algoritması, NSGAI, yöntemi geliştirilmiştir. NSGAI domine edilmemiş sıralamanın genetik algoritmalarla birleştirilmiş halidir, algoritmanın çözümleri sıralamasını en iyi çözümü seçmesini mümkün kılar. Domine edilmemiş sıralama, rastgele çözümlerin olduğu bir popülasyonla başlar. Her çözüm bir kromozomla gösterilir. Her bir iterasyonda, popülasyonun bazı üyeleri rastgele ebeveyn olarak seçilir, ebeveynler eşleştirilir, yavru çözümler üretilir ve çözüm popülasyonuna eklenir. Çözümler uygunluk fonksiyonu değerine göre sıralanır. Bir sonraki nesil üstün çözümlerin seçilmesiyle gelişir. Birden fazla amaç olduğunda, domine edilmemiş sıralama tek başına yeterli değildir, bu nedenle genetik algoritmayla birleştirilmiştir. NSGAI ile sonuçların hesaplanma süresi önemli ölçüde azalmıştır.

Yıldız ve diğerleri [10] çalışmalarında m CNC makine ve merkezinde bir robot olan birbirine eş parçalar üreten esnek robotik hücreyi incelemiştir. Bir parça sadece bir makine üzerinde işlenmekte ve her bir döngüde m birim üretilmektedir. İki makineli hücre için çevrim süresini minimize etmek amacıyla önce iki saf döngü belirlenmiş, daha sonra bunlara göre çevrim sürelerinin daha kısa olduğu döngüler araştırılmıştır. Daha genel durumlara öngörü sağlamak amacıyla 3 makineli durumlar yoğun olarak çalışılmıştır. Makinelerin işlem süresinin kısılması, üretim maliyetinin artmasına neden

olmaktadır. Üretim maliyetlerini minimize etmek de amaçlardan olduğundan makinelerin işlem sürelerinin kontrol edilebildiği varsayılmıştır. 3 makineli bir hücrede muhtemel 120 farklı döngü, 120 robot hareketi vardır. Bu robot hareketlerinin minimum çevrim süreleri hesaplanmıştır. İşlem süreleri düşürülüp artırılarak farklı sürelerle çalışmalar yapılmıştır. Sırasıyla, çevrim süresini ve üretim maliyetini minimize eden uygun robot hareketlerinin sıraları bulunmuştur. Üç farklı döngü seviyesi K için ikişer farklı sıralama olmak üzere toplam altı sıralama önerilmiştir. Çevrim süresine göre üretim maliyetlerinin bir grafiği elde edilmiştir.

Literatürde tek kısıkaçlı robotlar yoğun olarak çalışılmıştır. Çift kısıkaçlı robotların yer aldığı hücreler daha üretken olmalarına rağmen, doğal karmaşıklıkları nedeniyle daha az dikkat çekmiştir. Jung ve diğerleri [20] çalışmalarında devamlı olarak aynı parçaları üreten, makinelerin robotun etrafında dairesel bir düzenle durduğu çift kısıkaçlı robotik hücrelerde çizelgeleme problemini incelemişlerdir. İncelenen robotik hücrede, bir parça tüm makineler üzerinde işlenmek zorundadır, makinelerin işleme sırası tüm parçalar için aynıdır. İki kısıkaçlı robotun olduğu tipik bir operasyonda, robot makinenin yanına bir kısıkaç boş ve diğer kısıkaçında bu makinede işlenecek bir parça tutuyor halde gelir. Robot, makineden işlenmiş parçayı boşaltır ve kısıkaçlarını döndürerek işlenmemiş parçayı makineye yükler. Robotik hücrelerde bir birim döngüsel çizelge, tek bir parça üreten sabit robot hareketlerinin sırasını ifade eder. Hücrede, verimi en üst düzeye çıkaracak robot hareketlerinin bir birimlik döngüsel çizelgesini bulmaya çalışmışlardır. Polinom-zamanlı algoritmalarla en uygun olarak ya da uygun sınırlar içerisinde problemin çözülebildiğini göstermişlerdir. Daha sonra sonuçlar lineer sıralı hücrelere genişletilmiştir. En kötü durum sınır önerilerinden bu algoritmanın ortalamada çok daha iyi olduğu hesaplanmıştır.

Gültekin ve diğerleri [4] doğrusal olarak yerleştirilmiş aynı CNC makinelerinden ve robottan oluşan bir esnek robotik hücre düşünmüşlerdir. Hücrede, bir çift kısıkaçlı robot ve taşıma robotu bulunmaktadır. Bu sistemde, aynı parçalar işlenecek ve CNC makineleri işlenecek bir parçanın ihtiyaç duyduğu tüm işlemleri gerçekleştirebilecek kapasitededir. Verimlilik oranını en üst düzeye çıkarmak için robotun aktivitelerinin sıralanması problemi ele alınmıştır. CNC makinelerinin esnekliğinin bir sonucu olarak ortaya çıkan yeni robot hareket dizilerini saf döngüler olarak adlandırmışlardır. Saf bir döngüde, robot her makineyi bir kez yükler ve boşaltır ve her parça tam olarak makinelerin birinde işlenir. Dolayısıyla problem, çıktı oranını maksimuma çıkaran en iyi saf döngüyü belirlemektir.

Çalışmada ilk olarak saf döngüler için uygunluk koşulları belirlenmiş ve uygun saf döngü sayısını azaltan bazı temel sonuçlar kanıtlanmıştır. İki makineli robot hücresi detaylı olarak analiz edilip çok sayıda uygun saf döngünün arasından beş döngünün diğerlerine baskın geldiği görülmüştür.

Sriskandarejah ve Shetty [21] çift kısıkaçlı robotun olduğu robotik hücrelerdeki çıktı eniyilemesi ile ilgili son yıllardaki teorik gelişmeleri incelemişlerdir. Devamlı aynı parçayı üreten, çift kısıkaçlı robotun olduğu hücrelerde çizelgeleme problemlerine odaklanmışlardır. Amaç, parçanın ortalama üretim süresini en aza indirerek çıktı miktarını en fazla hale getirecek olan robot hareketlerinin bir döngüsel sırasını bulmaktır. Son zamanlarda raporlanmış çözümlere ek olarak pratikte düşünülmemiş ama verimliliği artıracak döngüsel sıralamalar düşünülmüştür.

Jung ve diğerleri [22] çalışmalarında tek kısıkaçlı ve iki kısıkaçlı robotun olduğu, ara stok alanları olmayan, makinelerin dairesel olarak dizildiği bir robotik hücrede çizelgeleme problemi incelemişlerdir. Hücreler aynı parçaların üretimi için tasarlanmıştır. Amaç, çıktıyı maksimize edecek şekilde bir parçanın üretilmesi için gerekli ortalama süreyi minimize eden robot hareketlerinin sırasını bulmaktır. $K \geq 1$ olmak üzere en uygun k birim döngüsel çözüme yaklaşım için etkili bir algoritma elde etmeye odaklanmışlardır. Polinom algoritmasıyla elde edilen sonuçlar, en kötü sınırların önerdiğinden çok daha iyi olduğu görülmüştür.

Foumani ve diğerleri [23] çalışmalarında çift kısıkaçlı robotun olduğu hücrelerde çevrim süresini optimize etmek için yeni bir yöntem geliştirmişlerdir. Hücrede aynı sırayla, aynı parçalar üretilmektedir. M makinenin olduğu robotik hücrede, birincil işlemler merkez makinede yapılmaktadır ve $m-1$ makinede ikincil işlemler yapılmaktadır. Parçalar önce merkez makineye gelir ve diğer makinelerde ikincil işlemleri yapıldıktan sonra tekrar merkez makineye gelir. Robotun çift kısıkaçlı olduğu hücrede, robotun hareketlerinin iki birim döngülü en uygun sıralamaları bulunmuştur.

Drobouchevitch ve diğerleri [24] tek parça tipinde üretim yapan robotik hücrelerde, her bir makinede giriş ara stok alanı ve çıkış ara stok alanı olduğu durumdaki çizelgeleme problemini incelemişlerdir. Tek kısıkaçlı bir robotla siparişe göre üretim yapan bir robotik hücrede çalışılmaktadır. Çıktı miktarını maksimize etmek ve bir parçanın ortalama üretim süresini minimize etmek için robot hareketlerinin en uygun sırasını bulmak istemişlerdir. Makinelerdeki çıkış ara stok alanlarının çıktı miktarını artırdığı, fakat çıkış ara stok

alanlarına bir de giriş ara stok alanları eklendiği durumda çıktı miktarının değişmediği görülmüştür.

Nejad ve diğerleri [5] ara stoklama alanları bulunan esnek robotik hücrelerde, robotun hareketlerinin en iyi sırasını bularak hücrenin çevrim süresini minimize etmeyi amaçlayan çizelgeleme problemini incelemişlerdir. Problem matematiksel olarak modellenip genetik algoritma, tavlama benzetimi ve hibrit genetik algoritmayla çözülmüştür. İki ve üç makineli hücreler için çeşitli senaryolar çözümlenerek makine sayısı arttığında çözüm süresinin üstel olarak arttığı gösterilmiştir. Hibrit genetik algoritmanın tavlama benzetimi ve genetik algoritma yöntemlerinden daha iyi sonuç verdiği görülmüştür.

Yukarıda, esnek robotik hücreler ve çalışma yapısı anlatıldıktan sonra bu alanda daha önce yapılmış çizelgeleme çalışmalarından bahsedilmiştir. Çalışmalarda görüldüğü üzere, çizelgeleme problemleri çözümü çok zor problemler olduğundan daha çok sezgisel algoritmalarla yararlanılmış ve uygun çözümler elde edilmiştir. Ancak bu çözümlerin birçoğu doğrulanmamıştır. Pahalı yatırımlardan önce doğrulanmış çözümlerin elde edilmesi ve bu çözümler arasından üretkenlik karşılaştırılması yapılması gerekir. Sezgisel algoritmalarla elde edilen uygun çözümlerin doğruluğunun araştırılmasında petri ağlarından yararlanılabilir. Aşağıda literatür araştırması, petri ağları ve bu alanla yapılmış çalışmalarla devam etmektedir.

2.3 PETRİ AĞLARI

Petri ağı, eş zamanlı ve paralel sistemlerin modellenmesini, performansının değerlendirilmesini sağlayan ve karmaşık sistemlerin görselleştirilmesini kolaylaştıran bir araçtır. Sistemdeki bilgi akışını gösteren matematiksel ve grafiksel bir özettir [25]. Bir petri ağı ile kullanıcılar genel sistemi grafiksel olarak görüntüleyebilir ve editör ile düzenleyebilir. Petri ağları ayrıca sistemin performansını analiz etmek, kod üretmek, sistemin benzetimini yapmak ve model kontrolü yapmak için kullanılabilir [6]. Petri ağları, yerleşim ve geçiş düğümlerinden oluşur. Yerleşimler, dairelerle; geçişler, dikdörtgenlerle gösterilir. Geçişler ve yerleşimleri birbirine bağlamak için, yerleşimlerden geçişlere ve geçişlerden yerleşimlere yönlü oklar kullanılır.

Petri ağlarında, sistem üzerindeki bilgi akışı izlenirken, izlenen bilgiyi temsil eden ve sistemin değişken davranışlarını incelemeye yarayan jetonlar mevcuttur; siyah içi dolu noktalarla gösterilir. Yerleşimler bir ya da daha fazla jeton barındırabilir ya da hiç jeton

barındırmayabilir. Bir yerleşimde jetonun olup olmayışı, bu yerleşimle sağlanan şartın doğru ya da yanlış olduğunu ifade edebilir [26].

Petri ağlarının parametreleri:

Petri ağları 5 parametreyle gösterilir [27], $N = (P, T, I, O, M_0)$

$P = \{p_1, p_2, \dots, p_m\}$ Yerleşimler kümesi,

$T = \{t_1, t_2, \dots, t_n\}$ Geçişler kümesi,

$I: P \times T \rightarrow N$, yerleşimlerden geçişlere yönlü okları tanımlayan giriş fonksiyonu,

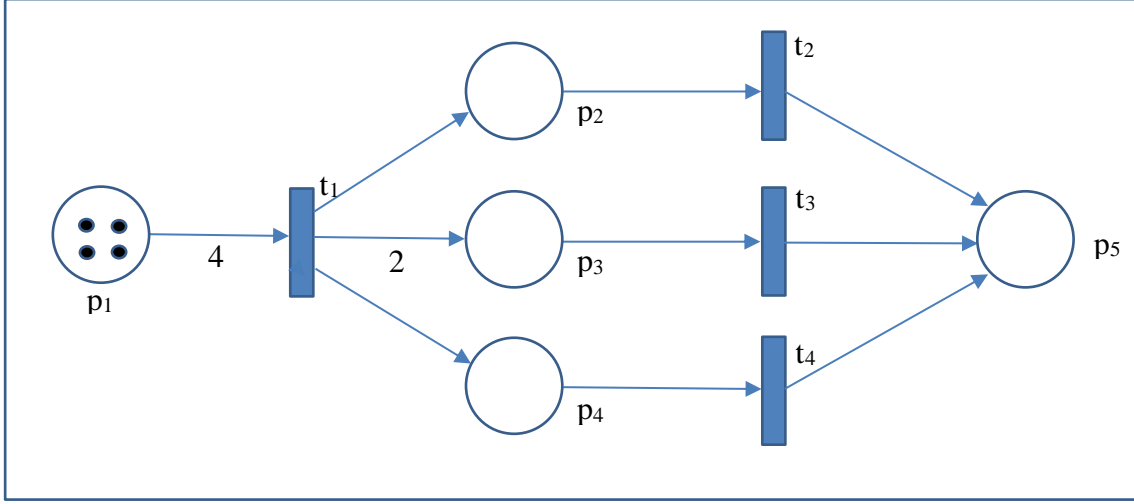
$O: T \times P \rightarrow N$, geçişlerden yerleşimlere yönlü okları tanımlayan çıkış fonksiyonu,

M_0 : Başlangıç işaretleme,

İşaretleme, yerleşimlerdeki jetonların dağılımını ifade eder. Petri ağı modelinin uygulanması sırasında, jetonların yerleşimlerdeki sayısı ve ikameti değişebilir.

Yerleşim p_j 'den geçiş t_i 'ye uzanan yönlü bir ok, $i= 1, 2, \dots, n$ ve $j= 1, 2, \dots, m$ için $I(t_i, p_j) = 1$ ile gösterilir. Bu durumda, p_j, t_i 'nin giriş yerleşimi olarak adlandırılır. Geçiş t_i 'den yerleşim p_j 'ye uzanan yönlü bir ok, $O(t_i, p_j) = 1$ ile gösterilir. Bu durumda, p_j, t_i 'nin çıkış yerleşimi olarak adlandırılır. $I(t_i, p_j) = k$ ya da $O(t_i, p_j) = k$ olduğunda, eşitlikteki k , okun ağırlığını ifade eder [26]. Yönlü okların ağırlıkları, bağlı oldukları geçişlerin etkin olup olmamasını etkiler.

Petri ağları konusunda yapılan pek çok çalışmada, ağın yapısının formal tanımı kullanılmaktadır. Petri net modelinin yapısının tanımını yapmak için de jetonların konumu ve sayısından yararlanır [28]. Aşağıda, Şekil 2.5'de verilen basit bir petri ağı modelinin grafiğiyle, petri ağlarının parametreleri gösterilmiştir.



Şekil 2.5. Basit bir petri ağı modeli

Şekil 2.5’de gösterilen petri ağı modelindeki parametreler:

$$P = \{p_1, p_2, p_3, p_4, p_5\};$$

$$T = \{t_1, t_2, t_3, t_4\};$$

$$I(t_1, p_1) = 4 \text{ ve } I(t_1, p_i) = 0, i=2, 3, 4, 5 \text{ için};$$

$$I(t_2, p_2) = 1 \text{ ve } I(t_2, p_i) = 0, i=1, 3, 4, 5 \text{ için};$$

$$I(t_3, p_3) = 1 \text{ ve } I(t_3, p_i) = 0, i=1, 2, 4, 5 \text{ için};$$

$$I(t_4, p_4) = 1 \text{ ve } I(t_4, p_i) = 0, i=1, 2, 3, 5 \text{ için};$$

$$O(t_1, p_2) = 1, O(t_1, p_3) = 2, O(t_1, p_4) = 1 \text{ ve } O(t_1, p_i) = 0, i=1, 5 \text{ için};$$

$$O(t_2, p_5) = 1 \text{ ve } O(t_2, p_i) = 0, i=1, 2, 3, 4 \text{ için};$$

$$O(t_3, p_5) = 1 \text{ ve } O(t_3, p_i) = 0, i=1, 2, 3, 4 \text{ için};$$

$$O(t_4, p_5) = 1 \text{ ve } O(t_4, p_i) = 0, i=1, 2, 3, 4 \text{ için};$$

$$M_0 = (4 \ 0 \ 0 \ 0 \ 0)^T.$$

M_0 ile verilen, sistemin ilk durumunu ifade eden başlangıç işaretleme, **geçiş (ateşleme) kuralıyla** değişir [29]. Ateşleme kuralı:

- 1) Eğer t geçişinin her bir p giriş yerleşimi, en azından p 'ye bağlı yönlü okun ağırlığına eşit sayıda jeton içeriyorsa, t geçişi **etkindir** denir. (**etkinlik kuralı**) Her p yerleşimi için $M(p) \geq I(t, p)$ şartı sağlanmalıdır.

- 2) Etkin bir geçiş, olayın gerçekten gerçekleşip gerçekleşmemesine göre ateşleyebilir ya da ateşleyemez.
- 3) Etkin geçiş t'nin ateşlemesiyle, geçişe giren okun ağırlığı kadar her bir p giriş yerleşimindeki jetonların sayısından azalma olur ve çıkış yerleşimindeki jetonların sayısı t geçişinden çıkan okların ağırlığı kadar artar.

Sadece geçişler etkinken ateşleme gerçekleşebileceğinden, geçişten önceki yerleşimlerde jeton olmadığında ateşleme gerçekleşemez; ateşleme geçişi olmayan bir jetonu yerleşimden eksiltmeyi deneyemez, bu nedenle yerleşimlerde jeton sayısı negatif olamaz [26].

Eğer bir p yerleşimi, t geçişinin hem giriş hem çıkış yerleşimiye, p ve t çifti *öz halka* olarak adlandırılır. Öz halka içermeyen petri ağlarının *saf (pure)* olduğu söylenir [27].

