T.C.
HARRAN ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

YÜKSEK LİSANS TEZİ

ÇİZELGELEME PROBLEMLERİNDE ENDÜSTRİ 4.0 UYGULAMASI

Mehmet Emin AYDOĞDU

ELEKTRİK ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI

ŞANLIURFA
2020

Dr. Öğr. Üyesi Nurettin BEŞLİ danışmanlığında Mehmet Emin AYDOĞDU ’nun hazırladığı **“Çizelgeleme Problemlerinde Endüstri 4.0 Uygulaması”** konulu bu çalışma 12/06/2020 tarihinde aşağıdaki jüri tarafından oy birliği ile Harran Üniversitesi Fen Bilimleri Enstitüsü Elektrik Elektronik Mühendisliği Ana Bilim Dalı’nda YÜKSEK LİSANS TEZİ olarak kabul edilmiştir.

İmza

Danışman : Dr. Öğr. Üyesi Nurettin BEŞLİ ..……

Üye : Dr. Öğr. Üyesi Serkan KAYA ……..

Üye : Dr. Öğr. Üyesi Faruk KÜRKER ……..

Bu tezin Elektrik Elektronik Mühendisliği Ana Bilim Dalında Yapıldığını ve Enstitümüz Kurallarına Göre Düzenlendiğini Onaylarım.

Doç. Dr. İsmail HİLALİ
Enstitü Müdürü

Not: Bu Tezde kullanılan özgün ve başka kaynaktan yapılan bildirişlerin, çizelge, şekil ve fotoğrafların kaynak gösterilmeden kullanımı 5486 sayılı Fikir ve Sanat Eserleri Kanunundaki hükümlere tabidir.

İÇİNDEKİLER

Sayfa No

ÖZET ii
ABSTRACT v
TEŞEKKÜR vi
ŞEKİLLER DİZİNİ iv
ÇİZELGELER DİZİNİ vi
SİMGELER ve KISALTMALAR DİZİNİ vii
1. GİRİŞ 1
2. ÖNCEKİ ÇALIŞMALAR 5
3. MATERYAL ve YÖNTEM 9
3.1. Çizelgeleme Problemleri 14
3.1.1. Statik çizelgeleme 17
3.1.2. Dinamik çizelgeleme 17
3.1.3. Atölye Tipi Çizelgeleme (ATÇ) ve Esnek Atölye Tipi Çizelgeleme (EATÇ) problemleri 18
3.1.4. Parçacık Sürü Optimizasyonu (PSO) 20
3.1.5. Yerel Arama (YA) 26
3.2. Nesnelerin İnterneti 28
3.2.1. Nesnelerin interneti mimarisi 29
3.2.2. Nesnelerin interneti alanında kullanılan teknolojiler 31
3.2.3. Nesnelerin interneti uygulama alanları 41
3.2.4. Geliştirmeye açık alanlar 47
3.2.5. Araştırma konuları 50
3.3. Endüstri 4.0 51
3.3.1. Dünyada endüstri 4.0 52
3.3.2. Türkiye ve endüstri 4.0 53
4. ARAŞTIRMA BULGULARI ve TARTIŞMA 56
4.1. Çizelgeleme İşlemleri ve Sonuçlar 56
4.1.1. WA veri seti için işlem adımları 58
4.1.2. EX4 veri seti için işlem adımları 62
4.1.3. Donanım özellikleri ve çizelgeleme süresi 65
4.2. IoT deney ortamı 66
5. SONUÇLAR ve ÖNERİLER 72
KAYNAKLAR 74
ÖZGEÇMİŞ 78
EKLER 79
EK 1: 15x10x56 Boyutlu DEATÇ (Ex4) problemi makine işlem süreleri 79

ÖZET

Yüksek Lisans Tezi

ÇİZELGELEME PROBLEMLERİNDE ENDÜSTRİ 4.0 UYGULAMASI

Mehmet Emin AYDOĞDU

Harran Üniversitesi
Fen Bilimleri Enstitüsü
Elektrik Elektronik Mühendisliği Ana Bilim Dalı

Danışman: Dr. Öğr. Üyesi Nurettin BEŞLİ

Yıl: 2020, Sayfa: 79

İnternet, dünyanın geleceğine yön vermeye başlamış, hayatımızda olmazsa olmaz araçlardan biri haline gelmiştir. Her geçen yıl içerik, kapasite ve erişilebilirlik bakımından hızla büyümektedir. Nesnelerin İnternetinin (IoT-Internet of Things) yaygınlaşmasıyla süreçlerin her seviyesinde veri toplanması ve bu verilere bağlı hızlı karar ve denetim çıkışları oluşturulması mümkün olmuştur. Endüstri alanında kullanıldığında elde edilecek verilerin analizi ile üretim planlaması optimize edilerek ciddi boyutlarda zaman, kaynak ve enerji tasarrufu sağlanabilmektedir.

Endüstri 4.0 ya da 4. Sanayi Devrimi, nesnelerin interneti gibi ileri teknolojilerin endüstriye entegrasyonu ile gerçekleşmiştir. Endüstri 4.0 ileri haberleşme teknolojilerini ve otomasyon sistemlerini kullanarak üretimin daha verimli gerçekleşmesini hedeflemektedir. Endüstride yaygın olarak ortaya çıkan atölye tipi çizelgeleme (EATÇ) problemleri bilgisayar işlem kapasitelerinin artmasıyla sezgisel yöntemler kullanılarak çözülmekte ve başlangıçta belirlenen sabit parametreler üzerinden gerçekleştirilmektedir. Gerçek hayatta ise makinelerin bozulması veya kaynakların yetişmemesi gibi ek problemler oluşabilmektedir.

Bu çalışmada, EATÇ ve Dinamik EATÇ problemleri için geliştirmiş olan algoritmalar incelenerek, DEATÇ problemlerinde kullanılabilecek Parçacık Sürü Optimizasyonu (PSO) ve Yerel Arama (YA) metotlarının kullanıldığı hibrit bir algoritma önerilmiş ve yapılan benzetimlerde klasik çizelgeleme yöntemleriyle karşılaştırmalar yapılmıştır.

Tezin en temel amacı Çizelgeleme Problemlerinde Endüstri 4.0 uygulamasını gerçekleştirmek ve literatüre katkı sağlamaktır. Bu amaçla üretimde kullanılacak olan makineler veya üretim bantlarının IoT ağına dâhil edilerek anlık durum verilerinin alınması, analiz edilmesi, esnek atölye tipi çizelgelemenin oluşturulması ve denetim bilgilerinin iletilmesini sağlayacak donanım ve yazılım alt yapısı tasarlanmıştır. Önermiş olduğumuz metot, literatürde yer alan bazı veri setleri üzerinde denenmiş ve ilk elde edilen çizelgeme sonucuyla, yeniden yapılan çizelgeleme sonucu arasındaki yüzde sapma değerleri hesaplanmıştır. Sonuçlar incelendiğinde veri setlerinin bir kısmında yüzde sapma %0, diğerlerinde ise maksimum %9.09 olduğu gözlemlenmiştir. Bu değerler önemiş olduğumuz metodun pratikte uygulanabilir olduğunu göstermiştir.

ANAHTAR KELİMELER: Nesnelerin İnterneti, Endüstri 4.0, Üretim Planlama, Çizelgeleme, Akıllı Makineler

ABSTRACT

MSc Thesis

INDUSTRY 4.0 APPLICATION IN SCHEDULING PROBLEMS

Mehmet Emin AYDOĞDU

Harran University
Graduate School of Natural and Applied Sciences
Department of Electrical and Electronics Engineering

Supervisor: Assist. Prof. Dr. Nurettin BEŞLİ
Year: 2020, Pages: 79

The Internet, starting to direct the future of the world, has become one of the essential tools in our lives. It is growing rapidly in terms of content, capacity and accessibility with each passing year. With the spread of the Internet of Things (IoT), it has been possible to collect data at all levels of the processes and to create rapid decision and control outputs based on these data. When IoT devices are used in Industry, it can optimize production planning by analyzing the data obtained from this network and save time, resources and energy in serious dimensions.

The Industry 4.0 or the 4th Industrial Revolution took place with the integration of advanced technologies such as the Internet of Things into the industry. Industry 4.0 aims to make production more efficient by using advanced communication technologies and automation systems. Flexible Job Shop Problems (FJSP), that are common in industry, are solved by using heuristic methods with the increase of computer processing capacities and realized through fixed parameters determined at the beginning. In real life, additional problems may occur, such as failure of machines or failure of resources.

In this study, by analyzing algorithms developed for FJSP and Dynamic FJSP, a hybrid algorithm using Particle Swarm Optimization (PSO) and Local Search (LS) methods is proposed. The comparisons were made with classical scheduling methods.

The main aim of the thesis is to implement Industry 4.0 application in Scheduling Problems and to contribute to the literature. For this purpose, the hardware and software infrastructure which upgrades the machines or production lines to IoT devices and provides analysis of instant status data, creation of flexible workshop type scheduling and transmission of audit information is designed. The method proposed in this study has been tested on some datasets in the literature and the percent deviation values between the first scheduling result and the rescheduling result have been calculated. When the results are analyzed, it is observed that the percentage deviation is 0% for some of the data sets and is up to 9.09% for the others. These values show that the method which we have proposed is practically feasible.

KEY WORDS: Internet of Things, Industry 4.0, Production Planning, Scheduling, Intelligent Machines.

TEŞEKKÜR

Bu tez çalışmasının hazırlanması sürecinde bana yardımcı olan, bilgi ve deneyimlerini titizlikle aktaran değerli danışman hocam Dr. Öğr. Üyesi Nurettin BEŞLİ' ye, bilgi ve tecrübesiyle desteklerini bizden esirgemeyen Dr. Öğr. Üyesi Serkan KAYA' ya, bu süreçte destekleriyle beni yalnız bırakmayan Prof. Dr. Mehmet Tahir GÜLLÜOĞLU ve Dr. Öğr. Üyesi Mehmet Emin TENEKECİ 'ye, değerli Bilgi İşlem Daire Başkanlığındaki arkadaşlarıma, maddi manevi destekleri ile her zaman yanımda olan aileme ve tüm dostlarıma sonsuz teşekkürlerimi sunarım.