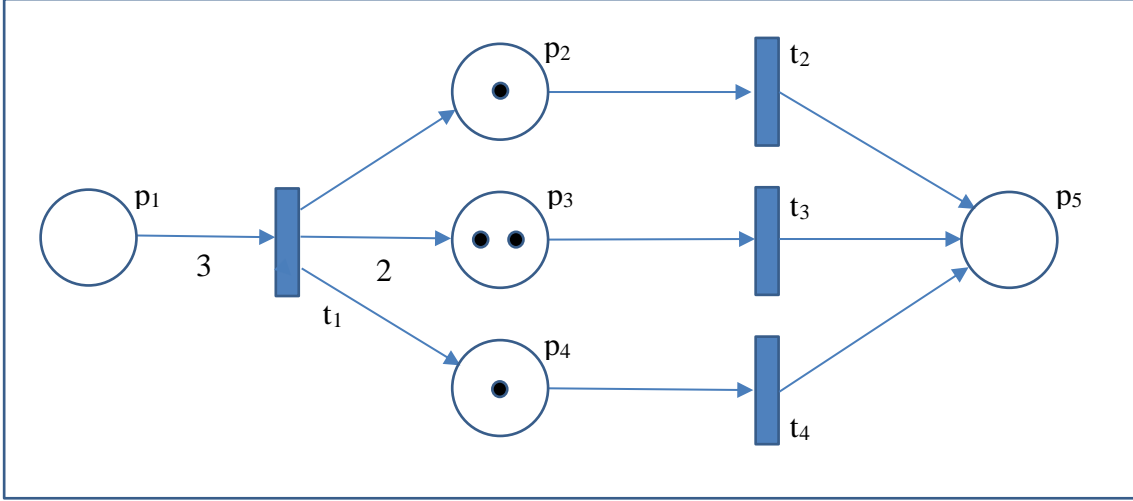
Ateşleme kuralını anlatmak için, Şekil 2.5'deki petri ağı modelini düşünelim. Etkin olma kuralına göre, tek etkin olan geçiş t_1 'dir. Çünkü sadece t_1 'e bağlı p_1 yerleşiminde, geçişe giren okun ağırlığına büyük eşit miktarda jeton bulunmaktadır. Diğer geçişler etkin değildir; çünkü girişlere giren okların ağırlığı 1 olmasına rağmen okların çıktığı yerleşimlerde bir ya da birden fazla jeton bulunmamaktadır.

Şekil 2.6'da t_1 'in ateşlemesi sonucu petri ağı modelinin durumu verilmiştir. Ateşleme kuralına göre, t_1 'de ateşleme gerçekleştiğinde, sistemin yeni durumunu ifade eden M_1 işaretlemesi, $M_1 = (0 \ 1 \ 2 \ 1 \ 0)$ olur. t_1 ateşlediğinde, t_1 'den çıkan okların ağırlığına göre t_1 'den sonraki yerleşimlere jeton eklenir. Örneğin p_3 yerleşimine giren okun ağırlığı 2 olduğundan, p_3 'e 2 jeton; p_2 ve p_4 'e giren okun ağırlığı 1 olduğundan, p_2 ve p_4 'e birer jeton eklenmiştir. Etkin olma kuralına göre, M_1 'de t_2 , t_3 ve t_4 'ün hepsi de etkindir.

Eğer t_2 ateşlerse, ateşleme kuralına göre, $M_2 = (0 \ 0 \ 2 \ 1 \ 1)$ olur.

Eğer t_3 ateşlerse, ateşleme kuralına göre, $M_3 = (0 \ 0 \ 1 \ 1 \ 2)$ olur.

Eğer t_4 ateşlerse, ateşleme kuralına göre, $M_4 = (0 \ 0 \ 1 \ 0 \ 3)$ olur.



Şekil 2.6. Geçiş t1 in ateşlemesi

Yukarıda verilen ateşleme kuralında, her bir yerleşimin sınırsız sayıda jeton barındırabileceği kabul edilmiştir. Bu tür petri ağlarına, *sınırsız kapasiteli ağlar* denilmektedir. Birçok sistemde de petri ağlarındaki yerleşimlerin her birinin tutabileceği jeton sayısına bir üst sınır koyulması düşünülür. Bu tip petri ağlarına *sınırlı kapasiteli ağlar* denir. Sınırlı kapasiteli ağlarda yerleşim p'nin kapasitesi $K(p)$ ile gösterilir. Bu ağlarda, t geçişinin etkin olabilmesi için, t ateşledikten sonra t'nin her bir çıkış p yerleşimindeki jetonların sayısı, $K(p)$ 'yi geçemez şeklinde ek şart da sağlanmalıdır. Kapasite kısıtlı kurala *katı geçiş kuralı* denirken, kapasite kısıtı olmayan kurala *zayıf geçiş kuralı* denir [30].

2.3.1 Petri ağlarıyla modellemede kullanılan temel kavramlar

Değişken, olay odaklı bir sistemde; aktiviteler tarafından gösterilen eş zamanlılık, karar verme, senkronizasyon ve öncelikler gibi tipik karakteristik özellikler, petri ağlarıyla etkili bir şekilde modellenebilir [31].

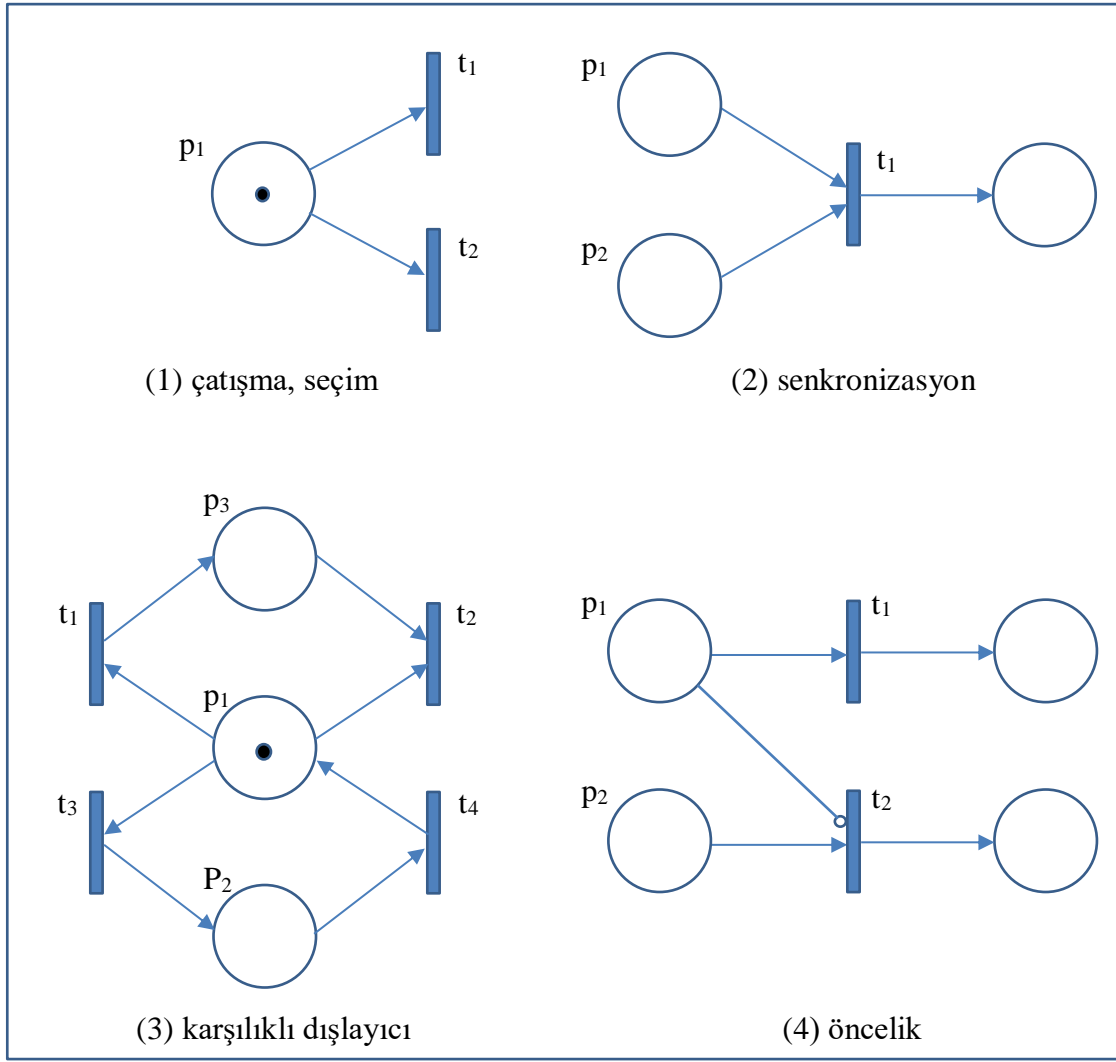
Eş zamanlılık: Bir geçişin eş zamanlı olabilmesi için iki ya da daha fazla çıkış yerleşimine sahip olması gerekir [26]. Şekil 2.3'de verilen petri net modelinde p₂, p₃ ve p₄ yerleşimleri eş zamanlıdır. t₂, t₃, t₄ yerleşimleri de eş zamanlı olarak etkindir, birinin ateşlemesi diğerinin durumunu değiştirmez. t₂, t₃, t₄'ün ateşlemesiyle paralel olarak üç olay gerçekleşmiş olur. Eş zamanlılık, sistemin etkileşimlerini gösterebilmek konusunda önemli katkı sağlar.

Sıralı yürütme: Şekil 2.3'deki modelde, t₁ ateşlemeden, t₂, t₃ veya t₄ ateşleyemez. t₂, t₃ veya t₄'ün ateşleyebilmesi için onlardan önce gelen geçiş t₁'in ateşlemesi gerekir. Böylece

sisteme öncelik kısıdı getirilmiş olur. Ayrıca sıralı yürütme özelliğiyle, deęişken bir sistemde çıktıları oluşurken, girdilerin hangi aşamalardan hangi sırayla geçtiğinin ifadesi mümkün kılınmış olur. Aktiviteler arasındaki nedensel ilişkiler modellenebilir.

Çatışma: Şekil 2.7(1)'deki modelde geçiş t_1 ve t_2 birbiriyle çatışma halindedir. İkisinin birden etkin olmasına rağmen, geçişlerden birinin ateşlemesi diğerrinin etkinliğini yitirmesine neden olur [28]. Böyle bir duruma örnek olarak, bir üretim sisteminde bir parçanın makinelerden birini seçme zorunluluğunu ya da makinenin parçaların arasından seçim yapmak zorunda kalmasını örnek verebiliriz. Ortaya çıkan çatışma, uygun olasılıkları çatışan geçişlere atayarak çözülebilir.

Senkronizasyon: Deęişken bir sistemde bir olay birden fazla kaynağa ihtiyaç duyabilir [26]. Şekil 2.7(2)'de görüldüğü gibi yerleşimlerle ifade edilen kaynaklar, geçişlerle toplanarak kaynakların senkronizasyonu sağlanabilir. Burada t_1 'in etkin olması için p_1 ve p_2 'nin jetona sahip olması gerekir. Bu yerleşimlere bir jetonun ulaşması, petri modelindeki geri kalan işlemlerin karmaşık bir şekilde sıralanmasına neden olabilir. Aslında t_1 geçişi, birleşen işlemleri modeller.



Şekil 2.7. Sistem özelliklerini göstermek için kullanılan petri net temellerinden bazıları

Karşılıklı dışlayıcılık: Kaynakların kullanılabilme kısıdından dolayı iki işlem aynı anda gerçekleştirilemiyorsa karşılıklı dışlayıcıdırlar [31]. Bu yapı Şekil 2.7(3)'te görülebilir. Karşılıklı dışlayıcılığa örnek olarak, bir üretim sisteminde aynı anda yalnızca bir makinede işlem yapabilen bir robotun iki makine tarafından yükleme ve boşaltma işlemleri için kullanılması verilebilir.

Öncelikler: Petri ağlarına önleyici bir yay eklenerek, ağ yapısına öncelik gösterme yetkinliği kazandırılıp modelleme gücü artırılabilir. Önleyici yay, ucunda küçük bir daire olan bir yayla gösterilir, bir giriş yerleşimiyle bir geçişi bağlar. Önleyici yayın varlığı, geçişin etkinlik durumlarını değiştirir. Bir petri net modelinde önleyici yay mevcut olduğunda; bir geçişin etkin olabilmesi için, geçişe bağlı giriş yerleşiminin en az geçişe bağlayan okun ağırlığına eşit sayıda jeton içermesi ve önleyici yayla geçişe bağlı olan hiçbir giriş yerleşiminde jeton bulunmaması gerekir [32]. Geçişin ateşleme kuralı aynıdır.

Fakat ateşleme, önleyici yayın bağlı olduğu yerleşimlerdeki işaretlemeyi değiştirmez. Şekil 2.7(4)'te görülen öncelik örneğinde, ucundaki küçük daireyle önleyici yay görülebilir. Şekilde, p_1 jeton içerdiğinde t_1 etkinken, t_2 'nin etkin olabilmesi için p_2 'nin jeton içermesi ve p_1 'in jeton içermemesi gerekmektedir. Bu da önceliği t_2 'den t_1 'e verir [26].

2.3.2 Petri Ağlarının Özellikleri

Petri ağı modellerinin özellikleri, ağ yapısına bağlıdır. Bu özellikler, davranışsal ya da yapısal olabilir. Davranışsal özellikler, ilk işaretlemeye bağlıdır. Ulaşılabilirlik, sınırlılık, karşılıklı dışlanma, canlılık gibi özellikler davranışsal özelliklerdendir. Yapısal özellikler, ilk işaretleme soyutlanmışsa ortaya çıkar. Örneğin, bir petri ağı herhangi bir ilk işaretleme için sınırlıysa, petri ağı yapısal olarak sınırlıdır; ya da herhangi bir ilk işaretleme için petri ağı canlı değilse, petri ağı yapısal olarak canlı değildir [6]. Eğer bir ilk işaretleme mevcutsa, petri ağı yapısal olarak canlıdır, ilgili sistem canlıdır.

Ulaşılabilirlik

Sistemlerin tasarımında önemli bir nokta, sistemin belirli bir duruma gelebilmesi ve istenilen belirli fonksiyonel davranışları gösterebilmesidir. Petri ağlarıyla modellenen sistemlerde de istenilen tüm özelliklerin ve belirli ihtiyaçların modelde görünmesi gerekir. Modelin başlangıç işaretlemesi M_0 'dan belirli istenen bir durumu ifade eden M_i işaretlemesine sistemin geçmesi için, geçişlerin nasıl bir sıralamayla ateşleneceği önemlidir. Ateşlemeler sırasında da ihtiyaç duyulan fonksiyonel davranışlar görünür. İşaretleme M_0 'dan işaretleme M_i 'ye taşıyacak ateşlemelerin sıralaması mevcutsa, M_i 'ye M_0 'dan ulaşılabilir denir. M_0 halinde etkin olan geçişler, olası ateşleme sıralamaları içerir. M_0 'dan ulaşılabilir olan tüm olası işaretlemelerin kümesi, ulaşılabilirlik kümesi olarak adlandırılır [32].

Canlılık

Petri ağlarında canlılık kavramı, daha çok çıkmaz durumlarla ilişkilidir [31].

Esnek Üretim Sistemlerinde Çıkmazlar

Çıkmaz, bir grup parçanın dairesel bekleme durumudur [31]. Örnek olarak, aynı grup içerisinde iki ya da daha fazla sürecin kaynaklara ulaşmak için sonsuz bir şekilde diğer diğer süreçleri beklediği durum verilebilir. Çıkmaz durumu, tüm sistemi bozup

çalışmasını durdurabilir. Çıkmaz durumunu düzeltmek için sistemi kapatıp işlenen parçayı kaldırıp yeniden başlatılır.

Bir çıkmazın oluşması için dört şart sağlanmalıdır [33].

- 1) Karşılıklı dışlama: Bir kaynak ya müsaittir ya da bu kaynağa bağlı bir işleme tahsis edilmiştir.
- 2) Tut ve bekle: Bir işlem daha fazla kaynağa ihtiyaç duyarken, ihtiyacından az sayıda kaynağı tutmaktadır.
- 3) Önlem yok: Bir işleme tahsis edilmiş kaynaklar, işlemin kendisi tarafından serbest bırakılıncaya kadar işlemde ayrılamaz.
- 4) Dairesel bekleme: Bir dizi halinde sıralanmış iki veya daha fazla işlem, her bir işlemin zincirdeki sonraki işlem/işlemler tarafından tutulan kaynak/kaynaklar için beklemektedir.

Örneğin, esnek bir üretim sisteminde bir makinenin tek bir palet kapasiteli giriş/çıkış ara stoku işlenmiş ürünlerle bir palet tutuyorsa ve üzerinde işlenecek malzemelerle başka bir palet daha gelmişse bir çıkmaz meydana gelir. (1. Şart) Burada paletleri ara stoka getirip ara stoktan götüren forklift tek bir paleti taşıma kapasitelidir, yine bir çıkmaza neden olur. (2. Şart) İşlenmiş parçalarla yüklü palet ara stoktan ayrılamaz. (3. Şart) İşlenecek malzemelerle yüklü palet de forkliftten ara stoka bırakılamaz. (4. Şart) Ara stok ve forkliftin birer kaynak olarak görüldüğü bu sistemde çıkmaz oluşması için dört şart da sağlanmıştır.

Çıkmazlardan arınmış bir sistem canlı olmak zorundadır. Bu durumda, ilk işaretleme M_0 'dan ulaşılabilir olan tüm M işaretlemeleri için, bazı ateşleme sıralamaları takip edildiğinde ağdaki herhangi bir geçişi ateşlemek mümkün olacaktır [32].

Sınırlılık

Üretim sistemlerinde yerleşimler sıklıkla ürün ve araç stoklama alanlarını göstermek için kullanılırlar. Araç gereç stoklama alanlarına çok fazla araç gereç depolamaya çalışmak, donanımın zarar görmesine neden olabilir. Kontrol stratejileriyle bu stoklama alanlarında olası taşmalar önlenebilir. Petri ağlarının özelliklerinden sınırlılık kavramı, modellenen sistemde taşmaların varlığını belirlemeye yardımcı olur. Eğer herhangi bir p yerleşiminde, k negatif olmayan bir tam sayı olmak üzere, jetonların sayısı daima k 'dan küçük ya da k 'ya eşitse, o petri ağına *k sınırlı* denir. Yerleşimlerden herhangi birinin sınırsız sayıda jeton tutma ihtimali olduğunda, petri ağı *sınırsızdır* [32].

Güvenlilik

Bir petri ağı, eğer 1 sınırlıysa güvenlidir. Ağıdaki yerleşimlerden hiçbirinin bir taneden fazla jeton içerme ihtimali olmamalıdır [28].

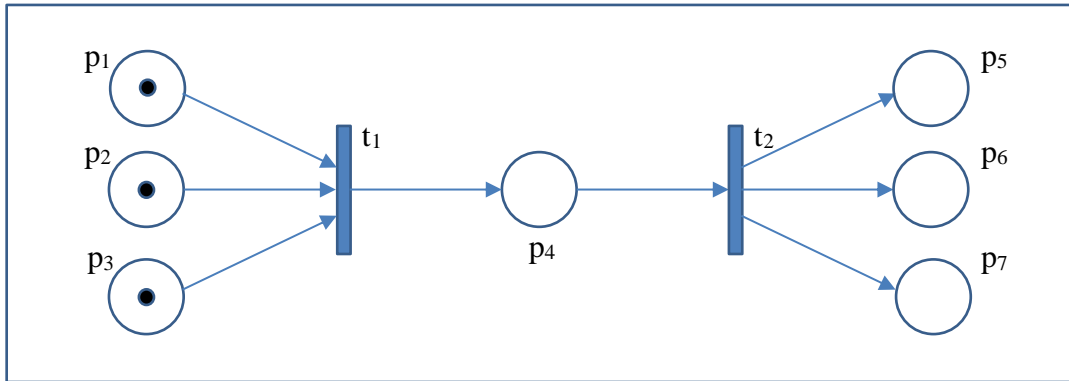
Tutuculuk

Gerçek sistemlerde, kullanımdaki kaynakların sayısı finansal sebepler nedeniyle genellikle kısıtlanmıştır. Petri ağlarında da eğer kaynakları göstermek için jetonlar kullanılıyorsa (sistemdeki kaynakların sayısı genellikle sabittir) modeldeki kaynakları belirten yerleşimlerde jetonların sayısı da değişmeden kalmalıdır. Bu, bir hüküm gelmediği sürece var olan kaynakların yok edilmeyeceği ve var olmayan kaynakların sistemde yaratılmayacağından ötürüdür. Örneğin, üretim hücresinde bir makine bozulduğunda, sistemin petri modelinde de makinelerden biri azaltılır.

Modeldeki ağırlıklı toplam jeton sayısı korunmuş ise, o petri ağı tutucudur [28]. Örneğin Şekil 2.8'deki petri ağı modelinde $M_0 = [1 \ 1 \ 1]$ ve jeton sayısı toplamı 3'tür. T_1 ateşlediğinde $M_1 = [0 \ 3 \ 0]$ ve jetonların toplamı 3 olacaktır. T_2 ateşlediğinde de $M_2 = [0 \ 0 \ 0 \ 3]$ ve jetonların toplamı yine 3 olacaktır. Modeldeki jeton sayısı korunduğundan, Şekil 2.8'deki petri ağı tutucudur.

Tersine çevrilebilirlik

Gerçek sistemlerin çalışmasında önemli bir nokta, bir hata görüldüğünde hatanın kurtarılabilmesidir. Sistemlerin hatalı durumdan önceki doğru çalışan duruma geri dönebilmesi gerekir. Petri ağlarının tersine çevrilebilirlik özelliği bu ihtiyaçlarla ilgilidir. Eğer bir petri ağında, her bir M işaretlemesinden M_0 ilk işaretlemesine dönülebiliyorsa, o petri ağına tersine çevrilebilir denir [27].



Şekil 2.8. Tutucu bir petri ağı örneği

2.3.3 Petri Ağlarıyla Modelleme

Petri ağları, yerleşim ve geçiş düğümleriyle bunlar arasında bulunan yönlü oklardan oluşur. Yerleşimler genellikle sistemin durumunu, şartları ya da kaynakları; geçişler olayları gösterir. Jetonların bir yerleşimde olması, yerleşimle ifade edilen şartın sağlandığı, jetonların yerleşimle ifade edilen bir kaynaktan olması kaynağın müsait olduğu anlamına gelir. Yerleşimler geçişlere yönlü oklarla bağlıdır ve geçişler giriş çıkış fonksiyonunu gösterirler. Bir geçişin ateşlemesi, bir olayın gerçekleşmesini ifade eder. Bir geçişin tüm giriş yerleşimlerinde yeterli miktarda jeton varsa, geçiş etkindir, ateşleyebilir. Başka bir deyişle, bir olayın gerçekleşebilmesi, gerekli tüm kaynaklar müsaitse ve şartlar sağlanıyorsa mümkündür. Petri ağları, ayrık olay sistemlerin modellenmesi, analizi ve tasarımı için uygun bir ortam sağlar [31]. Sıralı yürütme, eş zamanlılık, çatışma, senkronizasyon gibi durumları basitçe modelleyebilme kabiliyeti, petri ağlarını esnek üretim sistemlerini modellemek için uygun hale getirir.

2.3.4 Petri ağlarıyla daha önce yapılmış çalışmalar

Esnek üretim sistemlerinde petri ağları yoğun olarak kullanılmıştır, bu alanda yapılmış birçok çalışma mevcuttur. Li ve diğerleri [34] çalışmalarında, sistem parçalarının akışının açıkça gösterilebildiği süreç odaklı petri ağlarını kullanmışlardır. Kilitlenme önleyici çalışmaları yaparken petri ağlarına eklenen kontrol yerleşimlerinin sayısını azaltmak amacıyla yeni bir yöntem geliştirmişlerdir. Li ve diğerleri [35] çalışmalarında yine süreç odaklı petri ağlarıyla, karışık tam sayılı programlama tabanlı bir kilit keşif tekniği geliştirmişlerdir.

Kaynak odaklı petri ağlarında parçaların işlendiği üretim süreci, önceden tanımlanmış süreçler boyunca kaynaklara sıralı ulaşımına dayanır. Wu ve diğerleri [36] kaynak odaklı petri ağlarından yararlanarak montaj sistemlerinin modellenmesini yapmış ve bu sistemlerdeki kilitlenme probleminin el almışlardır. Modellere dayanarak bir kilit kontrol politikası önermişler ve hesaplama açısından etkinliğini kanıtlamışlar, bir vaka analiziyle desteklemişlerdir. Zhu ve diğerleri [37] çalışmalarında üretim sistemini, kaynak odaklı petri ağlarıyla modellemiş ve bu modele dayanarak bir birimlik en uygun çizelgelemenin bulunabildiğini kanıtlamışlardır.

Renkli petri ağları, farklı tipteki parçaların operasyon yerleşimlerinde bir arada karşılanmasını sağlarken, kaynak yerleşimlerinin de farklı çeşitteki kaynakları içermesini mümkün kılar. Shah ve diğerleri [38] sistemin analiz süresini kısaltmak amacıyla, devre

elemanlarının toplam sayısında azalma için üretim sisteminin petri ağıları modelinin sadeleştirilmesine odaklanmıştır. Esnek bir üretim sistemi modeli geliştirilerek, renkli petri ağılarıyla sistem üzerinde araştırma yapılmıştır. Önerilen model, birden fazla makine ve alete sahip geniş esnek üretim sistemlerinde genişletilip uygulanmaktadır. Modelle çeşitli senaryolar araştırılıp karşılaştırılmıştır. Geliştirilmiş renkli petri ağ yapısı, modeldeki toplam yer sayısını azaltan araçların gruplandırılmasına izin vermiştir. Aguiar ve diğerleri [39] çalışmalarında, bir uçak sanayii firmasının montaj hattında hücre davranışlarını modellemek için renkli petri ağlarını kullanmışlardır.

Nesne odaklı petri ağıları, mesajlarla ve bağlantılarla çeşitli işlem birimleri arasında birbirinden bağımsız çalışabilme ve birbirini tamamlama özelliği (modülerlik) kazandırır. Başak ve Albayrak [40] çalışmalarında nesne odaklı petri ağlarını kullanarak siparişe göre üretim hücrelerinde, çizelgeleme problemini analiz etmek ve modellemek için çalışmışlardır. Nesne odaklı petri ağları yaklaşımı, etkili bir üretim kontrol modeli kurmakta ve modelin performansını değerlendirmekte kullanılmıştır. Üretim sistemi, Petri ağlarının alt sınıflarından, zamanlanmış işaretli grafik olarak modellenmiş ve performans değerlendirme probleminin basit lineer bir probleme küçültülebileceği gösterilmiştir. Sunulan petri ağı tabanlı yöntem, Valeo-Türkiye firmasındaki esnek otomotiv üretim sistemi için, gerçek zamanlı bir çizelgeleme ve kontrolün modellenmesiyle gösterilmiştir.

İmalat sistemlerinin işleyişini iyileştirmek için bir paradigmanın getirilmesi durumunda, pahalı ve zaman alıcı bir uygulamadan önce sistemin performansını tahmin etmek önemlidir. Long ve diğerleri [9] çalışmalarında üç makineli bir esnek üretim sisteminin modellenmesi için genişletilmiş renkli stokastik petri ağlarını kullanmışlardır. Sonuçlar bu modelin esnek üretim benzetiminde kullanılabileceğini göstermiştir. Üretim sisteminin üretkenliği kuyruk kullanılmayan ve kuyruk kullanılan modellerde karşılaştırılmış ve kuyruk kullanılmayan modelin üretkenliğinin daha yüksek olduğu görülmüştür.

Jeton odaklı petri ağı modelinde, ateşleme şartlarının sağlanıp sağlanmadığını belirlemek için bir jetonun hareket edebileceği yerler tespit edilir [41]. Kilitlenmeden kaçınmak için, geçiş durumunda jeton önce kaynaktan yararlanma önceliğini tahmin eder. Makineler, parçalar, robot jeton olarak tanımlanabilir.

Zamanlı petri ağları, belli bir zamanı olan olayların belirli sıralarla gösterilmesini sağlar. Geçişler gerçek zamanda ateşler, ateşlemenin başlamasıyla jetonlar giriş stokundan çıkar, ateşlemenin bitmesiyle jetonlar çıkış stokuna gelir. Zamanlı petri ağlarının her durum tanımı, jetonların yerleşimlere dağılımını hesaba katmak zorundadır. Zhang ve diğerleri [42] esnek bir montaj sisteminde en uygun üretim sıralaması için çizelgeleme modeli geliştirmişlerdir. Sistemde birlikte çalışan makinelerin her biri parçaların üretilmesinde gerekli operasyonların bir kısmını yapabilmektedir. Çalışmadaki amaç, operasyonların makinelere uygun dağılımını yaparken aynı zamanda ürün ya da ürünlerin tamamlanma süresini minimize etmektir. Montaj süreci, zamanlı petri ağlarından yararlanarak modellenmiş ve görev çizelgelemesi, dinamik programlama algoritması tarafından çözülmüştür. Yöntemin ihtiyaç duyulan zamanı doğru bir şekilde hesapladığı görülmüştür.

Zamanlı petri ağları, ayırık olay dinamik sistemlerini modellemek, çizelgelemek ve analiz etmek için yaygın olarak kullanılmıştır [43]. Çalışmada, özellikle otomatik üretim sistemleri için çevrim süresini minimize etmek amaçlı, zamanlı petri ağlarının döngüsel çizelgeleme problemleri incelenmektedir. Her çatışma yerinde uygun jeton yönlendirmesi, bir TPN'in aynı ateşleme sırasını tekrarlamasını sağlayabilir. Çizelgeleme problemini çözmek için etkili bir dal sınır algoritması geliştirmişlerdir. Ayrıca, zamanlı petri ağlarından zamanlı olay grafiklerine geçiş prosedürü ve arama ağacındaki her bir düğüm için çevrim süresinin alt sınırları bulunmuştur. Algoritmanın verimliliği bir robotik hücrede doğrulanmıştır.

Fathian ve diğerleri [44] çalışmalarında, iki makine ve bir robottan oluşan, parçaların iki makinede de işlenmek zorunda olduğu bir hücre düşünmüşlerdir. Makinelerin işleme süreleri aynıdır, robotun yükleme ve boşaltma süreleri tüm şartlarda sabittir, robot bir çizgi üzerinde hareket etmektedir. Robotun iki ardışık hücre konumu arasındaki ulaşım süreleri eşittir. İki makineli hücrelerde literatürde var olan robot hareketlerinin döngüleri incelenmiş ve çevrim süresini kısaltmak amacıyla robot hareketlerinin sırası için yeni bir hareket döngüsü sunmuşlardır. Petri ağlarına dayanan, çevrim süresinin daha kısa olduğu, yeni bir matematiksel model ortaya koymuşlardır.

Çalışma, robot hareketlerinin çizelgeleme problemi için zamanlı petri ağlarından yararlanarak bir en uygun çözüm getirmektedir [45]. Robotun hücre istasyonları arasındaki tüm mümkün robot hareketlerini hesaba katan bir matematiksel model oluşturmak için, zamanlı petri ağları yaklaşımı kullanılmıştır. TPN geçişlerinin en iyi

ateşleme sıralamasını belirlemek amacıyla, en son geçişin ateşlenmesinden önce geçen zamanı minimize etmek için, elde edilen matematiksel model çözülmüştür. Geçişlerin en uygun sıralaması, nihai TPN modelinin oluşturulmasında kullanılmıştır.

Pham ve Dang [46] çalışmalarında, esnek üretim sistemleri için zamanlı petri ağı modellerine dayanan, yeni bir en iyi çizelgeleme yöntemi geliştirmişlerdir, sezgisel arama yöntemiyle birleştirilmiştir. Esnek üretim sistemi için bir petri ağı modeli oluşturulduktan sonra, petri ağının ulaşılabilirlik grafiği üzerinde en uygun çizelgeleme aranıp belirlenmiştir. En iyi sonuçlara en hızlı şekilde ulaşmak için, bir sezgisel arama algoritması geliştirilmiştir. Algoritmanın kullanılabilirliği bazı denemelerle test edilmiştir.

Barenji ve diğerleri [47] çalışmalarında bir montaj hattında kit besleme metodunun avantajını ölçmek için petri ağlarını kullanmışlardır. Kaynak odaklı ve süreç odaklı modelleme yaklaşımlarının bütünleştirilmesiyle petri ağları geliştirilmiştir. Daha sonra mevcut ve önerilen sistemin detaylı bir nicel analizi yapılarak metodun kullanıma uygun olduğu görülmüştür.

Esnek üretimlerde meydana gelebilecek problemlerden biri kilitlenmedir. Kilitlenme, bir veya daha fazla işlemin, diğer işlemler tarafından kullanılan kaynakların beklenilmesi nedeniyle, görevlerini tamamlayamadığı durumdur. Bu, kaynakların uygunsuz bir şekilde paylaşılması ya da paylaşılan kaynakların farklı görevlere verimli bir şekilde atanmasının karmaşıklığı nedeniyle uygun olmayan kaynak ayırma mantığından kaynaklanabilir. Taha ve diğerleri [48] çalışmalarında kilitlenmeleri modellemek ve tespit etmek için petri ağlarını kullanmışlardır. Matlab yazılımı, bir paylaşım makinesi ile iki paralel hattaki kilitlenmeyi tespit etmek için kullanılmıştır. Sonuçta, makine birde işlenmek için bekleyen parçalardan kaynaklandığı ortaya çıkmıştır.

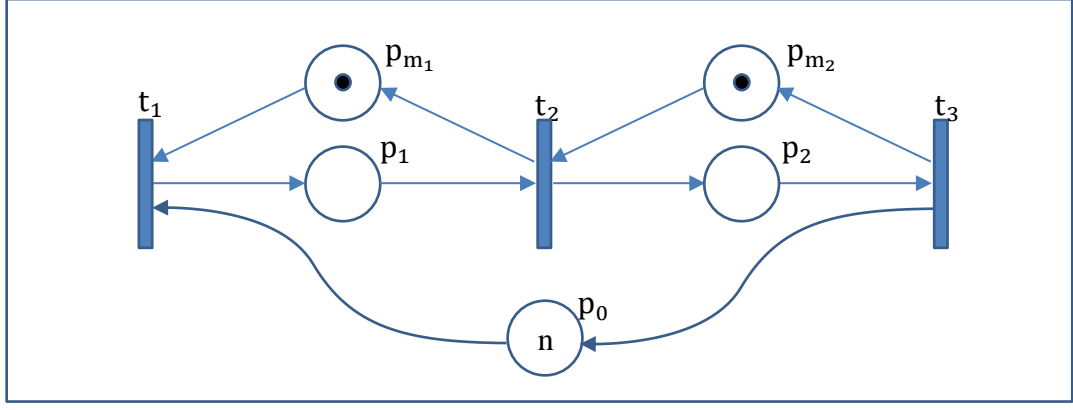
Kesikli ve sürekli işlemlerin birleşimi, ham petrol işleme operasyonlarının kısa dönemli çizelgeleme problemini tümleşik hale getirmektedir. Bu nedenle tam olarak en iyi olandan başka tatmin edici çözümler bulunmalıdır. Bunu yapmak için, şarj kaynakları bir tür kritik kaynak olduğundan kaç tane şarj tankının gerekli olduğuna karar vermek çok önemlidir. Zhang ve diğerleri [49] ham petrol işlemlerinin davranışını tanımlamak için hibrid petri ağlarını kullanmışlardır. Her bir damıtma için iki şarj tankı varsa, iki veya daha fazla damıtma sistemi bulunan bir sistem için maksimum üretkenliği elde etmek amacıyla uygun bir program bulunabileceğini göstermişlerdir. Uygulanabilir bir program elde etmek için başlangıç durumunun gerekleri belirtilerek zamanlama yöntemi önerilmiştir.

3 YÖNTEM

3.1 SÜREÇ ODAKLI PETRİ AĞI MODELLEME

Esnek üretim sistemlerini süreç odaklı yöntemle modellemek için üç aşama mevcuttur. Bunlar: üretim süreçlerinin sıralanmasının modellenmesi, kaynakların modellenmesi ve operasyonlar için kaynak ihtiyaçlarının modellenmesidir [50]. Her bir parça tipinin üretim süreçlerini modellemek için, önce parçayı işlemekte gerekli aktiviteler belirlenir. Aktiviteler, işlemlerin gerçekleştirilmesini ve depolamasını içerir. Her bir aktivite, operasyon yerleşimi denilen bir yerleşimle modellenir. Daha sonra bu yerleşimler, ilgili işlemlerin sırasına göre geçişlerle bağlanır. Esnek üretim sistemlerindeki kaynakları modellemek için, içinde jetonlar bulunan yerleşimler eklenir. Bu tip yerleşimlere, kaynak yerleşimi denir. Eğer bir p yerleşimiyle modellenmiş bir işlem, bir kaynak yerleşimine ihtiyaç duyuyorsa, kaynak yerleşimi p_r 'dan p 'nin her bir giriş geçişine ve her bir p 'nin çıkış yerleşiminden p_r 'a yönlü bir ok konulur. Böylece, bir süreç odaklı petri ağı (POPN) modeli kurulmuş olur [50].