ŞEKİLLER DİZİNİ

Sayfa No

Şekil 3.1. C# Uygulamasına Ait Genel Süreci Gösteren Akış Diyagramı 10
Şekil 3.2. C# Sunucu Bilgisayar Üzerinde Gerçekleşen Süreci Gösteren Akış Diyagramı..... 11
Şekil 3.3. C# Yazılım Dili Kullanılarak Geliştirilen Otomasyon Görseli 12
Şekil 3.4. Makineler Üzerinde Gerçekleşen Süreçleri Gösteren Akış Diyagramı 13
Şekil 3.5. Sanayide kullanılan makinaları temsil eden Raspberry Pi mini bilgisayarı ve sistem tasarımı..... 14
Şekil 3.6. Esnek Atölye Tipi (EAT) üretim ortamı 20
Şekil 3.7. Xp Vektörünün operasyon işlem sırasına dönüşümü 22
Şekil 3.8. İlk parçacığa ait Çözümün Gantt şeması 23
Şekil 3.9. YA stratejisinin uygulandığı 1. Parçacığa ait çözümün Güncellenmiş Gantt şeması..... 27
Şekil 3.10. Nesnelerin İnternetinin Teknolojiler Arasındaki Yeri..... 28
Şekil 3.11. Nesnelerin İnterneti Mimarisi 29
Şekil 3.12. İvmeölçerin mekanik modeli..... 32
Şekil 3.13. Tıbbi algılayıcı içeren Geçici Dövme 34
Şekil 3.14. Etiketlerin güç ve iletişim mekanizması (100 Mhz'den daha düşük frekanslarda)..... 37
Şekil 3.15. Zigbee'nin bina otomasyonunda kullanımı 39
Şekil 3.16. Akıllı Evler için Blok Diyagram 42
Şekil 3.17. Akıllı Sağlık Sistemi için Blok Diyagram..... 45
Şekil 3.18. Anket çalışmasının gerçekleştirildiği ülkeler 52
Şekil 3.19. ABD, Japonya ve Almanya'da 2015 yılında gerçekleştirilen çalışmaya göre toplam gelire göre harcama payları ve toplam gelire göre gelir payları (McKinsey) 53
Şekil 3.20. Google' de Türkiye üzerinden Endüstri 4.0 kavramının aranması..... 54
Şekil 4.1. WA veri seti için HA ile elde edilen Gantt şeması..... 62
Şekil 4.2. DO1 Dinamik olayından sonra WA veri seti için HA ve S1 stratejisi kullanılarak elde edilen Gantt şeması..... 62
Şekil 4.3. Ex4 veri seti için HA ile elde edilen Gantt şeması 63
Şekil 4.4. (5, 3, 5) Dinamik olayından sonra Ex4 veri seti için HA ve S1 stratejisi kullanılarak elde edilen Gantt şeması..... 64
Şekil 4.5. Makineleri temsil etmek için kullanılan mini bilgisayar 67
Şekil 4.6. Deney düzeneği için hazırlamış olduğumuz makine tasarımı, Ledler, button ve buzzer 68
Şekil 4.7. Hazırlanan deney düzeneğinin görseli 71

ÇİZELGELER DİZİNİ

Sayfa No

Çizelge 3.1. 3x4x8 Boyutlu EATÇ problemi makine işlem süreleri 21
Çizelge 3.2. RFID Sistemlerde kullanılan farklı etiketler 36
Çizelge 3.3. Zigbee teknolojisinin farklı kablosuz iletişim teknolojileriyle karşılaştırılması 39
Çizelge 4.1. Dinamik olay (Makine arızası) senaryoları 57
Çizelge 4.2. HA ile elde edilen sonuçların HGR ve VIRS ile karşılaştırılması 58
Çizelge 4.3. HA ile elde edilen sonuçların farklı çalışmalar ile karşılaştırılması 58
Çizelge 4.4. 5x6x25 Boyutlu DEATÇ (WA) problemi makine işlem süreleri 59
Çizelge 4.5. Üretim sırasında acil gelen işe (J_6) ait makine işlem süreleri 60
Çizelge 4.6. HA ile gerçekleştirilen yeniden çizelgeleme stratejilerinin karşılaştırılması 61
Çizelge 4.7. WA veri seti için HA ve S1 ile elde edilen sonuçların VIRS ile karşılaştırılması 61
Çizelge 4.8. Arıza İçeren Üretim Süreci 64
Çizelge 4.9. Çizelgeleme işlemleri için kullanılan sunucu bilgisayar özellikleri 65
Çizelge 4.10. Veri setlerinin çizelgeleme işlem süreleri (milisaniye) 66
Çizelge 4.11. Makine üzerindeki Ledlerin Anlamları 69

SİMGELER ve KISALTMALAR DİZİNİ

API	Application Programming Interface
ATÇ	Atölye Tipi Çizelgeleme
A-GPS	Assisted Global Positioning System
C_{max}	İşlerin Maksimum Tamamlanma Süresi
DEATÇ	Dinamik Esnek Atölye Tipi Çizelgeleme
DFJSP	Dynamic Flexible Job Shop Problem
DO1	Dinamik Olay 1
EATÇ	Esnek Atölye Tipi Çizelgeleme
IEEE	The Institute of Electrical and Electronics Engineers
FJSP	Flexible Job Shop Problem
GPS	Global Positioning System
GPRS	General Packet Radio Service
GSM	Global System for Mobile Communications
GUI	Graphical User Interface
HA	Çalışmamızda Önerilen Hibrit Algoritma
IoT	The Social Internet of Things
IP	Internet Protocol Address
IPv6	Internet Protocol Address Version 6
NFC	Near Field Communication
PMÇ	Paralel Makine Çizelgeleme
PSO	Parçacık Sürü Optimizasyonu
QoS	Quality of System
RFID	Radio Frequency Identification
SIoT	The Social Internet of Things
S1	Strateji 1
VIRS	Variable Interval Rescheduling Strategy
WSN	Wireless Sensor Network
YA	Yerel Arama

1. GİRİŞ

İnternet, dünyanın geleceğine yön vermeye başlamış, hayatımızda olmazsa olmaz araçlardan biri haline gelmiştir. Ortalama her 10 yılda bir içerik, kapasite ve erişilebilirlik bakımından hızla büyümektedir. Artık çok çeşitli ev aletlerinin, aklımıza gelebilecek her türlü cihaz ve aracın internet ortamına bağlanacağı bir döneme giriyoruz. "Nesnelerin İnterneti (IoT)" dönemine giriyoruz. Bu terim farklı araştırmacılar tarafından farklı şekillerde tanımlanmıştır. Örneğin, Vermesan ve ark. (2011) nesnelerin internetini "fiziksel ve dijital dünyalar arasındaki etkileşim" olarak tanımlarlar. Dijital dünya bir sürü algılayıcı ve aktüatör kullanarak fiziksel dünya ile etkileşime girer. Bir diğer araştırmacı ise Nesnelerin İnternetini "Bilgisayar ve Ağ yeteneklerinin her türlü akla yatkın nesne üzerine yerleştirildiği bir paradigma" olarak tanımlar (López, 2005). Ortak bir deyişle Nesnelerin İnterneti, kullanılan her türlü nesnenin internete bağlı olduğu ve karmaşık görevleri yerine getirmek için birbirleriyle haberleştikleri bir dünyayı işaret etmektedir.

Nesneler, Fiziksel dünyayla etkileşime geçip bilgileri dijital dünyaya aktarabilmeleri için, gömülü algılayıcılar, aktüatörler ve işlemcilerle donatılmıştır. Nesnelerin İnterneti (IoT) tek başına bir teknoloji değildir. Birlikte çalışan çeşitli teknolojilerin bir araya gelmesiyle ortaya çıkmaktadır (Sethi ve Sarangi, 2017).

Verilerin depolanması ve işlenmesi, IoT ağı içerisinde algılayıcıların bulunduğu yerel ağda veya uzak bir sunucuda gerçekleştirilebilir. Genel olarak veriler uzak bir sunucuya gönderilmeden önce ön işleme tabi tutulur. İşlenen veriler daha sonra uzaktaki bir sunucuya gönderilir. IoT nesneleri genellikle bellek, enerji, güç ve hesaplama yeteneği açısında kısıtlı cihazlar olduklarından verilerin istenen doğrulukta filtrelenmesi ve depolanması problemleri çözüm bekleyen en önemli konular arasındadır. Bir diğer önemli problem ise IoT nesneleri arasındaki iletişim problemleridir. IoT nesneleri arasındaki iletişim, coğrafi olarak dağınık konumlarda konumlandırıldıklarından genellikle kablosuz gerçekleştirilir. Kablosuz iletişim kanalları yüksek oranda bozulma oranına sahip olduklarından veri iletişiminde ciddi

sorunlar yaşanabilmektedir. IoT de çok fazla yeniden iletim olmaksızın hızlı ve güvenilir bir şekilde veri iletimi önem arz etmektedir.

İnternet ağına dâhil edilen verilerin işlenmesi analiz edilmesiyle birlikte geleceğimizi ciddi anlamda etkileyebilecek kararlar alınabilmekte, yine bu analizler ve hesaplamalar ışığında üretim planlama daha verimli bir şekilde gerçekleştirilmekte ve kaynak kullanımı kontrol altına alınabilmektedir. Ayrıca verilerin tek bir merkezde toplanması için bulut bilişim alt yapı oluşturmaktadır. IoT teknolojileri, Sağlık, fitness, eğitim, eğlence, sosyal yaşam, enerji tasarrufu, çevre izleme, ev otomasyonu ve ulaşım sistemleri vb. alanlarda kullanılmaktadır. Endüstri alanında kullanıldığında bu ağdan elde edilecek verilerin analizi ile üretim planlaması optimize edilerek ciddi boyutlarda zaman, kaynak ve enerji tasarrufu sağlanabilmektedir.

Endüstrinin verimliliği, gerçekleştirilen üretimin pazarda ne kadar başarılı olduğuna ve pazar koşullarına ne kadar uyum sağlayabildiğine bağlıdır. Bu nedenle günümüz pazar koşullarında değişken müşteri talepleri, rekabet dolayısıyla maliyetin düşürme gerekliliği, tüketici taleplerini hızlı ve zamanında karşılama gerekliliği ve ürün ömrünün değişken olması gibi nedenlerden dolayı işletmeler hızlı karar verme süreçlerinde sıkıntı yaşamaktadırlar. İşletmeler taleplerdeki değişimlerden dolayı çok fazla stok bulundurmak istemezlerken, aynı zamanda müşterilerin isteklerine zamanında cevap verebilmek için ise yeterli miktarda stok bulundurmak isterler. Üretim gerçekleştirilirken ortaya çıkan bu değişiklikler ve belirsizlikler nedeniyle işletmeler için çizelgeleme problemleri daha önemli hale gelmiştir (Kaya, 2014).

Çizelgeleme problemleri, üretim için kullanılan makinelerin optimum verimlilikle kullanılması, siparişlerin zamanında tamamlanması ve kaynakların daha verimli kullanılmasını amaçlamaktadır.

Çizelgeleme problemlerinin çözümü için işletmeler günümüze kadar çeşitli yöntemler geliştirmişlerdir. Klasik çizelgeleme yöntemleri çok fazla insan kaynağı ve zaman gerektirdiğinden, teknolojinin gelişmesiyle birlikte sezgisel yöntemlerden yararlanılmaya başlanmıştır. Sezgisel yöntemlerle elde edilen çözümler diğer

yöntemlere nazaran daha basit hesaplama teknikleri içirdiğinden ve daha kısa sürede gerçekleştiğinden günümüz araştırmacılarının diğer yöntemlere nazaran daha fazla tercih ettiği bir konu halini almıştır.