Süreç odaklı petri ağı modelleme metodunu daha iyi açıklamak için, Şekil 3.1'de basit bir üretim sisteminin süreç odaklı yöntemle modellenmiş gösterimi verilmiştir. Bir A ürününün üretilmek için iki operasyona ihtiyaç duyduğunu ve bu operasyonlardan birinin m_1 makinesine, diğerinin m_2 makinesine ihtiyaç duyduğunu düşünelim. Şekil 3.1'de birinci operasyon p_1 ile, ikinci operasyon p_2 ile, giriş ve çıkış stoku p_0 ile gösterilmiştir. m_1 makinesi p_{m_1} kaynak yerleşimi ile m_2 makinesi p_{m_2} kaynak yerleşimi ile ifade edilmiştir. p_{m_1} ve p_{m_2} makineleri kaynak olduklarından, içlerinde bir jetonla çizilmiştir ve bağlı oldukları operasyon yerleşimlerinin giriş geçişlerine bir okla bağlanmıştır. Operasyon yerleşimlerinin çıkış geçişlerinden de kaynak yerleşimlerine yönlü oklar girmiştir.



Şekil 3.1. Süreç odaklı petri ağı (POPAN) modeli örneđi

A ürününün üretim işlemleri p_0 , t_1 , p_1 , t_2 , p_2 , t_3 , p_0 sırası ile modellenmiştir. m_1 ve m_2 kaynaklarının kullanımı ise sırasıyla, p_{m_1} , t_1 , p_1 , t_2 , p_{m_1} ve p_{m_2} , t_2 , p_2 , t_3 , p_{m_2} ile ifade edilmiştir. p_0 'daki n sayısı, yerleşimdeki jetonların sayısıdır ve stoktaki A ürününün miktarını gösterir.

3.1.1 Malzeme yükleme-boşaltma işlemlerinin POPAN ile modellenmesi

Otomatik makine sistemlerini süreç odaklı petri ağılarıyla modellerken, sistemdeki yükleme-boşaltma-taşıma işlemlerini yapan robotlar kullanılır, robotlar da bir kaynak türüdür [50]. Malzeme yükleme-boşaltma işlemlerinin süreç odaklı yöntemle modellenmesini anlatırken, Şekil 3.1'de POPAN modeli gösterilen üretim sistemine ek olarak A ürününü üretmek için üçüncü bir operasyona daha ihtiyaç olduğunu ve bu operasyonu gerçekleştirmek için m_3 makinesi kullanıldığını düşünelim. m_1 , m_2 ve m_3 makineleri otomatik olarak yüklenmekte ve bitmiş ürünler makinelerden bir robot tarafından boşaltılmaktadır. İkinci makineye bağlı bir ara stok mevcuttur ve en fazla 3 adet malzeme alabilmektedir. Bu sistemin süreç odaklı petri ağı modeli Şekil 3.1'de gösterilmiştir. Şekildeki yerleşim ve geçişlerin tanımları aşağıda Çizelge 3.1'de verilmiştir.

Çizelge 3.1. Şekil 3.2'deki geçişlerin ve yerleşimlerin tanımları

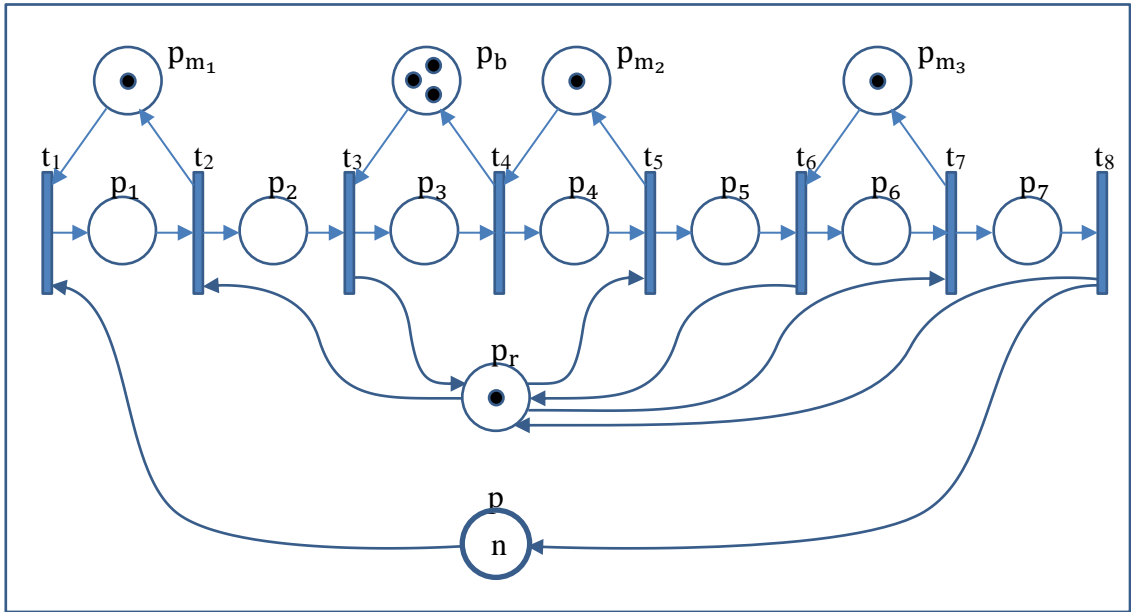
Yerleşimler		Geçişler	
p_0	Stokta parçaların beklemesi	t_1	m_1 'e bir parçanın yüklenmesi
p_1	Operasyon 1, m_1 makinesi tarafından parçanın işlenmesi	t_2	Robotun m_1 'den bir parça alması

Yerleşimler		Geçişler	
p_2	Robotun bir parçayı taşıması	t_3	Robotun ara stoka bir parça koyması
p_3	Ara stokta parçanın beklemesi	t_4	m_2 'ye ara stoktan parça yüklenmesi
p_4	Operasyon 2, m_2 makinesi tarafından parçanın işlenmesi	t_5	Robotun m_2 'den bir parça alması
p_5	Robotun bir parçayı taşıması	t_6	Robotun m_3 'e bir parça koyması
p_6	Operasyon 3, m_3 makinesi tarafından parçanın işlenmesi	t_7	Robotun m_3 'ten bir parça alması
p_7	Robotun bir parçayı taşıması	t_8	Robotun stoka bitmiş bir parça koyması
p_{m_1}	m_1 makinesinin hazır olması		
p_{m_2}	m_2 makinesinin hazır olması		
p_{m_3}	m_3 makinesinin hazır olması		
p_b	Ara stokun müsait olması		
p_r	Robotun müsait olması		

Şekil 3.1'deki modelde p_{m_1} , p_{m_2} , p_{m_3} , p_b ve p_r kaynak yerleşimi olduğundan içlerinde jetonla çizilmiştir. p_b ara stokun müsait olmasını ifade ettiğinden ve en fazla üç parça alabileceğinden içerisinde üç jetonla gösterilmiştir. Makinelerin ve robotun yaptığı işlemleri ifade etmek için, kaynak yerleşimlerinden yapılan işlemin giriş geçişine bir ok girmektedir. İşlemin çıkış geçişinden de kaynak yerleşimlerine bir ok uzanmaktadır. A ürününün üretim işlemleri p_0 , t_1 , p_1 , t_2 , p_2 , t_3 , p_3 , t_4 , p_4 , t_5 , p_5 , t_6 , p_6 , t_7 , p_7 , t_8 , p_0 ile modellenmiştir. p_r kaynağının kullanımı p_r , t_2 , p_2 , t_3 , p_r ; p_r , t_5 , p_5 , t_6 , p_r ; p_r , t_7 , p_7 , t_8 , p_r ile tarif edilmiştir.

Şekil 3.2'ye bakıldığında, p_0 stokundan çıkan parçanın t_1 geçişiyle m_1 makinesinin hazır olması durumunda m_1 'e yüklendiği, p_1 ile operasyon 1'in gerçekleştirileceği ve robotun müsait olmasıyla t_2 'den bir parça alacağı, p_2 işlemi ile o parçayı taşıyarak t_3 geçişiyle

p_b müsait olduğunda ara stoka bırakacağı görülür. p_3 ara stokta bekleme işleminden sonra, t_4 geçişiyle p_{m_2} hazır olduğunda parça m_2 'ye yüklenir ve p_4 işlemiyle ikinci operasyon gerçekleşir. t_5 geçişi ile robot müsait olduğunda parçayı m_2 'den alır ve p_5 taşıma işlemi gerçekleştirerek p_{m_3} hazır olduğunda t_6 ile m_3 'e bırakır. m_3 makinesinde p_6 işlemiyle üçüncü operasyon gerçekleşir ve robot müsait olduğunda t_7 'den bitmiş parçayı alarak p_7 taşıma işlemi gerçekleştirip t_8 geçişiyle bitmiş parçayı p_0 stokuna bırakır.



Şekil 3.2. Robot ve makinelerden oluşan bir üretim sisteminin POPN modeli örneği

3.1.1.1 Süreç odaklı petri ağlarının karakteristikleri

Örneklere anlaşılabileceği üzere, süreç odaklı petri ağlarıyla, üretim sistemi açık ve düzgün bir şekilde modellenilebilir. Modelden sistemin işleyişinin anlaşılması kolaydır. POPN ile sistem, detaylı bir şekilde modellenilebilir [50]. Fakat parça çeşidi veya operasyon sayısı arttığında, petri ağı modeli oldukça genişler. Çünkü tüm kaynaklar ve onların tüm bağlantıları, modelde gösterilen sistemin diğer elemanlarıyla, yerleşim, geçiş ve ok sayısında büyük artışa neden olabilir. Bu da analiz yapmayı zorlaştırabilir [51].

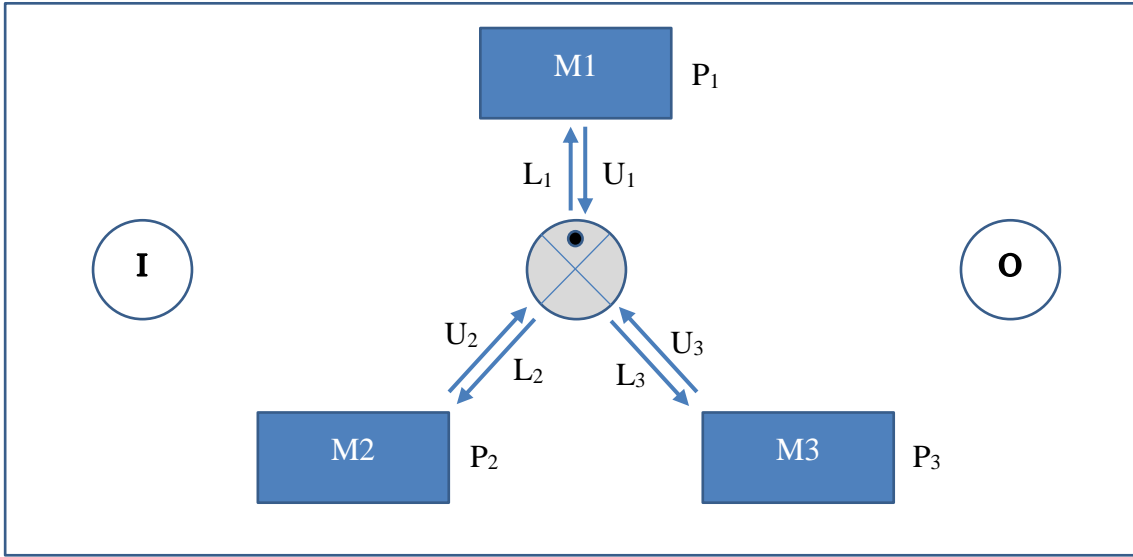
3.2 ARA STOKSUZ, TEK KISKAÇLI, ÜÇ MAKİNELİ FRC İÇİN PETRİ AĞI MODELİ OLUŞTURULMASI

Çalışmada ilk olarak makinelerin ara stok alanı olmayan, robotun tek kısıkaçlı olduğu bir esnek robotik hücre için petri ağı modeli kurulması gösterilecektir. Hücre, Şekil 3.3'de gösterildiği gibi bir giriş stoku, bir çıkış stoku, üç makine ve tek kısıkaçlı robottan

oluşmaktadır. M1, birinci makineyi; M2 ikinci makineyi, M3 üçüncü makineyi göstermektedir. Makinelerde sırasıyla özdeş P₁, P₂, P₃ özdeş işlemleri yapılmaktadır. I, giriş (hammadde) stokunu; O, çıkış (son ürün) stokunu ifade etmektedir. Ortadaki gri renkli daire ve içindeki tek jeton, tek kısıkaçlı robotu temsil etmektedir. Tek kısıkaçlı robot, başlangıç noktasından giriş stokuna gelir, tek kısıkaçıyla parçaları önce giriş stokundan alır; M1, M2, M3'ten birine götürerek yükleme işlemini yapar. İşlemler gerçekleştiikten sonra makinenin yanına tekrar gelerek bitmiş parçaların boşaltma işlemini gerçekleştirir ve çıkış stokuna bırakır. Makine meşgulken ikinci bir parça makineye yüklenemez.

L_i: Robotun i. makineye yükleme işlemi, i=1, 2, 3

U_i: Robotun i. makineden boşaltma işlemi, i=1, 2, 3

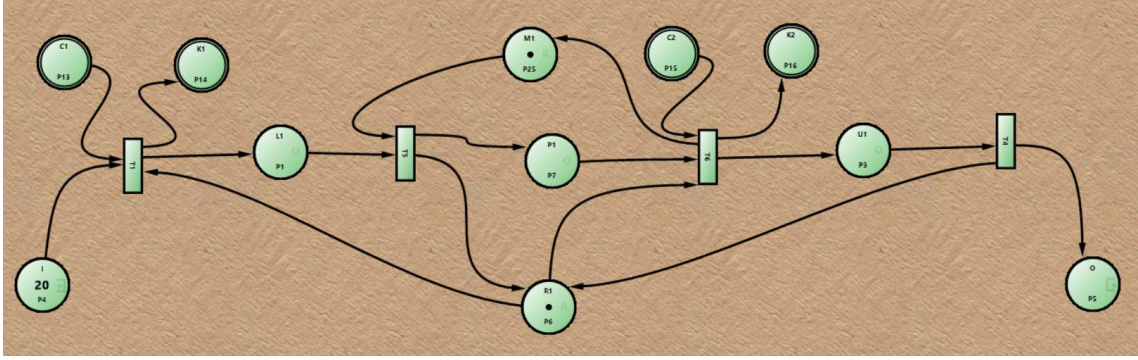


Şekil 3.3. Ara stok alanı olmayan, robotun tek kısıkaçlı olduğu model

Ara stok alanı olmayan tek kısıkaçlı robotun olduğu hücrenin yapısı, çalışma şekli ve hücrede gerçekleşen işlemler yukarıda açıklanmıştır. Hücrenin süreç odaklı petri ağı modelinin oluşturulması aşağıda üç adımda anlatılmıştır.

3.2.1 Üretim süreçlerinin sıralanmasının modellenmesi

Ara stok alanları olmayan, tek kısıkaçlı robotun olduğu esnek robotik hücredeki aktiviteler; işlenmemiş parçaların giriş stokunda beklemesi, M1, M2, M3 makinelerinde parçaların işlenmesi, işlenmiş parçaların çıkış stokunda beklemesinden oluşur. Bu aktiviteler operasyon yerleşimleri ile gösterilir. Şekil 3.4'de M1 makinesinde işlenen parçalar için hücrenin süreç odaklı petri ağı modeli verilmiştir.



Şekil 3.4. M1 makinesinde işlenen parçalar için süreç odaklı petri ağı modeli

Parçaların M1 makinesinde işlenmesi P1 yerleşimiyle, parçaların M1 makinesine yüklenmesi L1 yerleşimiyle, parçaların M1 makinesinden boşaltılması U1 yerleşimi ile gösterilmiştir. Şekildeki tüm geçiş ve yerleşimlerin tanımları Çizelge 3.2’de verilmiştir. Parçanın üretim işlemleri yerleşim ve geçiş düğümlerinin I-T1-L1-T5-P1-T6-U1-T4-O sırasıyla modellenmiştir.

Çizelge 3.2. Şekil 3.4’deki yerleşim ve geçiş düğümlerinin tanımları

Yerleşimler		Geçişler	
I	Giriş stokunda işlenmemiş parçaların beklemesi	T1	Robotun L1 yükleme işlemine başlaması
L1	Robotun M1 makinesini yüklemesi	T5	M1 makinesinin parçaları işlemeye başlaması
P1	M1 makinesinin parçaları işlemesi	T6	Robotun U1 boşaltma işlemine başlaması
U1	Robotun M1 makinesini boşaltması	T4	Robotun çıkış stokuna işlenen parçayı bırakması
O	Çıkış stokunda işlenmiş parçaların beklemesi		
M1	M1 makinesinin hazır, yeni bir işe başlayabilir olması		
R1	R1 robotunun hazır, yeni bir işe başlayabilir olması		

Yerleşimler		Geçişler	
C1	L1 yüklemesinin kontrolü, başlama şartı		
K1	L1 yüklemesinin gerçekleşme sayısı		
C2	U1 yüklemesinin kontrolü, başlama şartı		
K2	U1 boşaltmasının gerçekleşme sayısı		

3.2.2 Kaynakların modellenmesi

Üretim hücrelerinde kaynaklar, üretim süreçlerindeki aktivitelerin gerçekleşebilmesi için gerekli kapasite, araç gereç ve ihtiyaçlardan oluşturur. Ara stok alanları olmayan, tek kısıkaçlı robotlu esnek üretim hücresindeki kaynaklar, makine ve tek kısıkaçlı robot kapasitelerinden ibarettir. Makine, parçaların işlenmesi için; robot parçaların yüklenmesi ve boşaltılması işlemleri için gereklidir. Şekil 3.4’de Makine 1, bir parça işleme kapasitesine sahip olduğundan içinde bir tek jetonla modellenmiştir. Robot da tek kısıkaçlı olduğundan yine bir jetonla resmedilmiştir. Kaynak yerleşimlerinin içindeki jeton, kaynağın kullanıma uygunluk ve kapasite durumlarını gösterir. Örneğin, robot kaynağını ifade eden R1 kaynak yerleşiminin içinde jeton olması, robotun sıradaki aktivite için müsait olduğunu ifade eder. Eğer jeton görünmüyorsa, robot şu anda meşgul, yeni bir aktiviteyi gerçekleştirmek için hazır değil demektir. Makine kaynağını ifade eden M1 kaynağının içinde jeton varsa, makine yeni parça işlemek için müsait; jeton yoksa makine hâlihazırda bir parça işlemekte ve yeni bir parçayı daha işlemeye başlamak için hazır değil, anlamına gelir.

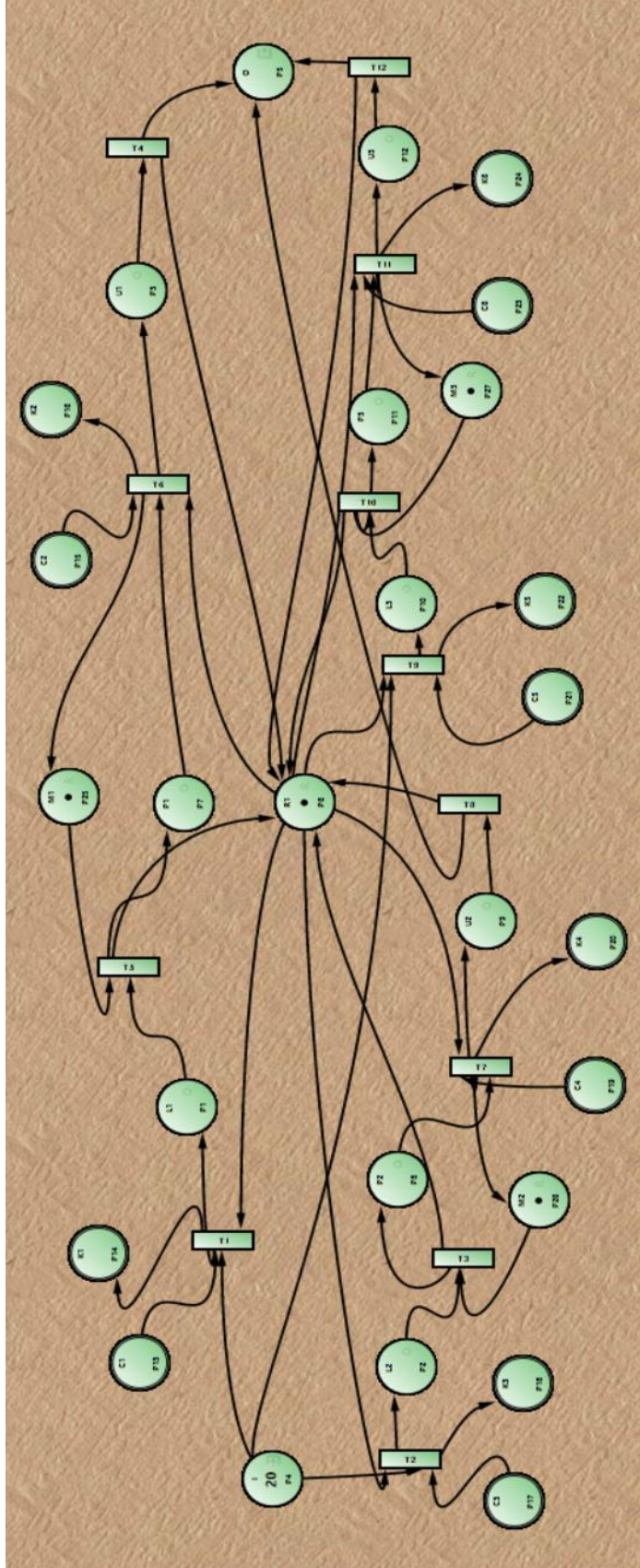
3.2.3 Operasyonlar için kaynak ihtiyaçlarının modellenmesi

Hücrede gerçekleşecek aktiviteler ve bunların gerçekleşebilmesi için gerekli kaynaklar belirlenip modellendikten sonra, kaynak yerleşimlerinin ihtiyaç duyulan operasyon yerleşimlerine bağlanması gerekir. Kaynak yerleşimlerinden yapılan işlemin giriş geçişine bir ok girer. İşlemin çıkış geçişinden de kaynak yerleşimine bir ok uzanır. Şekil 3.4’de M1 yerleşiminden P1 yerleşiminin giriş geçişi T5’e bir ok ve P1 yerleşiminin çıkış

geçışı T6'dan M1 yerleşimine bir ok uzatılmıştır. R1 yerleşiminden L1 yerleşiminin giriş geçışı T1'e ve L1 yerleşiminin çıkış geçışı T5'ten R1'e bir ok getirilmiştir. Yine, R1 yerleşiminden U1 yerleşiminin giriş geçışı T6'ya ve U1 yerleşiminin çıkış geçışı T4'ten R1'e bir ok gelir. R1 (robot) kaynağının kullanımı: R1-T1-L1-T5-R1 ve R1-T6-U1-T4-R1 sıralarıyla; M1 (Makine 1) kaynağının kullanımı: M1-T5-P1-T6-M1 sırasıyla modellenmiştir.

Tek bir makine için ara stok alanları olmayan, tek kısıkaçlı robotun olduğu esnek robotik hücrenin süreç odaklı petri ağlarıyla modellenmesi gösterilmiştir. Üç makine olduğunda da her bir makine için parça işleme, yükleme, boşaltma süreçleri, kaynaklar, operasyonların kaynak ihtiyaçları aynı şekilde modellenir. Şekil 3.5'de üç makineli hücre için süreç odaklı petri ağı modeli gösterilmiştir. Şekildeki tüm geçiş ve yerleşimlerin tanımları Çizelge 3.3'te verilmiştir. Her makine birbirinden bağımsız çalıştığı için, makinelerde gerçekleşen işlemler arasında ya da farklı makinelerdeki yükleme boşaltma aktiviteleri arasında bir bağ, ok bulunmamaktadır. Her makine için süreçler ayrı ayrı modellenir ve giriş stokundan L1, L2, L3 yükleme yerleşimlerine ayrı birer ok; U1, U2, U3 boşaltma işlemlerinden de çıkış stokuna ayrı birer ok çıkar. Robot ortak kaynakları olduğundan, her bir makinedeki yükleme ve boşaltma yerleşimlerine, kaynak ihtiyaçlarının modellenmesi yöntemine göre yine ayrı ayrı bağlanır.

Modelde görülen C1, C2, C3, C4, C5, C6, K1, K2, K3, K4, K5, K6 kontrol yerleşimleri petri ağı modelini çalıştırırken kolaylık sağlar. Hangi aktivitelerin hangi sırayla gerçekleştirileceği bu yerleşimlerle modellenir. Çünkü aktivitelerin çok sayıda farklı gerçekleşme sırası olabilir. Örneğin giriş stoku yerleşiminden çıkan işlenmemiş parçaların önce hangi makineye gideceği veya daha sonraki işlemin giriş stokundan makinelere yeni bir parça yüklenmesi mi yoksa işlenmiş parçanın boşaltılması mı olacağı bu kontrol yerleşimleriyle yazılan kurallar sayesinde belirlenir.



Şekil 3.5. Tek kısıkaçlı, ara stoksuz sistemin petri net modeli

C_i kontrol yerleşimleri, aktivitelerin kontrolünü sağlar, başlama şartını belirtir. K_i kontrol yerleşimleri, bağlı oldukları geçişlerin çıkış yerleşimlerindeki aktivitenin kaç defa gerçekleştiğini gösterir, kural yazımını kolaylaştırır. Bir geçişe C_i kontrol yerleşiminden bir ok uzanır ve yine aynı geçişten K_i kontrol yerleşimine bir ok girer. Böylece, C_i yerleşimi geçişin giriş yerleşimiyken K_i yerleşimi geçişin çıkış yerleşimi olur.

Çizelge 3.3. Şekil 3.5’deki yerleşim ve geçiş düğümlerinin tanımları

Yerleşimler		Geçişler	
I	Giriş stokunda işlenmemiş parçaların beklemesi	T1	Robotun L1 yükleme işlemine başlaması
L1	Robotun M1 makinesini yüklemesi	T5	M1 makinesinin parçaları işlemeye başlaması
P1	M1 makinesinin parçaları işlemesi	T6	Robotun U1 boşaltma işlemine başlaması
U1	Robotun M1 makinesini boşaltması	T4	Robotun çıkış stokuna M1’de işlenen parçayı bırakması
O	Çıkış stokunda işlenmiş parçaların beklemesi		
M1	M1 makinesinin hazır, yeni bir işe başlayabilir olması		
R1	R1 robotunun hazır, yeni bir işe başlayabilir olması		
C1	L1 yüklemesinin kontrolü, başlama şartı		
K1	L1 yüklemesinin gerçekleşme sayısı		
C2	U1 yüklemesinin kontrolü, başlama şartı		
K2	U1 boşaltmasının gerçekleşme sayısı		

Yerleşimler		Geçişler	
		T2	Robotun L2 yükleme işlemine başlaması
L2	Robotun M2 makinesini yüklemesi	T3	M2 makinesinin parçaları işlemeye başlaması
P2	M2 makinesinin parçaları işlemesi	T7	Robotun U2 boşaltma işlemine başlaması
U2	Robotun M2 makinesini boşaltması	T8	Robotun çıkış stokuna M2'de işlenen parçayı bırakması
M2	M2 makinesinin hazır, yeni bir işe başlayabilir olması		
C3	L2 yüklemesinin kontrolü, başlama şartı		
K3	L2 yüklemesinin gerçekleşme sayısı		
C4	U2 yüklemesinin kontrolü, başlama şartı		
K4	U2 boşaltmasının gerçekleşme sayısı		
		T9	Robotun L3 yükleme işlemine başlaması
L3	Robotun M3 makinesini yüklemesi	T10	M3 makinesinin parçaları işlemeye başlaması
P3	M3 makinesinin parçaları işlemesi	T11	Robotun U3 boşaltma işlemine başlaması
U3	Robotun M3 makinesini boşaltması	T12	Robotun çıkış stokuna M3'de işlenen parçayı bırakması
M3	M3 makinesinin hazır, yeni bir işe başlayabilir olması		

Yerleşimler		Geçişler	
C5	L3 yüklemesinin kontrolü, başlama şartı		
K5	L3 yüklemesinin gerçekleşme sayısı		
C6	U3 yüklemesinin kontrolü, başlama şartı		
K6	U3 boşaltmasının gerçekleşme sayısı		

4 ANALİZLER VE TARTIŞMALAR

Bu kısımda Petri ağlarından yararlanarak robotu tek kısıkaçlı, ara stok alanlarına sahip esnek robotik hücrelerle robotu çift kısıkaçlı, ara stok alanı olmayan hücrelerin üretkenliğinin karşılaştırılması yapılmıştır.

Günümüzde üreticiler, yüksek kaliteli ürünleri daha az maliyetle, daha verimli bir şekilde üretmenin yollarını aramaktadırlar. Esnek robotik hücreler verimliliği artırırken, makinelerin esnekliği, bir makinede çok çeşitli parçalar üretilebilmesi, robotların geniş aralıktaki boyutta ve ağırlıkta malzeme taşıyabilmesi, esnek robotik hücrelere çok çeşitli konfigürasyonlarda üretim yapabilme yeteneği kazandırmıştır.

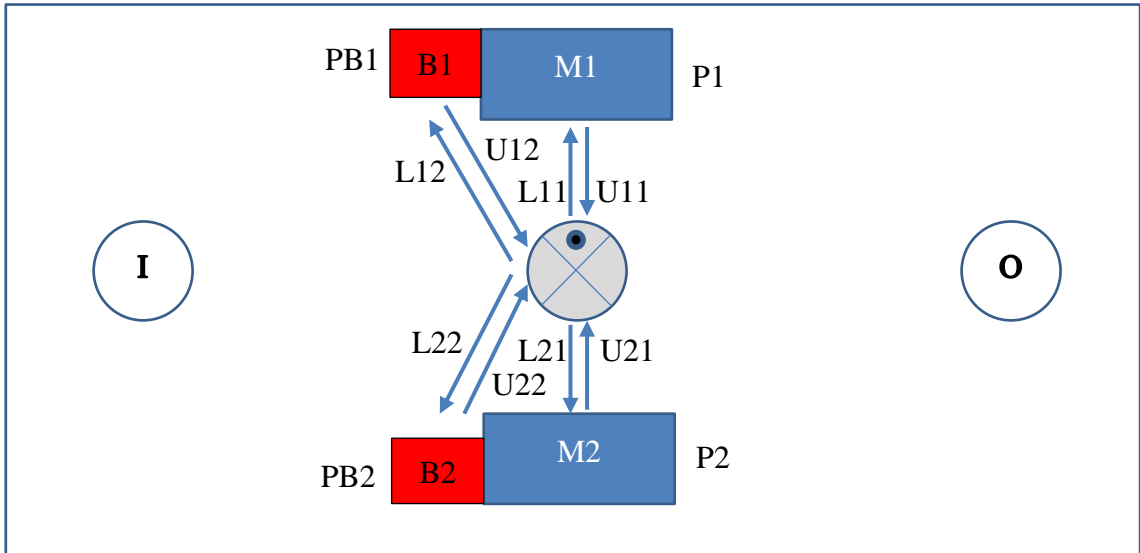
Esnek robotik hücrelerde genellikle bir makine bir parça işleme kapasitesine sahip olduğundan ikinci bir parça makineye yüklenemez. Ara stok alanı olmayan hücrelerde robot, makinelere parçaları yükledikten sonra parçaların makinedeki işlemleri tamamlanana kadar bekler, robotun boş kalma süresi artar. Yine robot, makinedeki parçayı boşalttıktan sonra yeni bir parça getirip yükleyene kadar da makine boş kalır, makine kullanım oranı azalır. Başka bir durumda da esnek robotik hücrelerde, makine sayısı arttıkça robotun meşgulliyeti artar. Bu da makinelere parça yükleyip boşaltma sıklığını azaltır. Ya işlenmiş parçalar makinelere boşaltılmayı bekler ya da parçalar boşaltıldıktan sonra makineler yeni parça yüklenmesini bekler, makinelerden yararlanma oranları azalır, sistemin üretkenliği düşer. Çift kısıkaçlı bir robot ise, makinenin yanına geldiğinde tek kısıkaçlı makinedeki bitmiş ürünü boşaltırken, diğer kısıkaçlı tuttuğu işlenmemiş parçayı makineye yükleyebilir. Bunun için çift kısıkaçlı robotun olduğu esnek üretim hücreleri de verimliliğin artırılmasında önemli bir seçenek olmuştur. Bunlardan dolayı, bir yönetici bakış açısıyla çift kısıkaçlı robota mı yoksa ara stokları olan makinelere mi yatırım yapılmasının hücredeki verimliliği daha fazla artıracığını değerlendirmek oldukça önemlidir.

4.1 Robotu tek kısıkaçlı, ara stok alanlarına sahip iki makineli FRC

Robotu tek kısıkaçlı, ara stok alanlarına sahip iki makineli bir esnek robotik hücre modeli Şekil 4.1'de gösterilmiştir. Bu modelde, her bir makineye bağlı bir ara stok alanı bulunmaktadır, ara stok alanları bir birim kapasitelidir. Ara stok alanı, makine meşgulken robot geldiğinde makineye yükleme yapamadığından, robotun işlenmemiş parçayı makineye yüklenmesi amacıyla bıraktığı bir bekleme alanıdır. Robot, giriş stokundan aldığı hammaddeleri ya makinelere bağlı ara stoklara yükler ya da direkt olarak makineye

yükler [5]. Daha sonra makineden boşaltma işlemini yaptığı ürünleri çıkış stokuna bırakır. Ara stoklardan makinelere yükleme işlemi, ara stokların akıllı aksamı tarafından otomatik olarak gerçekleştirilir. Her makine, parçanın ihtiyaç duyduğu tüm işlemleri gerçekleştirir ve her makinenin tek parça işleme kapasitesi vardır. Her parça yalnızca hücredeki bir makineye uğrar.

Ara stoklar şekilde sırasıyla B1 (birinci makineye bağlı ara stok), B2 (ikinci makineye bağlı ara stok), olarak verilmiştir. Ara stoklardan makinelere yükleme işlemleri de birinci ara stok için PB1, ikinci ara stok için PB2 ile gösterilmiştir. Özdeş M1 ve M2 makineleri, özdeş P1 ve P2 işlemlerini yapmaktadır. I, giriş stokunu; O, çıkış stokunu ifade etmektedir. Şekilde içinde içi dolu bir nokta olan gri daire tek kısıkaçlı robotu, içi dolu nokta tek kısıkaçı temsil etmektedir.



Şekil 4.1. Robotu tek kısıkaçlı, ara stok alanları olan model

Robotun gerçekleştirdiği aktiviteler:

L11: Robotun birinci makineye birinci parçayı yükleme işlemi

L21: Robotun ikinci makineye birinci parçayı yükleme işlemi

L12: Robotun birinci makineye ikinci parçayı yükleme işlemi

L22: Robotun ikinci makineye birinci parçayı yükleme işlemi

U11: Robotun birinci makineden birinci parçayı boşaltma işlemi

U21: Robotun ikinci makineden birinci parçayı boşaltma işlemi

U12: Robotun birinci makineden ikinci parçayı boşaltma işlemi

U22: Robotun ikinci makineden ikinci parçayı boşaltma işlemi

Robotun yaptığı yükleme ve boşaltma aktivitelerinin sıralamaları daha önce Nejad ve diğerlerinin [5] çalışmalarında makinenin işleme süresinin 30 saniye olduğu durum için L11 -L21-L22-L12-U11-U21-U12-U22 ve makinenin işleme süresinin 120 saniye olduğu durum için L11-U12-L21-L12-U22-U11-L22-U21 şeklinde belirlenmiştir. Bu sıralamalar burada süreç odaklı petri ağlarıyla modellenerek, sıralamanın ve ara stoklu modelin performansı incelenecektir.

4.2 Robotu çift kısıkaçlı, ara stok alanı olmayan iki makineli FRC

Çift kısıkaç konfigürasyonu, robot kolunun aynı anda birden fazla nesne taşımalarını sağlayarak daha karmaşık görevleri gerçekleştirmesine olanak tanır. Şekil 4.2’de esnek robotik hücrelerde kullanılan çift kısıkaçlar gösterilmiştir. Bu çift kısıkaçlar robotun tek koluna bağlıdır ve birbirinden bağımsız şekilde iki ayrı kısıkaç olarak çalışırlar. Robot kolunun üzerinde dönme kabiliyetleri vardır. Bir kısıkaç boşaltma işlemini yaptıktan sonra robot kolun üzerinde dönerek aşağı iner ve diğer kısıkaç onun yerine yukarı çıkarak yükleme işlemini gerçekleştirir.