Çizelgeleme problemlerinin türü üretim ortamına göre farklılık göstermektedir. Üretim ortamında her bir işin kendine ait bir rotası var ve her iş sadece bir makinede yapılabiliyor ise burada kullanılan çizelgeleme problemlerine ATÇ (Atölye Tipi Çizelgeleme) problemleri denir. Şayet üretim ortamında paralel makineler var ise yani bir iş birden fazla makine üzerinde gerçekleştirilebiliyor ise bu tür üretim ortamlarında kullanılan çizelgeleme problemlerine ise EATÇ (Esnek Atölye Tipi Çizelgeleme) problemleri denir. Çizelgeleme sonucunda, her işin hangi makinelerde, ne zaman ve hangi sürelerde gerçekleşeceği belirlenmiş olur.

Gerçek bir üretim ortamında üretim süreci başladıktan sonra, makine arızası, planlı bakımlar, kaynakların zamanında yetişmemesi, yeni ve acil işlerin gelmesi gibi dinamik olaylar çizelgeleme sonucunun değişmesini gerektirmektedir. Dinamik bir olay gerçekleştikten sonra mevcut çizelgeleme sonucunun güncellenmesi ve ilk yapılan çizelgeleme sonucuna göre sapmanın en aza indirilmesi gerekir. Bu tür çizelgeleme problemleri ise "Dinamik Çizelgeleme Problemleri" olarak adlandırılmaktadır.

Yapmış olduğumuz tez çalışmasında literatürde daha önce çalışılmış olan EATÇ problemlerinin sezgisel yöntemlerle yapılan çözümleri incelenmiş, PSO (Parçacık Sürü Optimizasyonu) ve YA (Yerel Arama) metotlarının kullanıldığı, DEATÇ (Dinamik Esnek Atölye Tipi Çizelgeleme) problemlerinde kullanılabilen hibrit bir algoritma geliştirilmiştir (HA). Bu algoritmanın sonuçlarının alınabildiği, dinamik olayların test edilebildiği bir otomasyon programı, C# yazılım dili kullanılarak geliştirilmiştir. Ayrıca, dinamik olayların çizelgeleme sonuçlarına etkisinin test edilebilmesi, Endüstri 4.0'ın gerektirdiği Nesnelerin İnterneti teknolojilerinin üretim makineleri üzerinde denenmesi amacıyla akıllı üretim makinalarını temsil eden Raspberry Pi Mini bilgisayarların kullanıldığı örnek bir atölye ortamı oluşturulmuştur. Hazırlamış olduğumuz deney seti üzerinde dinamik olay olarak makine arızası ele

alındı. Üretim sırasında oluşabilecek bir arıza sonrası çizelgelemenin yenilenmesini sağlayarak üretimin küçük bir sapmayla devam edebilmesini sağlamış olduk. Çizelgeleme sonucun makinelere bildirilmesi, arızanın tekrardan sunucu bilgisayara bildirilmesi ve çizelgelemenin yenilenerek tekrardan makinelere gönderilip üretimin devam ettirilmesi işlemler nesnelerin interneti teknolojileri kullanılarak, insan müdahalesi olmadan otomatik olarak gerçekleştirilebilmiştir.

2. ÖNCEKİ ÇALIŞMALAR

Seçmiş olduğumuz konu, hem üretim planlama hem Nesnelerin İnterneti hem de Akıllı Makine kavramlarını kapsamaktadır. Bu tez çalışmasında bu konularla alakalı ulusal ve uluslararası literatür taraması neticesinde edindiğimiz bilgileri paylaşıp bu bilgiler ışığında Nesnelerin İnterneti teknolojisini üretim planlama (Dinamik Çizelgeleme) işlemlerinde kullanılmasını örnek bir uygulama ile gerçekleştirmeye çalışacağız. Şuana kadar yaptığımız literatür taramalarında çizelgeleme problemleri çözülürken daha çok statik durumlar için çalışmalar yapıldığını gözlemledik.

Hauber-Davidson ve Idris, akıllı su sayaçları için çeşitli tasarımları tarif etmektedirler. Bu sayaçlar su giriş ve çıkış miktarlarını ölçüp muhtemel su sızıntılarını tespit etmek ve verimli analizler yapmak için veri toplamaktadırlar (Hauber-Davidson ve Idris, 2006).

Bo ve Guangwen, bir tedarik zinciri yönetimini gerçek zamanlı gerçekleştirmek için bir uygulama geliştirmişlerdir. RFID (Radio Frequency Identification) etiketleri kullanılarak geliştirilen uygulamada etiketler ürünü otomatik olarak tanımlar ve konum bilgisi ile birlikte IoT ağında paylaşılır. Bu sistemle geçmişteki talepler incelenerek gelecek talepler üzerine çok verimli analizler gerçekleştirilebilir (Bo ve Guangwen, 2009).

Zhao ve arkadaşları, tarım alanında en çok seralarda kullanılan IoT teknolojileriyle ilgili yapılan çalışmaları incelemişlerdir. Bu çalışmalarda genel olarak sıcaklık, nem, toprak durumu gibi parametreler gerçek zamanlı ölçülüp sonuçlar analiz edilmek üzere bir sunucu üzerine kaydedilir (Zhao ve ark, 2010).

Lee ve Cho, sıcaklık, hava durumu, zaman ve mekân gibi kavramlara göre müzik öneren “Logmusic” adlı bir eğlence uygulaması geliştirmişlerdir (Lee ve Cho, 2011).

Dimitrakopoulos, ulaşım sektöründe gerçekleştirilmiş pek çok uygulama türünü yapmış olduğu çalışmada anlatmıştır (Dimitrakopoulos, 2011).

Yu ve arkadaşları insan vücudunun duruşlarını analiz etmek ve bu analiz sonucunda işlemler gerçekleştirilmesi için bilgisayar tabanlı teknikler kullanmışlardır (Yu ve ark, 2012).

Liang ve arkadaşları, çeşitli algılayıcı verilerinin yardımıyla bireyin sosyal faaliyetlerini takip eden "Circle Sense" adlı uygulamayı geliştirmişlerdir. Bu uygulama, sosyal faaliyet kalıplarını ve sosyal faaliyetlerde bulunan insanları analiz ederek kişinin sosyal çevresini tanımlar. Bu doğrultuda çeşitli sosyal etkinliklere katılan insanları guruplar. GPS algılayıcı kullanarak kişinin bulunduğu yeri, Bluetooth kullanarak ise çevresindeki benzer faaliyette bulunan insanları tespit eder. Bu sistem makine öğrenme algoritmaları kullanarak sürekli kendini geliştirmektedir (Liang ve ark, 2013).

Kundakçı, 2013 yılında yapmış olduğu çalışmasında, Tabu Arama ve Genetik Algoritma yöntemlerini kullanarak dinamik faktörlerin (Yeni İş, Makine Arızası, Takvim Değişikliği vb.) göz önünde bulundurulduğu dinamik atölye tipi iş çizelgeme problemlerine çözüm aramış ve elde edilen sonuçları diğer çalışmalarla karşılaştırmıştır (Kundakçı, 2013).

Çörekçi, 2014 yılında yapmış olduğu çalışmasında, 4 iş merkezli, 4’er adet birbirinin benzeri iş yapan makineler bulunan aynı zamanda 50 farklı ürün üretebilen bir atölye tasarlamıştır. Tasarlanmış olan bu atölyede zamana duyarlı, sürekli kendini yenileyen dinamik bir modele sahip çizelgeleme simülasyonu geliştirmiştir (Çörekçi, 2014).

Kaya, 2014 yılında yapmış olduğu doktora çalışmasında, Parçacık Sürü Optimizasyonu (PSO) meta sezgisel yöntemini kullanılarak Esnek Atölye Tipi

Çizelgeleme Algoritması geliştirmiştir. Algoritma, makine arızası, kaynak tükenmesi vb. gibi durumları hesaba katılmadan sonuç elde etmektedir (Kaya, 2014).

Jiang ve arkadaşları, 2016 yılında yapmış olduğu çalışmalarında, IoT teknolojilerini kullanarak çizelgeleme sürecindeki bozulmayı gerçek zamanlı tespit eden ve yeniden çizelgeleme yapan bir model önermişlerdir. Bu model, bozulmanın süreci hedeften ne kadar saptırdığını tespit edip, bu sapmayı en aza indirgeyen yeni çizelgelemeyi, Gelişmiş Karınca Koloni Optimizasyonu sezgisel yöntemini kullanarak gerçekleştirir (Jiang ve ark, 2016).

Sethi ve Sarangi, 2017 yılında yapmış oldukları çalışmalarında, Nesnelerin İnterneti(IoT) mimarileri, protokolleri ve uygulamalarını incelemiş ve bu teknolojiler hakkında bilgiler sunmuştur. Aynı zamanda bu çalışma ile IoT teknolojileri için yeni bir taksonomi önermektedirler (Sethi ve Sarangi, 2017).

Trstenjaka ve Cosic, 2017 yılında yapmış olduğu çalışmalarında, Endüstri 4.0 kavramı içerisinde üretim planlama ve çizelgeleme işlemlerinin nasıl olması gerektiği konusunda öneriler sunmuşlardır. Uygulanabilirliği, avantajları ve dezavantajlarını detaylı bir şekilde ele almışlardır (Trstenjaka ve Cosic, 2017).

Wang ve arkadaşları, 2017 yılında yapmış oldukları çalışmalarında, Sezgisel yöntemlerden olan Genetik Algoritmayı kullanarak Dinamik Esnek Atölye Tipi Çizelgeleme problemleri için çözüm önerileri getirmişlerdir. Bu çalışmada DE1 (Belirli bir zaman aralığında makine arızasının oluşması), DE2 (Bir işin tamamının bozulması), DE3 (Bir işin sadece bir operasyonunun bozulması) ve DE4 (Yeni acil bir işin gelmesi) olmak üzere 4 farklı dinamik olay göz ününde bulundurularak problem çözülmeye çalışılmıştır. Elde edilen sonuçlar farklı çalışmaların sonuçlarıyla karşılaştırılarak değerlendirmeler yapılmıştır (Wang ve ark, 2017).