İki kısıkaçlı robotlu üretim sistemlerinde robot, işlenmiş parçayı boş kısıkaçıyla makineden alır ve yerine diğer kısıkaçıyla tuttuğu işlenmemiş parçayı bırakır. Daha sonra işlenmiş parçayı çıkış stokuna götürüp bırakır. Çıkış stokundan giriş stokuna gelir ve tek kısıkaçıyla işlenmemiş yeni bir parça alarak makineye tekrar gider. Yine bitmiş ürünü makineden boşaltarak, işlenmemiş parçayı makineye yükler. Böylece makinenin boş kalma süresi azalır ve çıktı miktarı, dolayısıyla üretkenlik artar.



Şekil 4.2. Esnek robotik hücrelerde kullanılan çift kısıkaç örneği³

Şekil 4.3'te çalışmada kullanılan iki makineli, çift kısıkaçlı robotun olduğu esnek robotik hücre modeli verilmiştir. Robotun iki kısıkaçı olduğundan aynı anda iki parça taşıyabilir, makinelerin yanına geldiğinde tek kısıkaçıyla boşaltma yaptıktan sonra diğer kısıkaçıyla yükleme işlemi yapabilir. Şekilde yine I, giriş stokunu, O, çıkış stokunu; M1 ve M2 sırasıyla birinci ve ikinci makineleri ifade etmektedir. Ortadaki gri renkli daire robotu temsil ederken, bu defa çift kısıkaçı ifade etmek için dairenin içinde iki jeton mevcuttur.

I1: Robotun giriş stokundan birinci parçayı boşaltması

I2: Robotun giriş stokundan ikinci parçayı boşaltması

L1: Robotun birinci makineyi yükleme işlemi

L2: Robotun ikinci makineyi yükleme işlemi

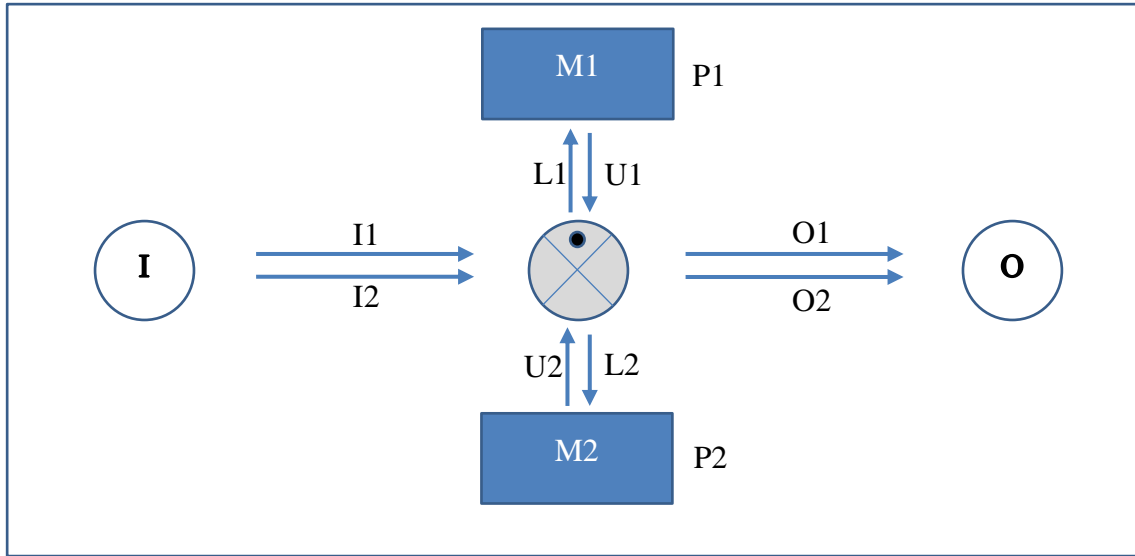
U1: Robotun birinci makineyi boşaltma işlemi

U2: Robotun ikinci makineyi boşaltma işlemi

O1: Robotun işlenmiş birinci parçayı çıkış stokuna bırakması

O2: Robotun işlenmiş ikinci parçayı çıkış stokuna bırakması

³ <https://onrobot.com> web adresinden 21.04.2019 tarihinde alınmıştır.



Şekil 4.3. Ara stok alanları olmayan, robotun iki kısıkaçlı olduğu model

Çizelge 4.1’de ara stok alanları olan esnek robotik hücrelerle ara stok alanı olmayan, çift kısıkaçlı robotun olduğu hücrelerin Nejad ve diğerlerinin [5] makalesinde ve Gültekin ve diğerlerinin [4] makalesinde daha önce en uygun olarak bulunmuş aktivite sıralamaları verilmektedir. İki konfigürasyon için de işleme sürelerinin 30 ve 120 saniye olduğu durumlardaki çizelgeleri görülmektedir. Hücrelerin üretkenliğini karşılaştırabilmek için iki hücrede de benzer ürünler üretilmektedir, robotun yükleme boşaltma süreleriyle makineler arası ulaşım süreleri aynıdır. (Yükleme/boşaltma süresi: $\epsilon = 2$ saniye, makineler arası ulaşım süresi: $\delta = 5$ saniye.) Tabloda parantez içinde yükleme/boşaltma süreleri de verilmiştir.

Çizelge 4.1. Birinci ve ikinci konfigürasyon için robot aktivite sıralamaları

	İşleme Süreleri	Robotun aktivite sıralamaları
Birinci Konfigürasyon (ara stok alanlarının olduğu)	30	L11(19s) – L21(24s) – L22(19s) – L12(14s) – U11(14s) – U21(24s) – U12(14s) – U22(24s)
	120	L11(14s) – U12(29s) – L21(19s) – L12(14s) – U22(24s) – U11(29s) – L22(9s) – U21(24s)
İkinci Konfigürasyon (robotun çift kısıkaçlı olduğu)	30	I1(7s) – U1(2s) – L1(7s) – I2(17s) – O1(7s) – U2(2s) – L2(7s) – O2(17s)

	120	I1(7s) – U1(2s) – L1(7s) – I2(17s) – O1(7s) – U2(2s) – L2(7s) – O2(17s)
--	-----	--

Önce birinci konfigürasyon için petri ağı modeli kurulacaktır. Süreç odaklı petri ağı yöntemine göre işlemler aşağıda belirtilmiştir:

Üretim süreçlerinin sıralanmasının modellenmesi:

Üretim süreçlerini modellerken, parçayı işlemek için ihtiyaç duyulan aktiviteler belirlenir. Ara stok alanları olan hücrelerde parçayı işlemekte gerekli olan aktiviteler, yükleme/boşaltma işlemleri, ara stokta parçaların beklemesi, makinelerin parçaları işlenmesi ve işlenmemiş/işlenmiş parçaların depolanmasını içerir.

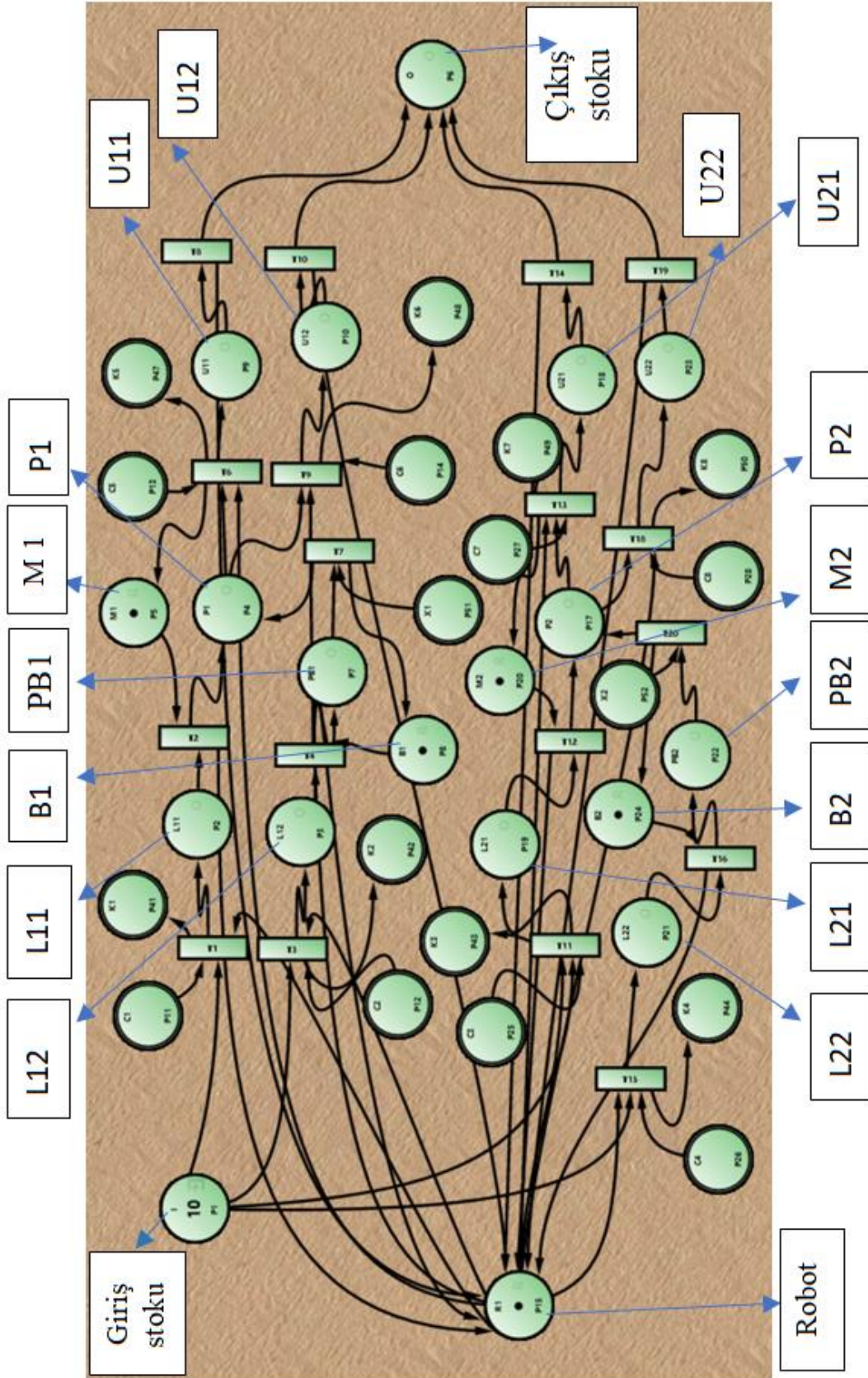
Kaynakların modellenmesi:

Parçayı işlemek için gerekli aktiviteler belirlendikten sonra bu aktivitelerin gerçekleşmesi için ihtiyaç duyulan kaynaklar belirlenir. Burada ihtiyaç duyulan kaynaklar, parçaları işleyecek makineler, bir birim kapasiteli ara stok alanları ve tek kısıkaçlı robottur.

Kaynak ihtiyaçlarının modellenmesi:

Kaynak ihtiyaçlarının modellenmesi de aşağıda Şekil 4.1’de verilen ara stok alanları bulunan esnek robotik hücrenin petri ağı modelinde görülmektedir. Kaynak yerleşimlerinden aktivite yerleşimlerinin giriş geçişlerine ve aktivite yerleşimlerinin çıkış geçişlerinden kaynak yerleşimlerine bir ok uzanır.

Şekil 4.1’de iki makineli, ara stok alanlarına sahip esnek robotik hücrenin süreç odaklı petri ağı modeli verilmiştir. Şekilde yükleme, boşaltma aktiviteleri ve ara stok alanında bekleme, makinelerin parçaları işlenmesi, giriş/çıkış stokları belirtilmiştir. Kontrol yerleşimleri kullanılarak robotun aktivitelerinin modelde istenilen sırada çalıştırılması sağlanmıştır. Giriş stokundan çıkan işlenmemiş parçalar makinelere ya da makinelerin ara stok alanlarına gelmekte, daha sonra makinelerde işlenip çıkış stokuna gitmektedirler. P1 ve P2 aktivite yerleşimlerinin süreleri önce 30, daha sonra 120 saniye yapılarak model çalıştırılıp sonuçlar elde edilmiştir.



Şekil 4.4. Ara stok alanları bulunan esnek robotik hücrenin süreç odaklı petri ağı modeli

Çizelge 4.2’de Şekil 4.4’deki petri ağı modelinin yerleşim ve geçiş düğümlerinin tanımları verilmiştir.

Çizelge 4.2. Şekil 4.4’deki petri ağı modelinin yerleşim ve geçiş düğümlerinin tanımları

Yerleşimler		Geçişler	
I	Giriş stokunda işlenmemiş parçaların beklemesi	T1	Robotun L11 yükleme işlemine başlaması
L11	Robotun M1 makinesini yüklemesi	T2	M1 makinesinin L11’den sonra parçaları işlemeye başlaması
		T3	Robotun L12 yükleme işlemine başlaması
L12	Robotun B1 ara stokunu yüklemesi	T4	Parçaların B1 ara stok alanına geçmesi
		T7	Parçaların B1’den M1’e geçmesi
P1	M1 makinesinin parçaları işlemesi	T11	Robotun L21 yükleme işlemine başlaması
L21	Robotun M2 makinesini yüklemesi	T12	M2’nin L21’den sonra parçaları işlemeye başlaması
P2	M2 makinesinin parçaları işlemesi	T15	Robotun L22 yükleme işlemine başlaması
L22	Robotun B2 ara stokunu yüklemesi	T16	Parçaların B2 ara stok alanına geçmesi
		T20	Parçaların B2’den M2’e geçmesi
PB1	B1 ara stokunda parçanın beklemesi		
PB2	B2 ara stokunda parçanın beklemesi	T6	Robotun U11 boşaltma işlemine başlaması
U11	Robotun M1 makinesinden birinci parçayı boşaltması	T13	Robotun U21 boşaltma işlemine başlaması

Yerleşimler		Geçişler	
U21	Robotun M2 makinesinden birinci parçayı boşaltması	T18	Robotun U22 boşaltma işlemine başlaması
U22	Robotun M2 makinesinden ikinci parçayı boşaltması	T9	Robotun U12 boşaltma işlemine başlaması
U12	Robotun M1 makinesinden ikinci parçayı boşaltması	T14	U21 boşaltmasıyla parçanın çıkış stokuna bırakılması
		T19	U22 boşaltmasıyla parçanın çıkış stokuna bırakılması
		T8	U11 boşaltmasıyla parçanın çıkış stokuna bırakılması
		T10	U12 boşaltmasıyla parçanın çıkış stokuna bırakılması
O	Çıkış stokunda işlenmiş parçaların beklemesi		
M1	M1 makinesinin hazır, yeni bir işe başlayabilir olması		
M2	M2 makinesinin hazır, yeni bir işe başlayabilir olması		
B1	Birinci ara stokun hazır, yeni parça alabilir olması		
B2	İkinci ara stokun hazır, yeni parça alabilir olması		
R1	R1 robotunun hazır, yeni bir işe başlayabilir olması		
C1	L11 yüklemesinin kontrolü, başlama şartı		
K1	L11 yüklemesinin gerçekleşme sayısı		

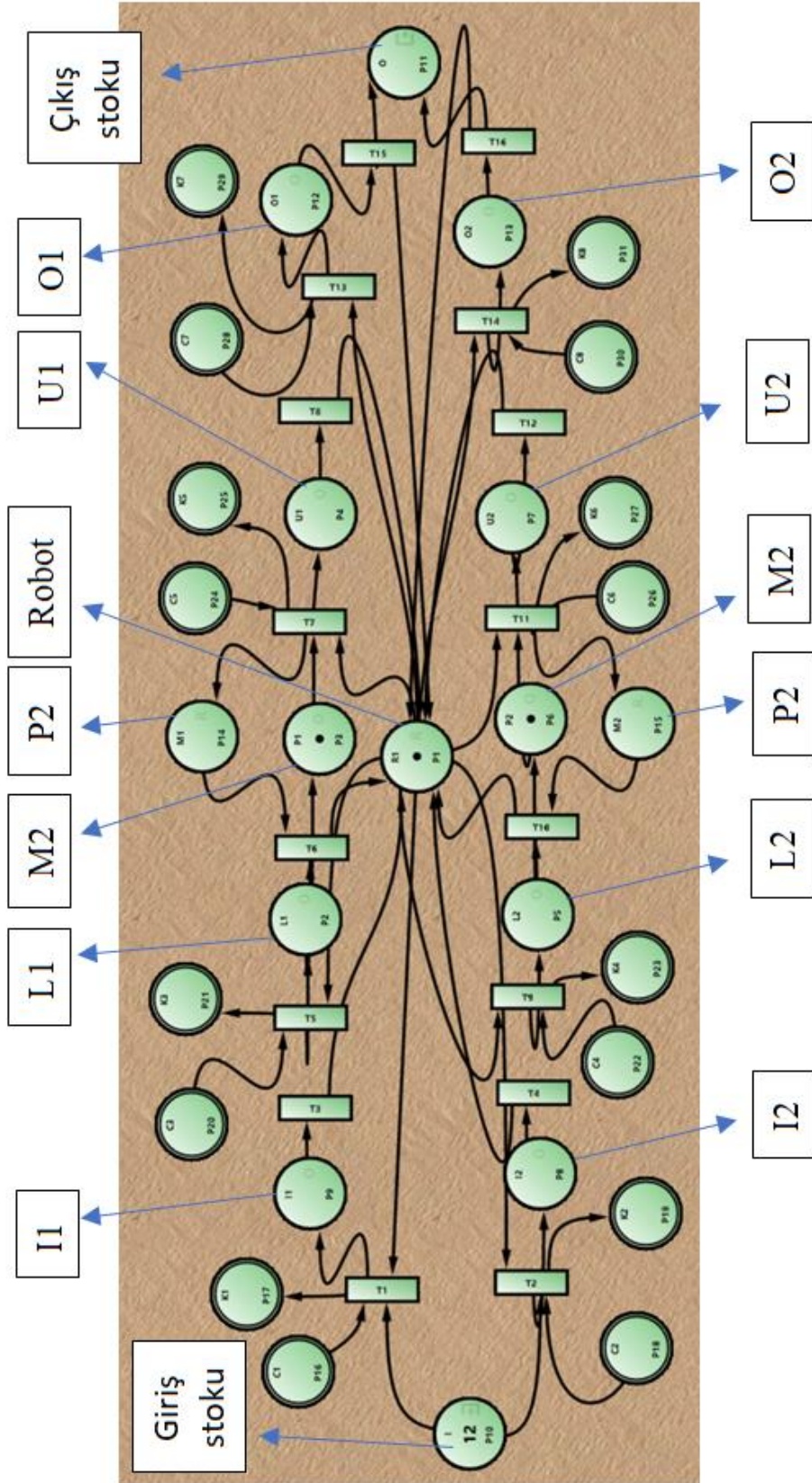
Yerleşimler		Geçişler	
C2	L12 yüklemesinin kontrolü, başlama şartı		
K2	L12 yüklemesinin gerçekleşme sayısı		
C3	L21 yüklemesinin kontrolü, başlama şartı		
K3	L21 yüklemesinin gerçekleşme sayısı		
C4	L22 yüklemesinin kontrolü, başlama şartı		
K4	L22 yüklemesinin gerçekleşme sayısı		
C5	U11 boşaltmasının kontrolü, başlama şartı		
K5	U11 boşaltmasının gerçekleşme sayısı		
C6	U12 boşaltmasının kontrolü, başlama şartı		
K6	U12 boşaltmasının gerçekleşme sayısı		
C7	U21 boşaltmasının kontrolü, başlama şartı		
K7	U21 boşaltmasının gerçekleşme sayısı		
C8	U22 boşaltmasının kontrolü, başlama şartı		
K8	U22 boşaltmasının gerçekleşme sayısı		

Ara stok alanları ve makineleri gösteren kaynak yerleşimlerinin içindeki jeton, kaynağın müsait olduğunu gösterir. Örneğin birinci makineyi gösteren kaynaktaki jeton görünmüyorsa, bu makinenin çalıştığını, yeni bir malzeme daha yüklenemeyeceğini gösterir. Aynı şekilde robotu ifade eden kaynak yerleşiminin üzerinde jeton bulunuyorsa, robot müsait durumda, yükleme ya da boşaltma işlemi yapabilir; ara stok alanını ifade eden yerleşimler üzerinde jeton bulunuyorsa ara stok alanı boş, yükleme gerçekleştirilebilir demektir. Jetonlar ayrıca, stoklarda kaç parça olduğunu da gösterir.

Ara stok alanları olan petri ağı modelinde robot, her döngüde ara stok alanını ya da direkt makineyi olmak üzere her makine için iki kere yükleme işlemi yapar; makinelerden de ikişer kere boşaltma işlemi yapar. Her bir makine için iki yükleme ve iki boşaltma yerleşimi bulunur. Yükleme yerleşimlerinden biri ara stok alanında bekleme işlemi gösteren yerleşime bağlıyken diğeri direkt olarak makinede yapılan işlemi gösteren yerleşime bağlıdır. Boşaltma yerleşimleri de makinede yapılan işlemi gösteren yerleşimden sonra gelir, buna bağlanmışlardır. Süreçte gerçekleşen tüm işlemler ayrıntılı olarak petri ağı modelinde görülebilmektedir, bu da izlenebilirliği artırmaktadır.

Robot aktivite sıralamaları Petri net programı üzerinde kontrol yerleşimleri üzerinden yapılmıştır. Modeldeki Ki ve Ci kontrol yerleşimlerindeki jetonun yer değiştirmesine göre aktiviteler gerçekleşir. “If/then” kalıbıyla adım adım kurallar yazılır. Ki yerleşimindeki jeton sayılarına dayanarak bir sonraki aktivitenin ne olması isteniyorsa, kuralda o aktivitenin giriş geçişine bağlı olan Ci kontrol yerleşiminin 1 olması sağlanır. Örneğin, L11’den sonra L21 işleminin gerçekleşmesi için L21 yerleşiminin giriş geçişi T11’e bağlı C3 kontrol yerleşimine 1 jeton eklenir.

Daha sonra ikinci konfigürasyon için petri ağı modeli kurulmuştur. Süreç odaklı petri ağı yöntemine göre iki makineli, robotun iki kısıkaçlı olduğu model aşağıda Şekil 4.1.2’de gösterilmiştir. Petri ağı modeli oluşturulurken, üretim süreçlerinin modellenmesi aşamasında parçayı işlemek için gerekli aktiviteler yine belirlenir. Burada aktiviteler, I1, I2, L1, L2, U1, U2, O1, O2 yükleme/boşaltmaları ve P1, P2 makine işlemleri ile I/O giriş/çıkış stoklamalarını içerir. Kaynaklar, P1, P2 işlemleri için gerekli olan M1 ve M2 makineleri ile yükleme boşaltma işlemleri için gerekli olan çift kısıkaçlı robottur. Kaynaklar modellenirken de aktivitelerin çıkış geçişinden kaynaklara ve kaynaklardan aktivitelerin giriş geçişine ok uzanır.



Şekil 4.5. Çift kısıkaçlı robotun olduğu esnek robotik hücrenin süreç odaklı petri ağı modeli

Aktivite sıralaması I1 – U1 – L1 – I2 – O1 – U2 – L2 – O2, kontrol yerleşimleriyle kurallar yazılarak gerçekleştirilmiştir. Sıralamada L1 ve L2’den önce U1 ve U2’in gerçekleşmesi gerektiğinden U1 ve U2’nin çalışması için giriş geçişlerine bağlı tüm yerleşimlerde en az bir jeton olmalıdır. Bu nedenle başlangıç işaretlemesinde P1 ve P2’ye de birer jeton eklenmiştir. Çizelge 4.3’te Şekil 4.5’deki yerleşim ve düğümlerin tanımları verilmiştir.

Çizelge 4.3. Şekil 4.5’deki yerleşim ve geçiş düğümlerinin tanımları

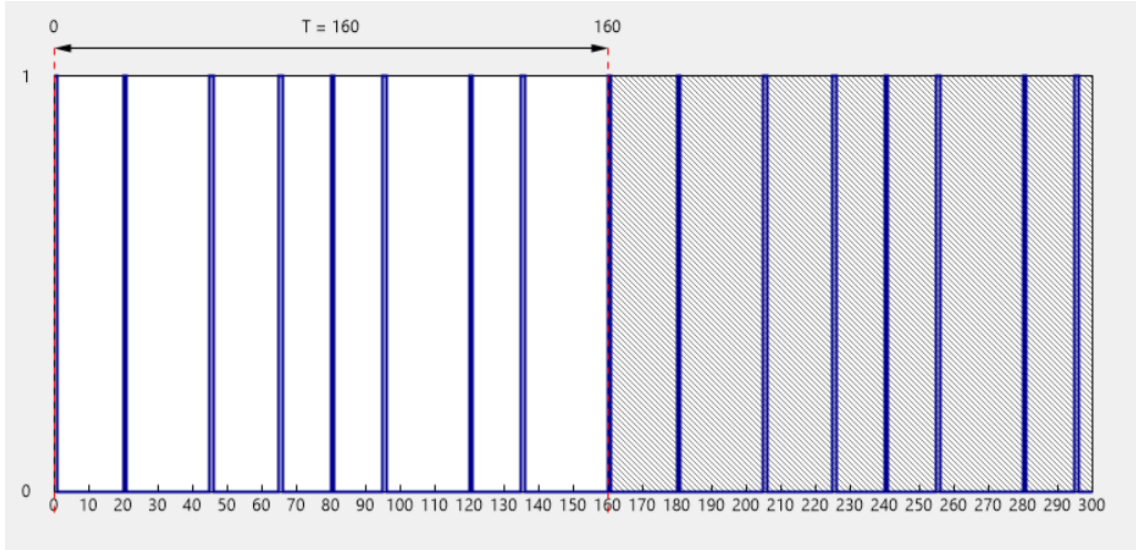
Yerleşimler		Geçişler	
I	Giriş stokunda işlenmemiş parçaların beklemesi	T1	Robotun I giriş stokundan birinci malzemeyi boşaltma işlemine başlaması
I1	Robotun I giriş stokundan birinci işlenmemiş malzemeyi boşaltması	T3	Robotun birinci kısılacında işlenmemiş birinci parça var, ikinci kısılacı boş
		T7	Robotun U1 boşaltma işlemine başlaması
U1	Robotun M1 makinesini boşaltması	T8	Robotun ikinci kısılacında işlenmiş birinci parça var, birinci kısılacında işlenmemiş birinci parça var
		T5	Robotun L1 yükleme işlemine başlaması
L1	Robotun M1 makinesini yüklemesi	T6	M1 makinesinin parçaları işlemeye başlaması, birinci kısılacın boşalması, işlenmiş parça robotun ikinci kısılacında
P1	M1 makinesinin birinci parçayı işlemesi		
		T2	Robotun I giriş stokundan ikinci işlenmemiş malzemeyi boşaltma işlemine başlaması
I2	Robotun I giriş stokundan ikinci işlenmemiş malzemeyi boşaltması	T4	Robotun ikinci kısılacında işlenmiş birinci parça var, birinci kısılacında işlenmemiş ikinci parça var
		T13	Robotun O1 işlemine başlaması

Yerleşimler		Geçişler	
		T14	Robotun O2 işlemine başlaması
O1	Robotun işlenmiş birinci parçayı çıkış stokuna bırakması	T15	Robotun ikinci kıskacı boş, birinci kıskacında işlenmemiş ikinci parça var
		T11	Robotun U2 boşaltmasına başlaması
U2	Robotun M2 makinesini boşaltması	T12	Robotun ikinci kıskacında işlenmiş ikinci parça, birinci kıskacında işlenmemiş ikinci parça var
		T9	Robotun L2 yükleme işlemine başlaması
L2	Robotun M2 makinesini yüklemesi	T10	M2 makinesinin parçaları işlemeye başlaması, birinci kıskacın boşalması, ikinci kıskacında işlenmiş ikinci parça var
P2	M2 makinesinin parçaları işlemesi		
O2	Robotun işlenmiş ikini parçayı çıkış stokuna bırakması	T16	İki kıskacın da boşalması
R1	R1 robotunun hazır, yeni bir işe başlayabilir olması		
M2	M2 makinesinin hazır, yeni bir işe başlayabilir olması		
C1	I1 boşaltmasının kontrolü, başlama şartı		
K1	I1 boşaltmasının gerçekleşme sayısı		
C2	I2 boşaltmasının kontrolü, başlama şartı		
K2	I2 boşaltmasının gerçekleşme sayısı		
C3	L1 yüklemesinin kontrolü, başlama şartı		
K3	L1 yüklemesinin gerçekleşme sayısı		

Yerleşimler		Geçişler	
C4	L2 yüklemesinin kontrolü, başlama şartı		
K4	L2 yüklemesinin gerçekleşme sayısı		
C5	U1 boşaltmasının kontrolü, başlama şartı		
K5	U1 boşaltmasının gerçekleşme sayısı		
C6	U2 boşaltmasının kontrolü, başlama şartı		
K6	U2 boşaltmasının gerçekleşme sayısı		
C7	O1 yüklemesinin kontrolü, başlama şartı		
K7	O1 yüklemesinin gerçekleşme sayısı		
C8	O2 yüklemesinin kontrolü, başlama şartı		
K8	O2 yüklemesinin gerçekleşme sayısı		

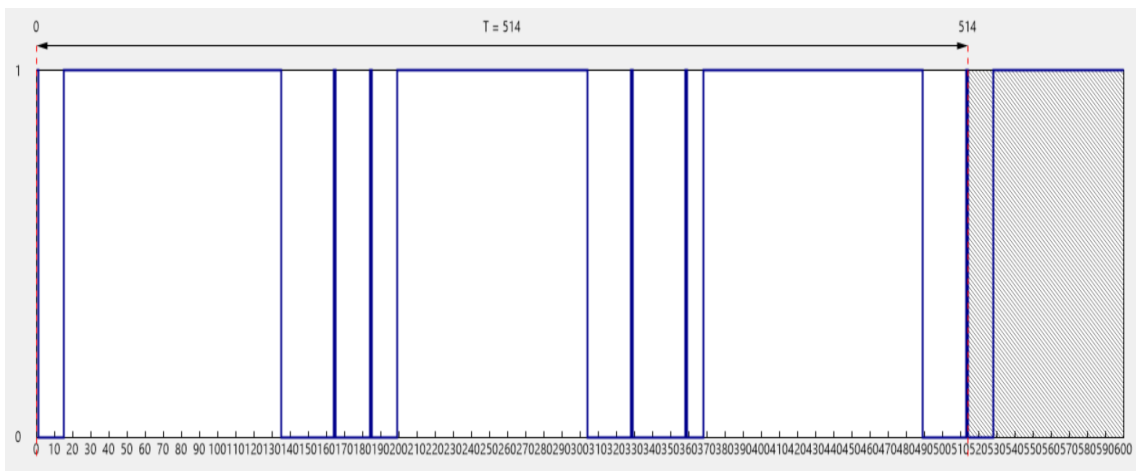
İlk konfigürasyonda, makinenin işleme süresinin 30 saniye olduğu durumda, petri ağlarının çalıştırılmasıyla elde edilen sonuçta, modelin beklediği gibi çalıştığı ve döngüsel olduğu görülmüştür. Teorik yaklaşımla elde edilen çevrim süresi, petri ağlarının çalıştırılmasıyla elde edilen çevrim süresiyle karşılaştırılmıştır. Aşağıda Şekil 4.6'da, ilk konfigürasyonun 30 saniye işleme süresiyle çalıştırıldığı durumun grafığında de görüleceği gibi, çevrim süresi 160 saniye olarak elde edilmiştir. Hücrenin teorik çevrim süresi 152 saniyedir ve 8 saniyelik farklılık bulunmaktadır. Bu farklılığın nedeni, teorik olarak robot için aktiviteler arası değişim süresinin sıfır kabul edilmesidir. Fakat petri ağı modeli gerçeğe daha yakın olduğundan, modelde bu zamanı ihmal etmek mümkün değildir, hücrenin tüm bileşenleri ve robotun aktiviteleri arasında değişim için en az 1 saniye petri ağı modeline yansıtılır. Her döngüde robot dört yükleme ve dört boşaltma işlemi olmak üzere toplam sekiz aktivite gerçekleştirdiğinden, aktiviteler arası değişim süresi 8 saniye olur, bu fark da 152 saniye olan teorik değere eklenerek petri ağı

modelinden elde edilen 160 saniyelik çevrim süresi doğrulanmış olur. Ayrıca hücrenin petri ağı modeli canlıdır. Bu modelin canlılığı; 30 saniyelik makine işleme süresi için, robotun aktivitelerinin sıralaması uygulanabilir, anlamına gelir.



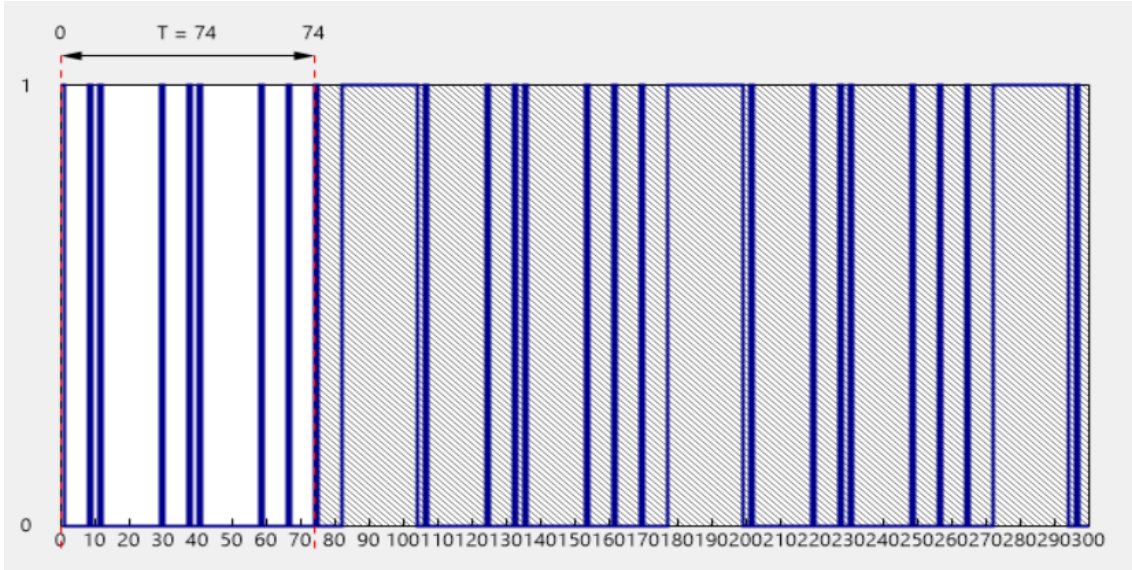
Şekil 4.6. 30 saniye işleme süresiyle çalıştırıldığında ilk konfigürasyon

Şekil 4.7, makinelerin işleme süresi 120 saniye olduğunda çalışma döngüsünü göstermektedir. Grafiğe bakıldığında çevrim süresinin 514 saniye olduğu, teorik olarak elde edilmiş olan 244 saniyelik çevrim süresinin gerçeğe uygun olmadığı görülmektedir. 120 saniye makine işleme süresiyle L11– U12– L21– L12– U22– U11 – L22– U21 robotun aktivite sıralaması pratikte kullanıldığında 244 saniyelik çevrim süresinin gerçekleşmeyeceği, bu sıralamanın en iyi olamayacağı anlaşılır.



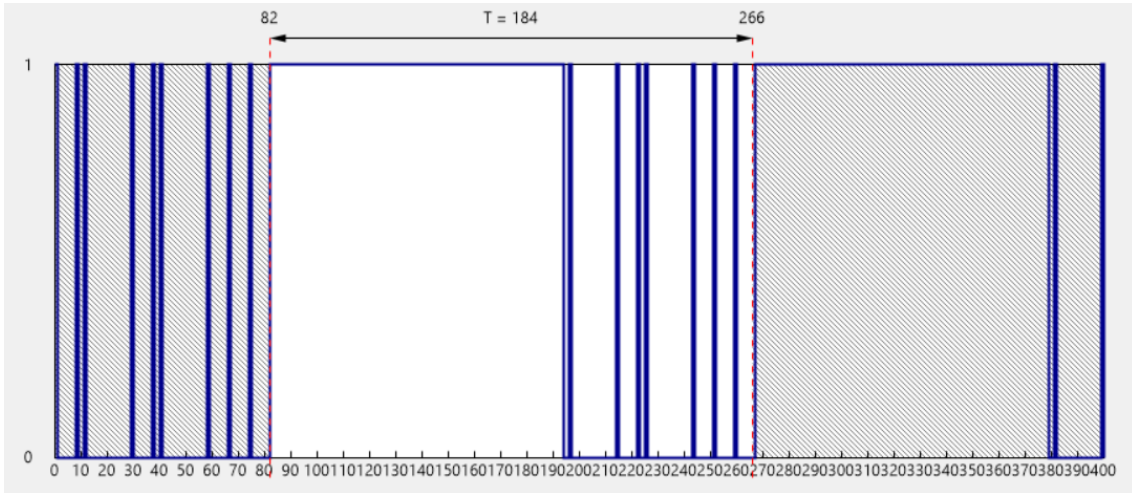
Şekil 4.7. 120 saniye işleme süresiyle çalıştırıldığında ilk konfigürasyon

Şekil 4.8, çift kısaçallı robotun olduđu hücrede, makinelerin işleme sürelerinin 30 saniye olduđu durumdaki çevrim süresini göstermektedir. Modelin canlı olduđu ve çevrim süresinin pratikte 74 saniye olduđu görölmektedir.