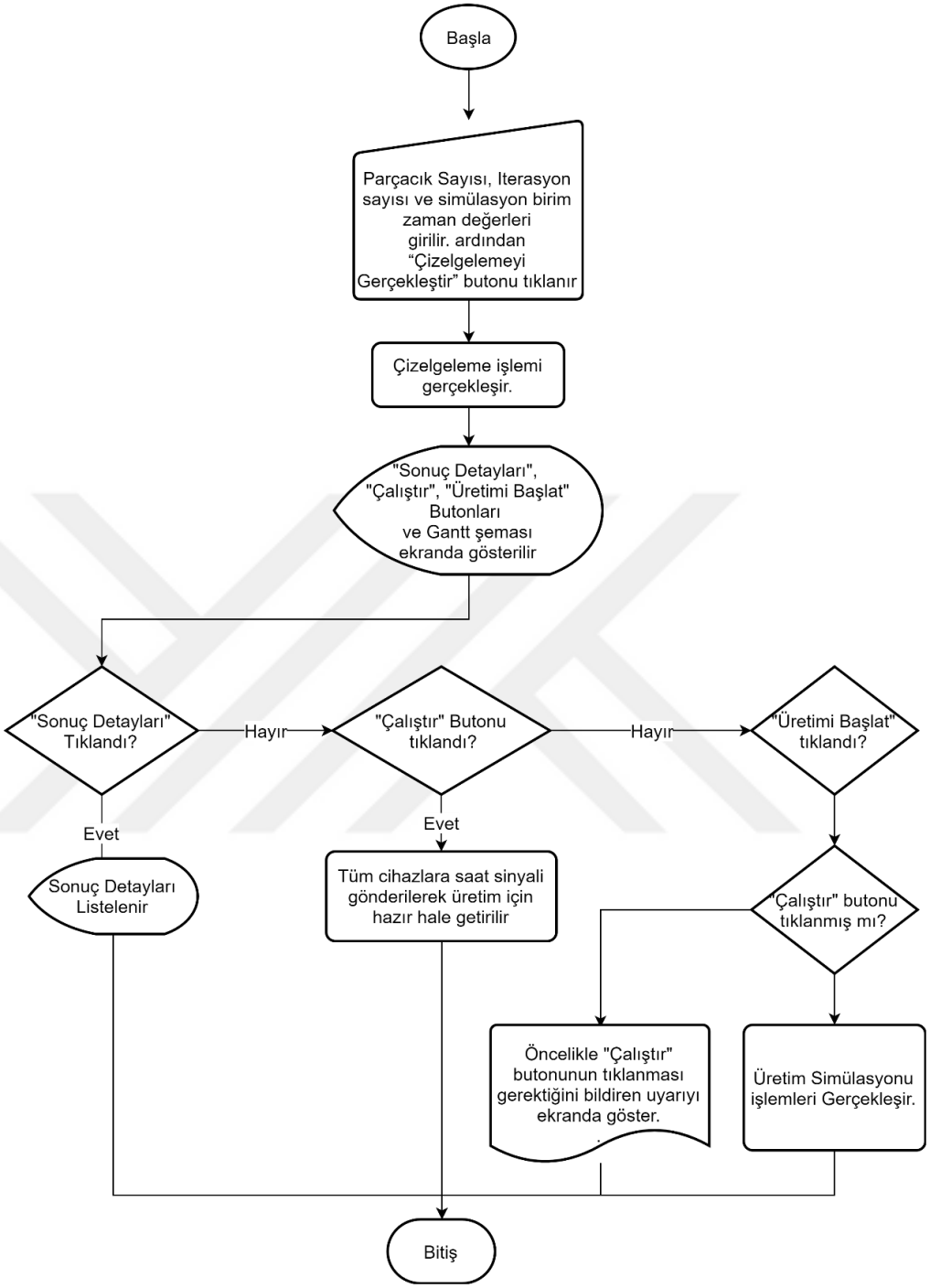
Aktaş, 2018 yılında yapmış olduğu çalışmasında, gömülü sistemler kullanarak üretim cihazlarından anlık verileri almış ve geliştirmiş oldukları otomasyon aracılığıyla anlık verileri yorumlayıp dijital ortama aktarılmasını sağlamışlardır. Bu

sayede elde edilen veriler anlık olarak raporlanarak sonraki süreçlerin iyileştirilmesine katkı sağlanmıştır (Aktaş, 2018).

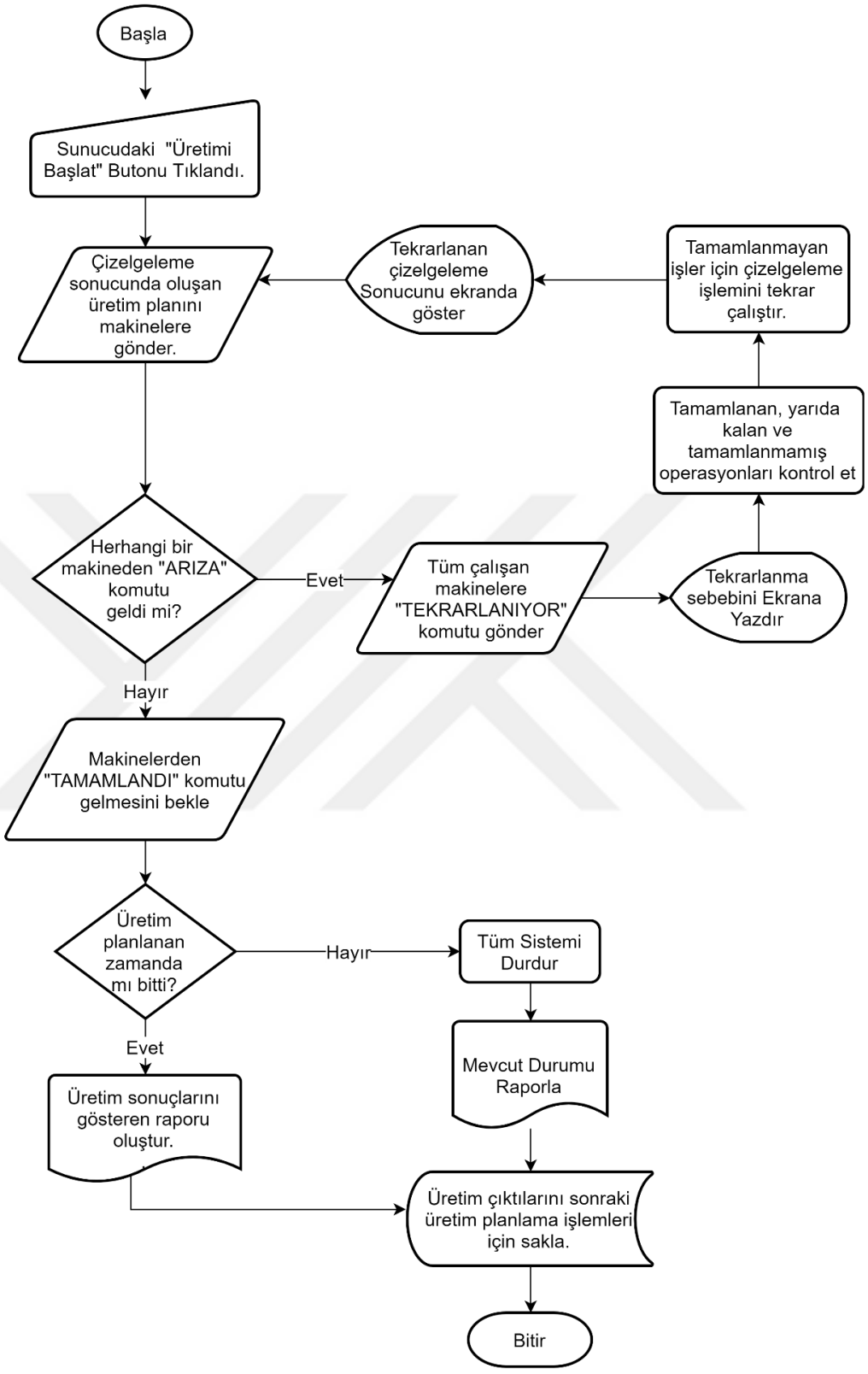
Tian ve arkadaşları, 2019 yılında yapmış oldukları çalışmalarında, dinamik esnek atölye tipi çizelgeleme (yeniden çizelgeleme) problemlerini çözebilen, oyun teorisi metodu ve Rolling horizon metotlarının birleştirilmesiyle geliştirilen hibrit bir algoritma ile çözüm önermişlerdir. Endüstriyel nesnelerin interneti teknolojilerini kullanarak üretim bilgi yönetim sistemi tasarlanmış ve literatürde var olan veri setleri üzerinde dinamik olaylarla birlikte çizelgeleme sonuçları elde ederek farklı çalışmalar ile karşılaştırmalar yapmışlardır (Tian ve ark, 2019).

3. MATERYAL ve YÖNTEM

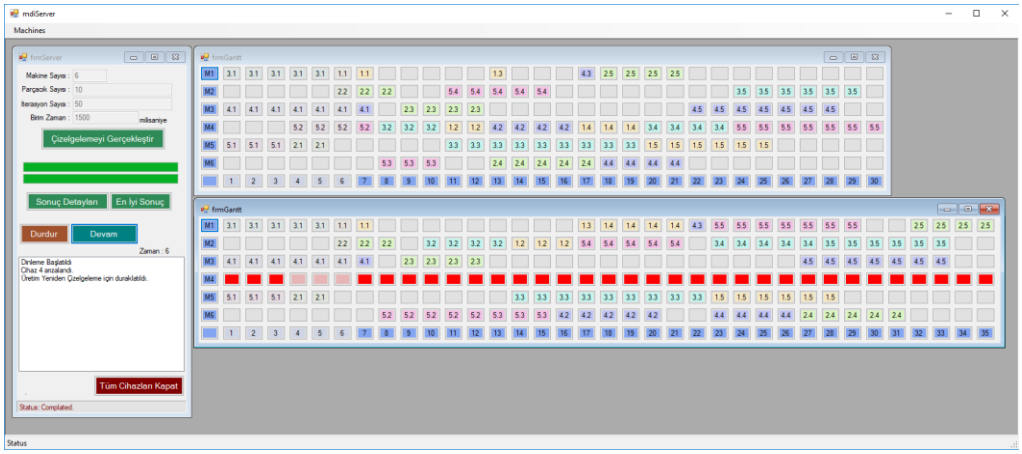
Çizelgeleme, belirli sayıdaki işlerin tamamlanabilmesi amacıyla elde bulunan kaynakların zamana bağlı olarak atanmasıdır. NP-zor problem olan çizelgeleme çözümlerinde en uzun işlem zamanı (LPT), en kısa işlem zamanı (SPT) ve en küçük teslim tarihi (EDD) gibi klasik yöntemler kullanılabilse de Meta sezgisel yöntemlerin gelişimiyle birlikte Genetik Algoritma, Fuzzy kontrol ve Parçacık Sürü Optimizasyon gibi sezgisel yöntemler kullanılmıştır. Ciddi boyutlarda zaman, kaynak ve enerji tasarrufu sağlanabilmektedir (Kaya, 2014). Bu çalışmada DEATÇ problemlerinin çözülebilmesi için PSO ve YA tabanlı hibrit bir algoritma (HA) geliştirilmiştir. IoT teknolojileri ve C# yazılım dili kullanılarak, makineler ve üretim bandından anlık durum verilerinin alınmasını, analiz edilmesini, çizelgelemenin yenilenmesini gerektirecek bir olay sonrasında çizelgelemenin sunucu tarafından otomatik olarak yenilenmesini sağlayacak donanım ve yazılım alt yapısı tasarlanarak bir deney ortamı oluşturulmuştur. Bu deney ortamında rastgele dinamik olaylar oluşturularak dinamik çizelgeleme problemleri için önermiş olduğumuz algoritmanın verimliliği ölçülmüştür. Bu dinamik olayların hızlı bir şekilde tespit edilip çizelgelenin yeniden gerçekleştirilmesi ve gelecekteki üretim planlamalarında kullanılmak üzere ilgili olayların kayıt altına alınması ve raporlanması için IoT Teknolojileri kullanılmıştır. C# yazılım dili ile geliştirmiş olduğumuz uygulamada test edilmek istenen veri setine ait makine sayısı, iş ve operasyon sayıları, operasyonlara ait işlem süreleri gibi veriler arka planda sisteme girildikten sonra açılan sayfa üzerinde HA algoritmasının çalıştırılabilmesi için gerekli olan parçacık sayısı ve iterasyon sayısı bilgileri girilir. Simülasyon içerisindeki operasyonların işlem zamanlarını belirleyecek olan birim zaman değeri de girildikten sonra "Çizelgelemeyi Gerçekleştir" butonu tıklanarak HA algoritması çalıştırılır ve maksimum tamamlanma süresi elde edilerek Gantt şeması görsel olarak oluşturulur. "Üretimi Başlat" butonu kullanılarak üretim makinalar üzerinde başlatılır ve üretim süreci görsel olarak izlenebilir. Hazırlamış olduğumuz uygulamanın genel sürecini gösteren akış diyagramı Şekil 3.1. de, Sunucu bilgisayar üzerinde gerçekleşen süreci gösteren akış diyagramı ise Şekil 3.2. de verilmiştir.