Şekil 4.8. 30 saniye işleme süresiyle çalıştırıldığında ikinci konfigürasyon

Şekil 4.9, çift kısaçallı robotun olduđu hücrede, makinelerin işleme süresinin 120 saniye olduđu durum için çevrim süresini göstermektedir. Model canlıdır. İşleme süresinin 120 saniye olduđu durumda, çevrim süresi pratikte 184 saniyedir.



Şekil 4.9. 120 saniye işleme süresiyle çalıştırıldığında ikinci konfigürasyon

İkinci konfigürasyonun pratikteki çevrim süreleri ilk konfigürasyonun çevrim süreleriyle karşılaştırıldığında, ikinci konfigürasyonun daha üretken olduđu görülür ancak bu sonuç oldukça yanıltıcıdır. Çünkü ilk konfigürasyonda her döngüde dört ürün üretilirken, ikinci konfigürasyonda her döngüde iki parça üretilir. Bundan dolayı ilk konfigürasyonun bir

çevrim süresini ikinci konfigürasyonun iki çevrim süresiyle karşılaştırmak daha doğru olacaktır. 30 saniye işleme süresi için iki seçeneğin çevrim süreleri incelendiğinde, birinci konfigürasyonun çevrim süresi ikiyle çarpılacağından 148 saniye, ikinci konfigürasyonun çevrim süresi 160 saniye olduğundan çift kısıkaçlı robotun olduğu hücrenin daha üretken olduğu sonucuna varılabilir. İki konfigürasyon için de hücrelerin çevrim sürelerinin makinelerin işleme sürelerine bağlı olduğu rahatlıkla söylenebilir; ancak çevrim süreleriyle işleme süreleri arasında bir ilişki belirlemek mümkün değildir. Bazı işleme süreleri için robot aktivitelerinin literatürde teorik olarak elde edilmiş sıralamaları gerçek hayatta uygulanamayabilir, çevrim süreleri uyuşmayabilir. Bu nedenle, yatırım yapılmadan önce, robot aktivitelerinin sıralaması, işleme süreleri gibi değişkenler petri ağlarıyla kontrol edilmelidir.

5 SONUÇ

Teknolojinin üretimdeki baskınlığı giderek artmaktadır. Makinelerin yeteneklerinin artmasıyla esnek robotik hücrelere doğru firmalar geçiş yapmaktadır. Esnek robotik hücreler bir parçanın tüm işlemlerini yapan makinelerin olduğu, parçaların tek bir makinede işlendiği üretim sistemleridir. Hücrede bir robot makineler arasında taşıma işlemi ve makineleri yükleme boşaltma aktivitelerini gerçekleştirir. Hücrede çevrim süresinin kısalması, üretkenliğin artmasını sağlar. Çevrim süresinde robotun aktivitelerinin sıralaması oldukça önemlidir. Bu konuda literatürde yapılmış birçok çalışma mevcuttur, ancak bu çalışmaların çoğunda en iyi sonuca ulaşılamamıştır ve elde edilen uygun sonuçların pratikte uygulanıp uygulanamayacağı araştırılmalıdır.

Bu tez çalışmasında esnek robotik hücrelerin süreç odaklı petri ağlarıyla modellenmesi anlatılmıştır ve bir uygulamayla daha önce bulunmuş uygun aktivite sıralamalarının gerçekte çalışıp çalışmadığı incelenmiştir. Uygulamada literatürde geçen çalışmalardan elde edilen robotun aktivite sıralamalarının, makinelerin işleme sürelerinin 30 ve 120 saniye olduğu durumlarda canlı olup olmadığı, çevrim sürelerinin ne kadar olduğu araştırılmıştır. Önce ara stok alanları olan, makinenin işleme süresinin 30 saniye olduğu bir esnek robotik hücre tipi için süreç odaklı petri ağı oluşturulmuş ve modelin canlı olduğu, çevrim süresinin teoride elde edilen süreyle uyumlu olduğu görülerek yöntemin doğru çalıştığı ispatlanmıştır. Daha sonra aynı hücre tipi için makine süresinin 120 saniye olduğu durum incelenmiş, ancak bu durumda teoride verilen aktivite sıralaması ile döngü süresinin gerçekte elde edilemeyeceği anlaşılmıştır. Sonra da çift kısıkaçlı robotun olduğu bir esnek robotik hücre için makinenin işleme sürelerinin 30 ve 120 saniye olduğu durumlarda çevrim süreleri incelenmiştir. Makinenin işleme süresinin 30 saniye olduğu durum için çift kısıkaçlı robotun olduğu hücrenin ara stoklama alanları olan hücreye göre daha üretken olduğu bulunmuştur.

Sonuç olarak, çift kısıkaçlı robotun olduğu hücrelerin ara stoklama alanlarına sahip olan hücrelere göre daha üretken olduğu, süreç odaklı petri ağlarının esnek robotik hücrelerin modellenmesinde ve üretkenliğinin incelenmesinde kullanılabileceği, yatırım yapılmadan önce teorik olarak elde edilmiş sonuçların petri ağlarıyla kontrol edilmesinin faydalı olacağı görülmüştür. Gelecekte bu konunun devamı olarak daha fazla sayıda makinenin olduğu hücrelerin petri ağı modellerinin oluşturulmasıyla, ara stoklama alanlarının sayısının, kapasitesinin ve yerinin ya da robotun kısıkaç sayısının değiştirilmesiyle elde

edilen farklı esnek robotik hücrelerin süreç odaklı petri ağlarıyla modellenmesi ve üretkenliklerinin karşılaştırılması yapılabilir.

6 KAYNAKLAR

- [1] W. Mahmudy, Optimisation of Integrated Multi-Period Production Planning and Scheduling Problems in Flexible Manufacturing Systems (FMS) Using Hybrid Genetic Algorithms, **2014**.
- [2] M. Ghadiri Nejad, G. Kovács, B. Vizvári, R.V. Barenji, *Int. J. Adv. Manuf. Technol.* 95 (**2018**) 3863–3873.
- [3] F. Tüysüz, C. Kahraman, *Expert Syst. Appl.* 37 (**2010**) 3910–3920.
- [4] H. Gultekin, O. Dalgic, S. Akturk, *Robot. Comput. Integr. Manuf.* 48 (**2017**) 121–131.
- [5] M. Ghadiri Nejad, S.M. Shavarani, H. Güden, R.V. Barenji, *Int. J. Adv. Manuf. Technol.* (**2019**).
- [6] A. Giua, M. Silva, *IFAC-PapersOnLine* 50 (**2017**) 1772–1783.
- [7] M. Ghadiri Nejad, H. Güden, B. Vizvári, R. Vatankhah Barenji, *Adv. Mech. Eng.* 10 (**2018**) 168781401775391.
- [8] N. Keddis, G. Kainz, A. Zoitl, A. Knoll, in: 2015 IEEE Int. Conf. Ind. Technol., **2015**, pp. 1901–1906.
- [9] F. Long, P. Zeiler, B. Bertsche, *IFAC-PapersOnLine* 50 (**2017**) 5680–5687.
- [10] S. Yildiz, O.E. Karasan, M.S. Akturk, *Comput. Oper. Res.* 39 (**2012**) 1290–1299.
- [11] H. Gultekin, A. Tula, S. Akturk, *Eur. J Ind. Eng.* 10 (**2016**) 499–526.
- [12] M. Foumani, K. Smith-Miles, I. Gunawan, *Rob. Auton. Syst.* 91 (**2017**) 210–225.
- [13] T.M. Anandan, https://www.robotics.org/content-detail.cfm/Industrial-Robotics-Industry-Insights/Robots-for-Rent-Why-RaaS-Works/Content_id/7665 (n.d.) (Erşim Tarihi 07.04.2019).
- [14] M. Ghadiri Nejad, S.M. Shavarani, B. Vizvári, R.V. Barenji, *Int. J. Adv. Manuf. Technol.* 96 (**2018**) 1081–1091.
- [15] R.V. Barenji, A.V. Barenji, M. Hashemipour, *Int. J. Adv. Manuf. Technol.* 71 (**2014**) 1773–1791.
- [16] S. Mosallaeipour, M.G. Nejad, S.M. Shavarani, R. Nazerian, *Prod. Eng.* 12 (**2018**) 83–94.

- [17] Y. Crama, J. Van de Klundert, *J. Sched.* 2 (1999) 35–54.
- [18] H. Gultekin, M.S. Akturk, O.E. Karasan, *Int. J. Prod. Res.* 46 (2008) 2105–2121.
- [19] H. Gultekin, O. Ekin Karasan, M.S. Akturk, *Comput. Oper. Res.* 36 (2009) 329–343.
- [20] K.S. Jung, H.N. Geismar, M. Pinedo, C. Sriskandarajah, *Prod. Oper. Manag.* 27 (2018) 285–303.
- [21] C. Sriskandarajah, B. Shetty, *Int. J. Prod. Res.* 56 (2018) 817–847.
- [22] K.S. Jung, H.N. Geismar, M. Pinedo, C. Sriskandarajah, *IIE Trans.* 47 (2015) 634–652.
- [23] M. Foumani, M.Y. Ibrahim, I. Gunawan, in: 2013 IEEE Int. Symp. Ind. Electron., IEEE, 2013, pp. 1–6.
- [24] I.G. Drobouchevitch, H. Neil Geismar, C. Sriskandarajah, *Eur. J. Oper. Res.* 206 (2010) 623–633.
- [25] J.L. Peterson, *ACM Comput. Surv.* 9 (1977) 223–252.
- [26] J. Wang, *Handb. Dyn. Syst. Model.* (2007).
- [27] T. Murata, *Proc. IEEE* 77 (1989) 541–580.
- [28] A. Bobbio, *Syst. Reliab. Assess.* (1990) 102–143.
- [29] M. Uzam, A. H. JONES, *Turkish J. Electr. Eng. Comput. Sci.* 10 (2002) 85–109.
- [30] I. Radojevic, Z. Salcic, *Embedded Systems Design Based on Formal Models of Computation*, Springer Netherlands, Dordrecht, 2011.
- [31] Shashank Satish More, Dr. S. G. Bhatwadekar, *Int. J. Eng. Res.* V5 (2016).
- [32] R. Zurawski, MengChu Zhou, *IEEE Trans. Ind. Electron.* 41 (1994) 567–583.
- [33] E.G. Coffman, M. Elphick, A. Shoshani, *ACM Comput. Surv.* 3 (1971) 67–78.
- [34] Z. Li, G. Liu, H.-M. Hanisch, M. Zhou, *IEEE Trans. Syst. Man, Cybern. - Part A Syst. Humans* 42 (2012) 178–191.
- [35] Z.W. Li, H.S. Hu, A.R. Wang, *IEEE Trans. Syst. Man Cybern. Part C (Applications Rev.)* 37 (2007) 517–526.
- [36] Naiqi Wu, MengChu Zhou, ZhiWu Li, *IEEE Trans. Syst. Man, Cybern. - Part A*

- Syst. Humans 38 (2008) 56–69.
- [37] Q. Zhu, N. Wu, Y. Qiao, M. Zhou, IEEE Access 4 (2016) 2096–2109.
- [38] S.A. Shah, E.L.J. Bohez, K. Shah, I. ul Haq, K. Azam, S. Anwar, Int. J. Adv. Manuf. Technol. 88 (2017) 1775–1787.
- [39] A.J. Cunha de Aguiar, E. Villani, F. Junqueira, Robot. Comput. Integr. Manuf. 27 (2011) 929–941.
- [40] Ö. Başak, Y.E. Albayrak, Comput. Ind. Eng. 86 (2015) 116–126.
- [41] X.D. Nie, X.D. Chen, X. Chen, Int. J. Simul. Model. 15 (2016) 566–576.
- [42] W. Zhang, T. Freiheit, H. Yang, Robot. Comput. Integr. Manuf. 21 (2005) 550–558.
- [43] C. Jung, H.-J. Kim, T.-E. Lee, IEEE Trans. Autom. Sci. Eng. 12 (2015) 309–323.
- [44] M. Fathian, I. Nakhai Kamalabadi, M. Heydari, H. Farughi, J. Softw. Eng. Appl. 04 (2011) 603–608.
- [45] A. Al-Ahmari, Inf. Sci. (Ny). 329 (2016) 638–648.
- [46] T.P. Truong, P.P. Dang, B.L. Duc, in: 2018 4th Int. Conf. Green Technol. Sustain. Dev., IEEE, 2018, pp. 763–770.
- [47] R. Vatankhah Barenji, B. Yuksel Ozkaya, A. Vatankhah Barenji, Int. J. Adv. Manuf. Technol. 93 (2017) 3677–3691.
- [48] A.H. Taha, M.H.F. Al Hazza, E.Y.T. Adesta, Int. J. Eng. Mater. Manuf. 2 (2017) 11.
- [49] S. Zhang, N. Wu, Z. Li, T. Qu, C. Li, Inf. Sci. (Ny). 417 (2017) 247–261.
- [50] N. Wu, M. Zhou, Asian J. Control 12 (2010) 267–280.
- [51] R. Davidrajuh, B. Skolud, D. Krenczyk, Computers 7 (2018) 8.

EKLER

EK 1 - Tezden Türetilmiş Bildiriler

A. Şahin, G. Güteryüz, R.V. Barenji, Ara stok alanı olan esnek robotik hücrelerde petri ağlarının kullanılması, 39. Yöneylem Araştırması ve Endüstri Mühendisliği Ulusal Kongresi-YAEM 2019, 12-14 Haziran, Ankara, **2019**.



HACETTEPE ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
YÜKSEK LİSANS/~~DOKTORA~~ TEZ ÇALIŞMASI ORJİNALLİK RAPORU

HACETTEPE ÜNİVERSİTESİ
FEN BİLİMLER ENSTİTÜSÜ
Endüstri Mühendisliği ANABİLİM DALI BAŞKANLIĞI'NA

Tarih:24/06/2019

Tez Başlığı / Konusu: Esnek Robotik Hücrelerin Süreç Odaklı Petri Ağlarıyla Modellenmesi

Yukarıda başlığı/konusu gösterilen tez çalışmamın a) Kapak sayfası, b) Giriş, c) Ana bölümler d) Sonuç kısımlarından oluşan toplam 62 sayfalık kısmına ilişkin, 23/06/2019 tarihinde şahsım/tez danışmanım tarafından Turnitin adlı intihal tespit programından aşağıda belirtilen filtrelemeler uygulanarak alınmış olan orijinallik raporuna göre, tezimin benzerlik oranı % 2 'dir.

Uygulanan filtrelemeler:

- 1- Kaynakça hariç
- 2- Alıntılar ~~hariç~~/dâhil
- 3- 5 kelimedenden daha az örtüşme içeren metin kısımları hariç

Hacettepe Üniversitesi Fen Bilimleri Enstitüsü Tez Çalışması Orijinallik Raporu Alınması ve Kullanılması Uygulama Esasları'nı inceledim ve bu Uygulama Esasları'nda belirtilen azami benzerlik oranlarına göre tez çalışmamın herhangi bir intihal içermediğini; aksinin tespit edileceği muhtemel durumda doğabilecek her türlü hukuki sorumluluğu kabul ettiğimi ve yukarıda vermiş olduğum bilgilerin doğru olduğunu beyan ederim.

Gereğini saygılarımla arz ederim.

Adı Soyadı: Ayşegül Şahin
Öğrenci No: N16120631
Anabilim Dalı: Endüstri Mühendisliği
Programı: Endüstri Mühendisliği
Statüsü: Y.Lisans Doktora Bütünleşik Dr.

Tarih ve İmza

24-06-2019

Ayşegül Şahin

DANIŞMAN ONAYI

UYGUNDUR.

Dr. Öğr. Üyesi Güldal Gökyüzü
(Unvan, Ad Soyad, İmza)

Güldal



HACETTEPE UNIVERSITY
GRADUATE SCHOOL OF SCIENCE AND ENGINEERING
THESIS/~~DISSERTATION~~ ORIGINALITY REPORT

HACETTEPE UNIVERSITY
GRADUATE SCHOOL OF SCIENCE AND ENGINEERING
TO THE DEPARTMENT OF Industrial Engineering

Date: 24/06/2019

Thesis Title / Topic: Modeling of Flexible Robotic Cells Using Process Oriented Petri Nets

According to the originality report obtained by myself/my thesis advisor by using the *Turnitin* plagiarism detection software and by applying the filtering options stated below on 23/06/2019 for the total of 62 pages including the a) Title Page, b) Introduction, c) Main Chapters, d) Conclusion sections of my thesis entitled as above, the similarity index of my thesis is 2 %.

Filtering options applied:

1. Bibliography/Works Cited excluded
2. Quotes ~~excluded~~ / included
3. Match size up to 5 words excluded

I declare that I have carefully read **Hacettepe University Graduate School of Science and Engineering Guidelines for Obtaining and Using Thesis Originality Reports**; that according to the maximum similarity index values specified in the Guidelines, my thesis does not include any form of plagiarism; that in any future detection of possible infringement of the regulations I accept all legal responsibility; and that all the information I have provided is correct to the best of my knowledge.

I respectfully submit this for approval.

Name Surname: Ayşegül Şahin

Student No: N16120631

Department: Endüstri Mühendisliği

Program: Endüstri Mühendisliği

Status: Masters Ph.D. Integrated Ph.D.

Date and Signature

24-06-2019

Ayşegül Şahin

ADVISOR APPROVAL

APPROVED.

Dr. Öğr. Üyesi *Gülden Güler*
(Title, Name Surname, Signature)

Gülden

ÖZGEÇMİŞ

Adı Soyadı : Ayşegül ŞAHİN
Doğum yeri : Çankaya
Doğum tarihi : 10.08.1989
Medeni hali : Bekâr
Yazışma adresi : Şehit Cengiz Topel Mah. 141/23 İncesu/ANKARA
Telefon : 90 (534) 637 06 88
Elektronik posta adresi : aysegul.ahin36@gmail.com
Yabancı dili : İngilizce, ileri seviye

EĞİTİM DURUMU

Lisans : Hacettepe Üniversitesi Endüstri Mühendisliği Bölümü
Yüksek Lisans : Hacettepe Üniversitesi Endüstri Mühendisliği Bölümünde devam durumunda

İş Tecrübesi

AFS Boru San. AŞ'de Satın Alma Uzmanı (2014)
Aselsan AŞ'de Üretim Kontrol Mühendisi (2015- halen)