Şekil 3.1. C# Uygulamasına Ait Genel Süreci Gösteren Akış Diyagramı



Şekil 3.2. C# Sunucu Bilgisayar Üzerinde Gerçekleşen Süreci Gösteren Akış Diyagramı

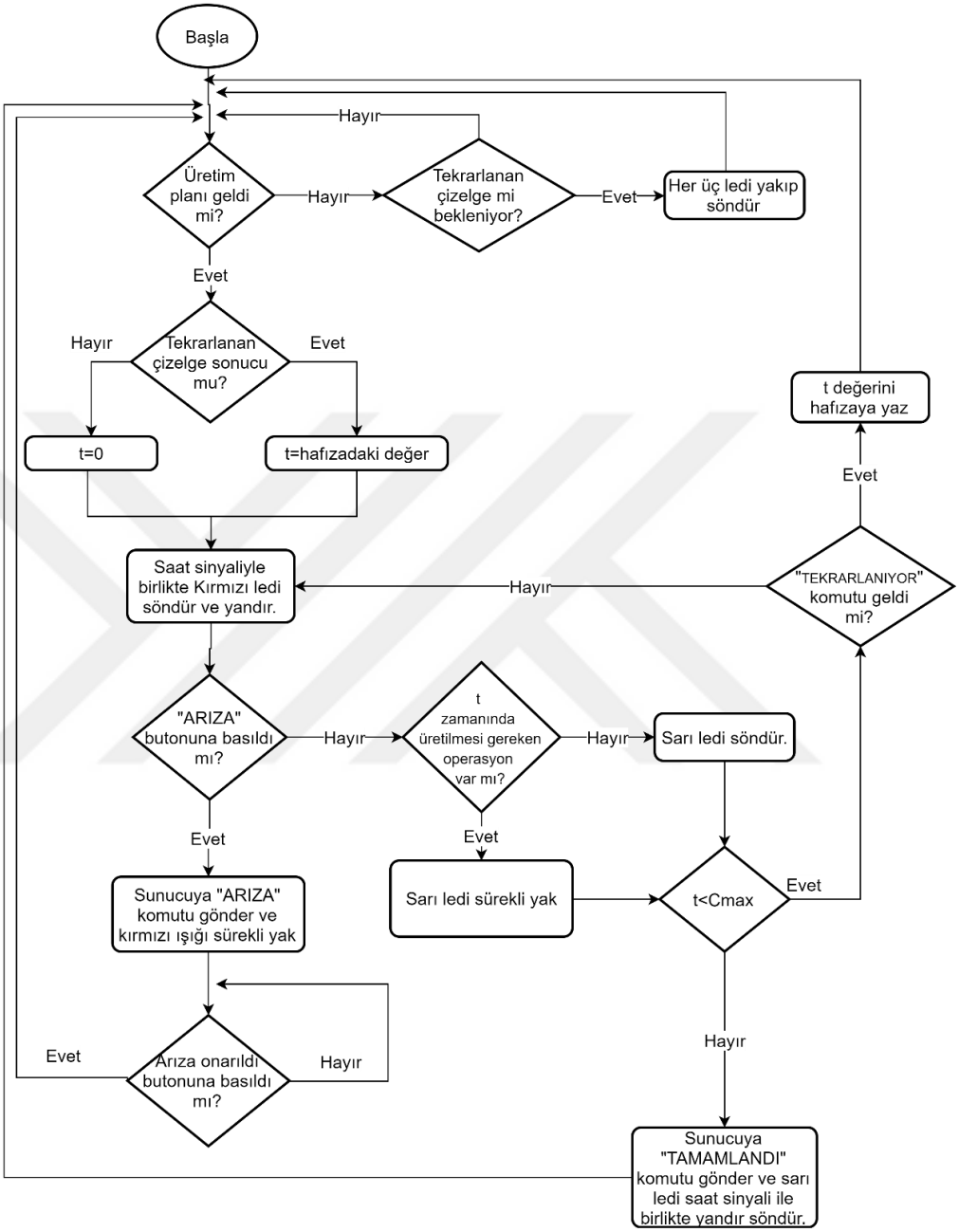


Şekil 3.3. C# Yazılım Dili Kullanılarak Geliştirilen Otomasyon Görseli

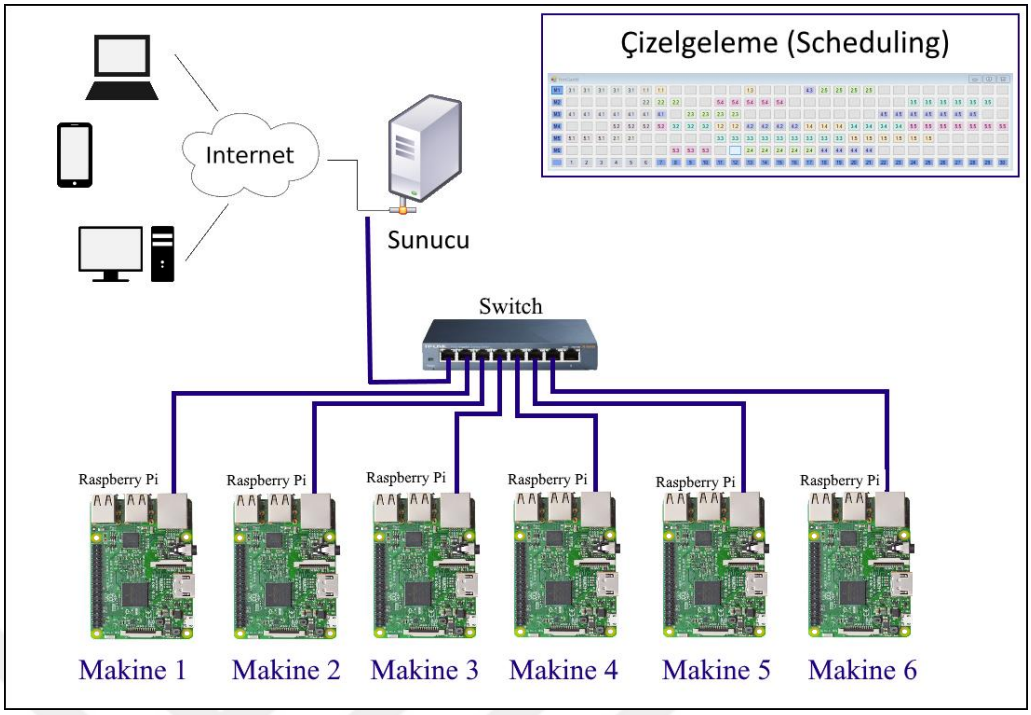
Makineler (Şekil 3.5.) üzerinde bulunan "Arıza Oluştur" butonu ile herhangi bir makinede arıza üretilebilmektedir. Makineler IoT teknolojileri kullanılarak akıllı hale getirilmiş ve sunucu ile haberleştirilerek çizelgeleme işlemlerini gerçekleştirmeleri sağlanmıştır. Bu sayede sunucu üzerinde elde edilen çizelgeleme sonucu doğrultusunda makineler üretim için başlatılacak ve gerek duyulması halinde durdurulabilecektir. Makineler üzerinde gerçekleşen süreçleri gösteren akış diyagramı Şekil 3.4. de verilmiştir.

2016 yılında TÜBİTAK özel sektörler arasında bulunan 1000 kuruluşla birlikte bir araştırma yapmıştır. Yapılan bu araştırma neticesinde sanayimizin Endüstri 2.0 ve Endüstri 3.0 arasındaki bir konumda olduğu, yani üretimde kullanılan çoğu makinenin bilgisayarlı (akıllı) makine olmadığını göstermektedir. Akıllı üretim makinelerinin maliyetlerinin yüksek olması nedeni ile mevcut makinelerin akıllı hale getirilmesi için yapılacak çalışmaların, Endüstri 4.0 a geçiş açısından firmalara kolaylık sağlayacağını düşünmekteyiz. Raspberry Pi (IoT uygulamaları yapmak için kullanabileceğiniz küçük ve ekonomik bir bilgisayardır.) mini bilgisayarı bu amaçla kullanılabilmektedir (Zhong ve Liang, 2016). Geliştirmiş olduğumuz otomasyon üzerinde gerçekleştirdiğimiz senaryoları uygulamak için bir deney ortamı hazırlandı. Sanayide kullanılan akıllı makineleri temsil etmesi açısından Raspberry Pi mini bilgisayarı kullanıldı. Bu cihazlardan örnek olarak 6 adet kullanılmıştır. Her cihaz bir üretim makinesini temsil etmektedir. Bu cihazlar birbirleriyle veya sunucu ile ağ üzerinden

haberleşmeleri sağlanmıştır. Oluşturulan bu deney ortamı Şekil 3.5.'deki tasarım ile anlatılmaya çalışılmıştır.



Şekil 3.4. Makineler Üzerinde Gerçekleşen Süreçleri Gösteren Akış Diyagramı



Şekil 3.5. Sanayide kullanılan makinaları temsil eden Raspberry Pi mini bilgisayarı ve sistem tasarımı

Bu bölümün devamında çalışmamızda kullandığımız teknolojiler ve yöntemlerin ne olduğu hakkında bilgilendirme yapılacaktır.

3.1. Çizelgeleme Problemleri

Çizelgeleme, mevcut kaynakların ve araçların belirli bir zaman aralığında işlere tahsis edilmesidir. Bu tahsis işlemi sırasında bir veya birden çok amacın optimum olması hedeflenir. Çizelgeleme süreci işletmelerin birçoğunda karar verme süreci olarak da değerlendirilirler (Pinedo, 2008). Farklı bir tanım yapmak istersek, belirli bir zamanda üretim yapan bir işletme içerisinde yapılması planlanan işlerin, ne zaman ve hangi sırayla yapılacaklarının belirlenmesi ve bu işlere kaynakların optimum bir şekilde tahsis edilmesidir (Mutlu, 1993). Çizelgeleme, kısıtlı kaynak ve kısıtlı zaman aralığında işlerin doğru zamanda doğru kaynağa atanmasıyla ilgili bir karar verme sürecidir (Büyüksünetçi, 2006).

Bir işletmedeki işler ve kaynaklar farklılıklar gösterebilirler. Kaynaklar bir üretim atölyesindeki makineler olabilir veya bir bilgisayardaki işlem birimleri veya bir

havaalanındaki pistler olabilir. İşler ise bir havaalanındaki kalkış ve inişler, bir üretim bandındaki işlemler, bir bilgisayar programının yürütülmesi gibidir. Her işin belirli bir öncelik ve zaman kısıdıyla hedefleri vardır. Bu hedefler farklı şekillerde olabilirler, örneğin bir işin mümkün olan en kısa sürede bitmesi, kaynak kullanımının en aza indirgenmesi gibi (Pinedo, 2008).

Çizelgeleme işlemleri, matematiksel teknikler ve sezgisel yöntemler kullanılarak gerçekleştirilirler. Bu sayede işlerin tamamlanması sırasında zaman ve kaynak tasarrufu sağlanmış olur. Tabi bu çizelgeleme problemlerinde her zaman daha iyisini bulmak mümkündür o nedenle en iyisini bulmak için sürekli faklı farklı teknikler denenebilmektedir. İşlerin sırasını ve kaynak kullanımlarını belirlemek için kullanılmakta ve müşteri memnuniyetini maksimum düzeyde tutabilmek için siparişler zamanında hazırlanabilmektedir. Makinelerin bulunduğu bir işletmede çizelgeleme yaparken yapılması gerekenler aşağıdaki gibi sıralanabilir (Aladağ, 2010):

1. İşlerin hangi makinelerde yapılabileceği belirlenmeli?
2. İşin teslim tarihi göz önünde bulundurularak işlemlerin başlama ve bitiş zamanı belirlenmeli
3. İşlemler makinelere atanmalı
4. Bu işlemlerin makineler üzerindeki sıralamaları yapılmalı

Çizelgeleme problemlerine problem çözümü mantığıyla bakıldığında, çizelgeleme problemleri bir optimizasyon problemidir. Her zaman en iyi sonuç elde edilmeye çalışılır. Bu en iyi sonuç bazen üretim zamanını minimize edilmesi, bazen de belirli bir zaman diliminde maksimum iş yapmak olabilir. Çizelgelemeye karar alma mantığı ile bakıldığında ise çizelgeleme bir karar verme sürecidir. Bu süreçte hata yapmamak ve en iyi kararı alabilmek için geçmişte yapılan işler, sonuçlar ele alınır ve olabilecek tüm riskler göz önünde bulundurularak kararlar verilir. Sorumlu kişi ne kadar fazla anlamlı veriye sahipse hata yapma oranı bu oranda düşecek ve başarıya ulaşması kolaylaşacaktır. IoT teknolojilerini kullanmak bu anlamada çok yararlı olacaktır. Kurumsal anlamda ise çizelgeleme, üretim planlama sürecinin olmazsa olmaz bir parçasıdır (Kundakçı, 2013).

Çizelgeleme problemlerinde ulaşılmak istenen amaçlar/hedefler mevcuttur. Bu hedefleri aşağıdaki gibi sıralamak istersek;

1. Bir işin tamamlanma süreci teslim tarihine göre ayarlanmaya çalışılır. Bu nedenle teslim tarihinden önce bitmesi gerekmektedir.
2. İşlem sürelerinin en aza indirgenmesi gerekmektedir. Bu sayede teslim tarihinden önce bitirmemiz kolaylaşacaktır.
3. Üretim kaynaklarının (Makine, personel vs.) en etkin ve verimli kullanılması gerekmektedir (Boray, 2007).

Çizelgeleme problemleri karmaşık bir yapıya sahip olduğundan anlaşılması veya anlatılmasında sıkıntılar yaşanabilmektedir. Bu nedenle bu karmaşıklığı azaltmak için biçimsel yaklaşımlardan faydalanılır. Çizelgeleme problemlerinde yaygın olarak kullanılan Gantt şeması bu biçimsel yaklaşımlardan bir tanesidir. Gantt şemasının en temel özelliği kaynakların zaman içirişindeki dağılımını göstermektir (Baker ve Trietsch, 2009). Yani işlemlerin (operasyonların) zaman çizelgesi üzerinde sıralanışını göstermektedir. Sorumlu kişiler Gantt şemalarına bakarak boş kalan makineler veya çakışan işler gibi durumları yorumlayabilmektedirler. Gantt şeması görüntüleme aracı olarak çok iyi bit model olmasına karşın çizelgenin iyileştirilmesi yönünde bir yöntem sunmaz. Sorumlu kişinin kendi sezgilerini ve yöntemlerini kullanması gerekmektedir. (Seçme, 2006).

Günümüz işletmeleri, gelişen teknolojiyle birlikte hızla artan müşteri taleplerine karşı karar verme süreçlerinde çok ciddi sıkıntılar yaşayabilmektedirler. Oluşan rekabet ortamı nedeniyle hızlı karar verme süreçlerini bünyelerine dâhil etme zorunluluğu doğmuştur. Müşteri taleplerinin değişim göstermesine karşın elinde stok bulundurmak istemezken, müşteri talebine hızlı bir şekilde cevap vermek için de sürekli stok bulundurmak isteyebilirler. Her iki amacı da optimize etmek bir zorunluluk haline gelmiştir (Kaya, 2014). Bu şekilde birbirleriyle ters orantılı farklı amaçların optimize edilmesi açısından çizelgeleme yöntemlerinin anlaşılması ve uygulanabilir olması önem arz etmektedir. Hem araştırmacılar hem de işletmeler için çizelgeleme problemleri konusu gün geçtikçe önemini arttırmaktadır.

Çizelgelemenin kullanılmasındaki amaç, üretim kaynaklarının optimize edilmesi, enerji verimliliğinin sağlanması, insan kaynaklarının verimli kullanılması ve müşteri taleplerinin zamanında ve eksiksiz karşılanmasıdır. Sonuç olarak İşlemelerin verimliliğinin arttırılması çalışmaları çizelgeleme konusudur.

Klasik algoritmalar kullanılarak çizelgeleme işlemleri gerçekleştirilmeye başlanmış sonrasında sezgisel yöntemlerin gelişimiyle birlikte sezgisel yöntemlerin çizelgeleme problemlerinde daha verimli, daha az karmaşık ve daha hızlı sonuç ürettiği görülmüştür.

İşletmelerin farklı üretim ortamına ve farklı ürün tiplerini üretmeleri gibi nedenlerden dolayı farklı çizelgeleme problemleri oluşmaktadır.

3.1.1. Statik çizelgeleme

Üretim işlemine girecek işler üretim periyodu süresince zamanla değişiklik göstermiyor ise, üretim ortamında hiç belirsizlik yok veya yok sayılıyor ise, bu çizelgeleme ortamı statik ortam olarak tanımlanır. Çizelgeleme işleminde işler, ortam veya kaynaklarda hiçbir değişiklik öngörülmüyor ise bu çizelgeme statik çizelgeleme olarak sınıflandırılır. Statik çizelgeme yöntemleri ile ilgili, anlaşılması daha kolay olduğundan gerçek yaşamda karşılığı olmasa da bu güne kadar araştırmacılar tarafından Dinamik çizelgelemeye göre daha çok çalışma yapılmıştır (Kundakçı, 2013).

3.1.2. Dinamik çizelgeleme

Çizelgeleme problemi içerisindeki tüm olayların çizelgeleme periyodu süresince değişiklik gösterdiği üretim ortamlarıdır. Gerçek üretim ortamlarında makine arızalanması, yeni acil siparişlerin gelmesi, kaynak tükenmesi ve kalite problemleri gibi nedenlerden üretimin durması veya kısmi olarak devam etmesi gerekebilmektedir. Bu nedenle dinamik çizelgeleme gerçek üretim ortamlarında daha uygulanabilir bir yöntemdir. Literatür çalışmalarındaki çizelgeleme çalışmaları ile gerçek uygulama

arasındaki boşlu kapatmak için araştırmacılar dinamik çizelgele yöntemlerini geliştirmişlerdir (Kundakçı, 2013). Uygulamadaki çizelgeleme işlemleri ile teoride anlatılan çizelgeleme işlemlerinin arasında büyük farklılıkların olduğunu belirterek bu alanda geliştirilmiş algoritmaların ve çizelgeleme modellerinin gerçek zamanlı kullanılan bilgiyi işlemede yetersiz olduklarını belirtmişlerdir. Bununla birlikte çizelgeleme alanında yapılan araştırmaların, çizelgeleme süreç denetleme ve kontrol mekanizmalarındaki teknolojik yeniliklere uyum sağlama konusunda yetersiz oldukları görüşünü bildirmektedirler. Gerçek yaşam problemleri ile teoride anlatılan çizelgeleme arasındaki farklılıkların en aza indirilmesi amacıyla gerçekleştirdikleri çalışmalarda, teknolojik sistemlerin üretmiş oldukları gerçek zamanlı veriyi teoride anlatılan çizelgeleme problemlerine adapte etmek amacıyla yeni bir yaklaşım önermişlerdir. Çizelgeleme alanında yapılan çalışmalarda da anlatılmak istendiği gibi, kitaplarda bahsi geçen çizelgeleme yaklaşımları ile üretim sahalarında meydana gelen çizelgeleme problemlerinin arasında büyük farklılıklar meydana gelmektedir. Üretim ortamlarında dinamik olaylar meydana geldiğinden, teoride anlatılan ile gerçek yaşam problemlerindeki farklılığı azaltmak amacıyla, çizelgeleme işlemleri yapılırken dinamik olaylar hesaba katılarak işlemler yapılmalıdır (Cowling ve Johansson, 2002).

3.1.3. Atölye Tipi Çizelgeleme (ATÇ) ve Esnek Atölye Tipi Çizelgeleme (EATÇ) problemleri

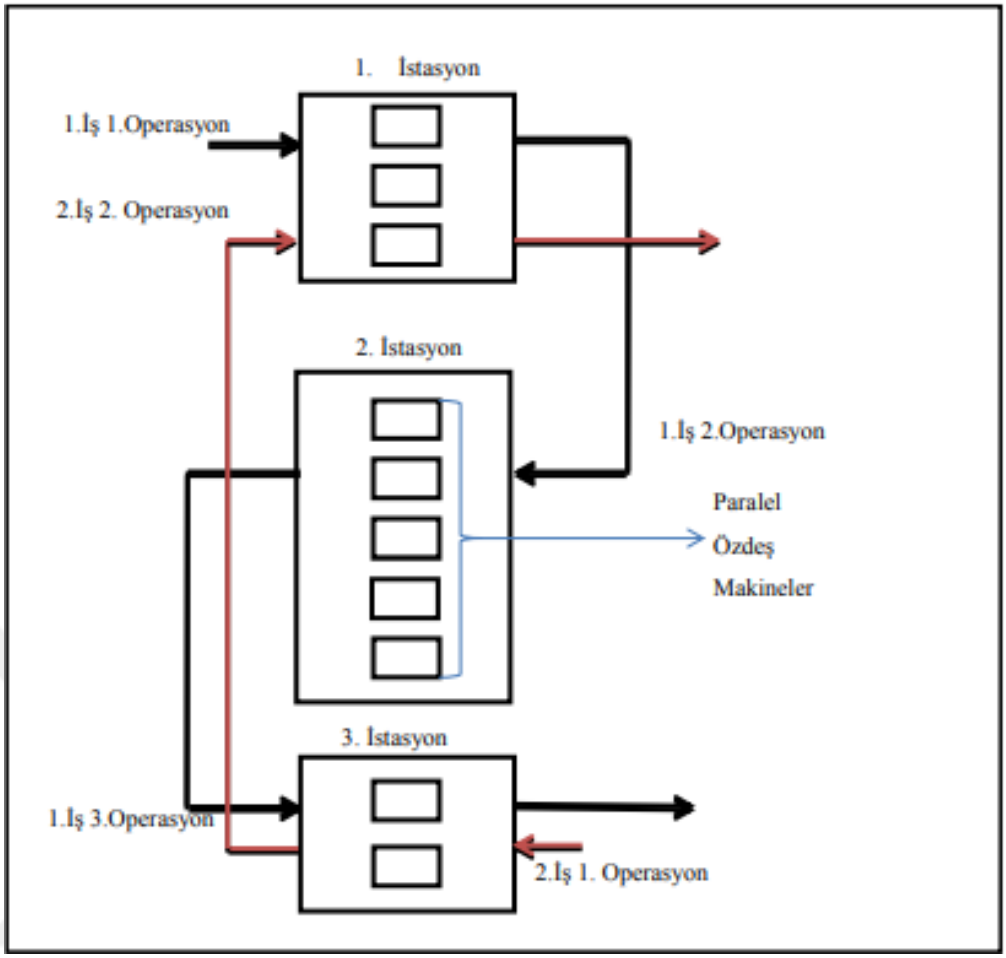
Üretim teknolojilerinin gelişmesiyle birlikte çok farklı işlemleri gerçekleştirebilen makineler üretim bantlarında yerlerini almışlardır. Günümüz üretim alanlarının gereksinimleri klasik çizelgeleme yöntemleriyle karşılamak mümkün değildir. Akademisyenler ve uzmanlar üretim bantlarına dâhil olan gelişmiş makinelerin özelliklerine karşılık verebilecek yeni çizelgeleme yöntemleri geliştirmişlerdir. Bu yöntemlerden bir tanesi ise EATÇ modelidir. Bu modelin çözümleri içinde algoritmalar oluşturulmuştur. Bu modelin desteklediği makineler farklı zaman dilimlerinde farklı işleri üretebilme yeteneğine sahiptir. Şöyle ki her bir işin her operasyonu farklı zamanlarda aynı işleri yapabilen farklı makinelerde üretilebilir. EATÇ modeliyle birlikte iki problem karşımıza çıkmaktadır.

1. Hangi işin hangi makineye yükleneceği problemi
2. Makinelere yüklenen işlerin hangi sırayla işleneceği problemi

Bu problemin türünde çözüm üretmek için, araştırmacılar hiyerarşik ve bütünleşik olarak isimlendirdikleri iki çözüm önermektedir. Yükleme ve sıralama işlemleri aynı zamanda yapıldığında bu çözüm bütünleşik olarak adlandırılmaktadır. Yükleme ve sıralama işlemler farklı zamanlarda yapıldığında ise hiyerarşik çözüm olarak adlandırılmaktadır.

Her bir operasyonun tek bir makinede işlem görebildiği problem türleri ise ATÇ olarak adlandırılmaktadır. Bu nedenle EATÇ problemleri ATÇ problemlerine nazaran daha karmaşık bir hal almaktadır. EATÇ problemlerinde bir operasyonun üretim ortamında bulunan tüm makineler üzerinde işlenebilme durumu söz konusu ise yani her bir operasyonun farklı bir makinede işlenebilme alternatifi mevcut ise "Tam Esnek", sadece belirli bir gurup makine üzerinde işlenebilme durumu söz konusu ise "Kısmi Esnek" olarak nitelendirilir.

ATÇ problemleri ile Paralel Makine Çizelgeleme (PMÇ) problemlerinin birleşiminden oluşan EATÇ problemlerine örnek bir esnek atölye tipi üretim ortamı Şekil 3.6.'da gösterilmektedir. Şekilde görüldüğü gibi paralel makinelerden oluşan iş bantları vardır. Bu tür üretim ortamlarında, paralel makinelerden oluşan üretim bantları vardır. Bu üretim bantlarının içerisinde ise benzer işleri yapabilecek kabiliyette özdeş makineler bulunmaktadır (Kaya, 2014).



Şekil 3.6. Esnek Atölye Tipi (EAT) üretim ortamı

3.1.4. Parçacık Sürü Optimizasyonu (PSO)

Diğer sezgisel algoritmalarla karşılaştırıldığında çok fazla hafızaya ihtiyaç duyulmayan ve aynı zamanda etkili bir hesaplama tekniğine sahip, uygulanması kolay bir algoritmadır. PSO algoritmasında amaç yerel optimum sonuçlara takılmadan en iyi sonuca ulaşmaktır. Dr. Kaya'nın 2014 yılında EATÇ problemlerinin çözümü için geliştirmiş/iyileştirmiş olduğu PSO algoritması, örnek bir problem üzerinden adım adım anlatılacaktır. Çizelge 3.1.' de örnek probleme ait iş, operasyon ve makine-işlem süreleri verilmiştir (Kaya, 2014)

Çizelge 3.1. 3x4x8 Boyutlu EATÇ problemi makine işlem süreleri

İşler	Operasyonlar	Makine İşlem Süreleri			
		M_4	M_3	M_2	M_1
J_1	O_{11}	2	3	7	4
	O_{12}	5	3	1	3
	O_{13}	7	6	5	4
J_2	O_{21}	6	4	3	1
	O_{22}	2	3	5	4
	O_{23}	3	1	2	6
J_3	O_{31}	5	4	3	8
	O_{32}	2	6	3	4

Bu örneğimizde 3 adet iş (J_1,J_2,J_3), Her işe bağlı operasyonlar J_1 (O_{11} - O_{12} - O_{13}), J_2 (O_{21} - O_{22} - O_{23}), J_3 (O_{31} - O_{32}) ve 4 adet makine (M_1 - M_2 - M_3 - M_4) bulunmaktadır. Amaç fonksiyonumuz ise maksimum tamamlanma süresidir (C_{max}). Çizelgeleme problemlerinde PSO algoritması C_{max} değerini en küçük yapmak amacıyla uygulanacaktır.

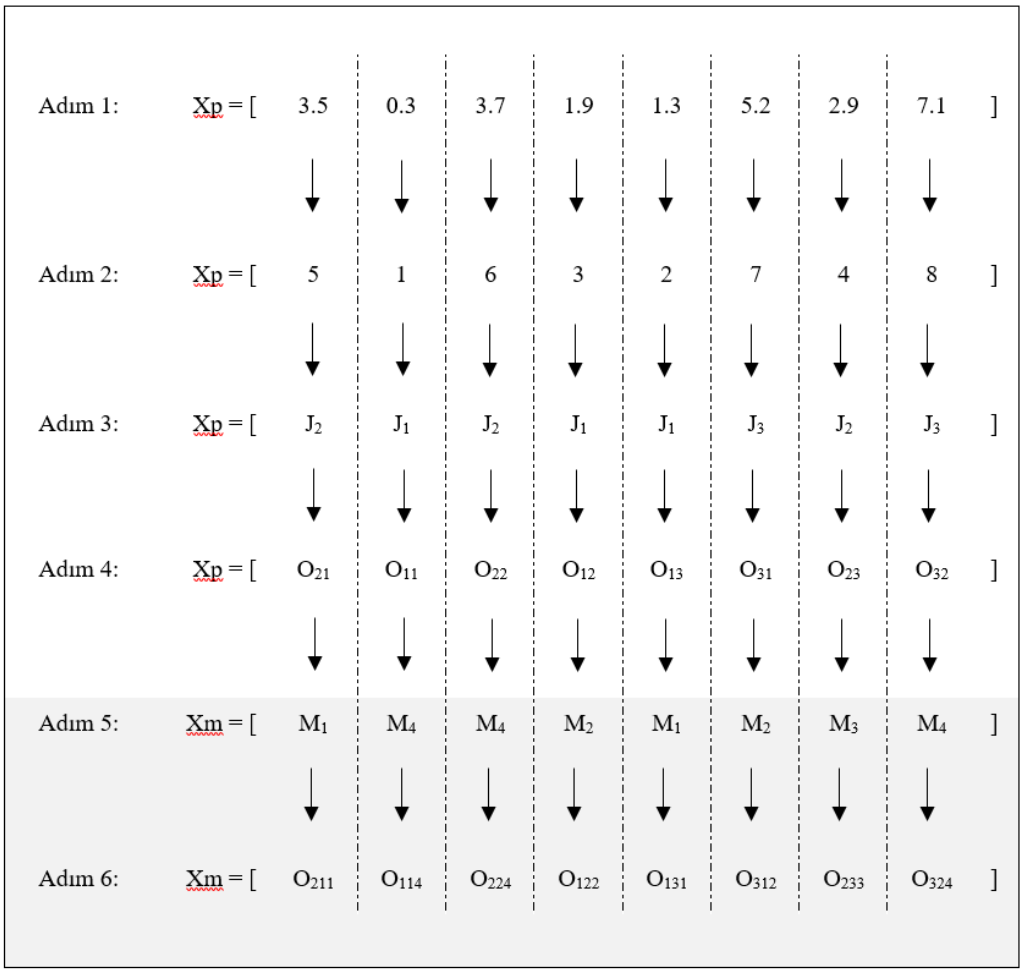
Adım 1: Öncelikle pozisyon vektörü ($Xp_)$ vektörü oluşturulur. Xp Vektörü operasyon sayısı kadar eleman içerir. k. iterasyon ve i. parçacığına ait pozisyon vektörü Xp_i^k şeklinde gösterilir. Bu örneğimizde k. iterasyon ve i. parçacığa ait Xp vektörü aşağıdaki gibidir;

Xp = [3.5 , 0.3 , 3.7 , 1.9 , 1.3 , 5.2 , 2.9 , 7]

Operasyon sayısı kadar rastsal değer üretilir. Rastsal değerler d operasyon sayısı olacak olursa [-d,d] aralığında seçilir.

Adım 2: 1. Adımda oluşturulan Xp vektörünün içerisinde bulunan rastsal rakamlar Şekil 3.7. deki gibi küçükten büyüğe doğru sıralanır.

Adım 3: Küçükten büyüğe doğru iş numaraları ve operasyon numaraları Şekil 3.7. deki gibi ilgili vektörün içerisine yazılarak atama yapılır. Örneğimizde J_1 e ait üç operasyon olduğunda ilk üç rakama J_1 ataması yapılmıştır.



Şekil 3.7. Xp Vektörünün operasyon işlem sırasına dönüşümü

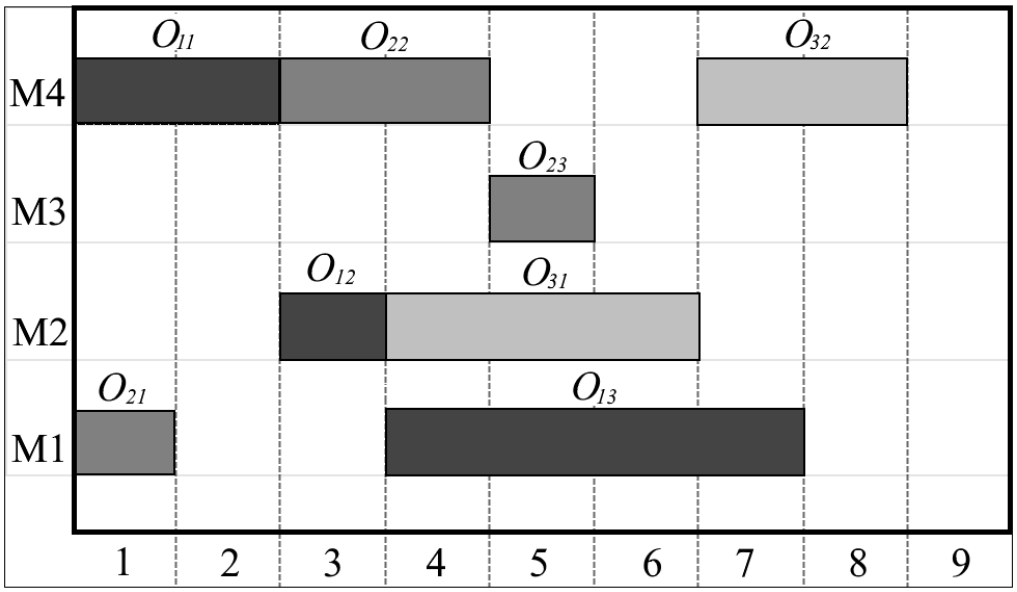
Adım 4: Her işin operasyonları arasında işlem sırası kısıtı olduğu için Xp vektörünün J_1 işine ait operasyonlar ilk üç rakama Şekil 3.7. deki gibi sırayla atanır.

Adım 5: Makine atamalarını temsil eden vektör ise Xm vektörüdür. PSO ile çözüm yapılırken Xm vektörünün boyutu operasyon sayısı kadardır. Xm vektörü EKİZ kuralına göre oluşturulur yani en kısa işlem süresi olan makine ilgili operasyona Şekil 3.7. deki gibi atanır

Adım 6: Birinci parçacığa ait Operasyon No ve makine atamalarını gösteren Xp_1^1 vektörü aşağıdaki gibi elde edilmiş olur. Burada 1. Parçacığa ait birinci sırada yer alan 2. İşin 1. Operasyonu 1. Makineye yerleştirilir. Diğer operasyonlarda aynı şekilde sırayla makinelere yerleştirilir.

Xp_1^1 = [O_{211}, O_{114}, O_{224}, O_{122}, O_{131}, O_{312}, O_{233}, O_{324}]

Adım 7: 1. Parçacığa ait çözümün Gantt şeması Şekil 3.8. daki gibi oluşturulur.



Şekil 3.8. İlk parçacığa ait Çözümün Gantt şeması

Gantt şemasına baktığımızda amaç fonksiyonumuz olan C_{max} = 8 değerinin elde edildiğini görmüş olacağız.

Şekil 3.7.' de anlatılan adımlarda biz sadece 1 parçacığa yani 1. İterasyonun 1. parçacığına ait çözüm için Xp, Xm vektörlerini ve Gantt şemasını oluşturduk. PSO algoritmasında bu adımlar her iterasyonun parçacık sayısı kadar tekrarlanacak ve bu parçacıklar arasında elde edilen optimum C_{max} değeri bizim yerel optimum değerimiz olacaktır. Çözüm işlemi tamamlandığında ise elimizde iterasyon sayısı kadar yerel C_{max} değerleri olacaktır. Elde edilen C_{max} değerlerinin arasındaki optimum C_{max} değeri ise bizim global optimum değerimizi yani PSO algoritmasından elde ettiğimiz nihai sonuç olacaktır.

PSO algoritması parçacıklar arası bilgi paylaşımıyla yol alır. İlk parçacık için Xp vektörü oluşturulurken Xp vektörünün elemanlarını [-d,d] arasında rastgele rakamlar seçerek belirlemiştik. Algoritma sonraki iterasyonlarda bulunan ilk

parçacığın eleman değerlerini rastgele değerler alarak belirlemeyecektir. İlgili değerlerin istenilen doğrultuda belirlenebilmesi için bir sonraki iterasyonda oluşturulacak parçacıkların elemanlarını **Hız Vektörünü (V_i)** kullanarak bulacaktır. Hız Vektörü parçacığın bir sonraki konumunu belirlemeye yarayan parametredir. V_i sayesinde parçacıklar arasında bilgi paylaşımı sağlanmış olacaktır. k. iterasyonun i. parçacığına ait hız vektörünün 1. elemanı V_{i1}^k şeklinde ifade edilir. Örneğimize ait hız vektörü denklem (3.1) de gösterildiği gibi ifade edilir. Hız vektörünün eleman sayısı Xp ve Xm vektörlerinde olduğu gibi operasyon sayısı kadardır.

$$V_i^k = [V_{i1}^k, V_{i2}^k, V_{i3}^k, V_{i4}^k, V_{i5}^k, V_{i6}^k, V_{i7}^k, V_{i8}^k] \quad (3.1)$$

$$V_i^{k+1} = W.V_i^k + C_1.Rand_1^k.(Pbest_i^k - X_i^k) + C_2.Rand_2^k.(Gbest_i^k - X_i^k) \quad (3.2)$$

$$W = W_{max} - (W_{max} - W_{min}).iter / iter_{max} \quad (3.3)$$

$$X_i^{k+1} = X_i^k + V_i^{k+1} \quad (3.4)$$

Bir adım sonraki iterasyonda kullanılacak olan konum vektörü hızını hesaplamak için denklem (3.2) de verilen fonksiyon kullanılacaktır. $Rand_1^k$ ve $Rand_2^k$ değişken değerleri her iterasyon için [0,1] arasında rastgele değerler olarak alınırlar (Kaya, 2014). Öğrenme faktörlerimiz ise C_1 ve C_2 değerleridir. Literatürde genel olarak $C_1 = C_2 = 2$ alınmaktadır (Kaya, 2014). Atalet ağırlığı olarak verilen W ise bir önceki iterasyonda hesaplanan hız vektörünün mevcut iterasyonda elde edilen hız vektörü değerine etkisini kontrol eder. W atalet ağırlığı denklem (3.3) de verilen formül ile hesaplanır. Bir parçacığa ait mevcut zamana kadar elde edilen en iyi yani minimum C_{max} değeri Pbest olarak isimlendirilir. Gbest ise popülasyon içerisinde elde edilmiş olan minimum sonuç değerine sahip parçacığın C_{max} değeridir. Parçacığın bir sonraki konumunu hesaplayan fonksiyon ise denklem (3.4) de verilmiştir.

Bu örneğimizde yer alan PSO algoritmasının tüm adımları, Kaya tarafından 2014 yılında yayımlanmıştır. Bu yayındaki algoritma statik EATÇ'yi çözümleyen çok amaçlı bir yöntemdir. Kayanın tezinde önerildiği gibi makine arızası, yeni gelen iş ve

benzeri dinamik durumlardan dolayı statik çizelgeleme her zaman yeterli olmamaktadır. Bu çalışmada PSO algoritmasının Dinamik Esnek Atölye Tipi Çizelgeleme (DEATÇ) üzerinde kullanımı gerçekleştirilmiştir (Kaya, 2014).

Dinamik çizelgelemede, üretim sürecinin herhangi bir zamanında üretim sonucunu veya zamanını etkileyecek bir olay gerçekleştiği zaman çizelgelemenin tekrar edilmesi gerekecektir. Yukarıda anlatılan adımlar tekrarlanacak fakat çizelgelemenin tekrarlanabilmesi için aşağıdaki verilerin toplanması gerekmektedir.

1. Dinamik olayın gerçekleştiği t_1 zaman bilgisi
2. Eğer dinamik olay bir makine arızası ise makine adı ve arızanın giderilme zamanı t_2
3. Eğer dinamik olay acil olan yeni bir işin gelmesi ise işle ilgili detaylı bilgiler
4. Her makinenin üzerlerindeki aktif işlerin bitiş zamanını dikkate alarak hangi zamanda yeni işin işlenebileceği bilgisi
5. Dinamik olayın gerçekleşme zamanına kadar tamamlanmış ve tamamlanmamış operasyonların bilgisi

Tekrar çizelgeleme yapılırken sadece geriye kalan operasyonlar üretileceğinden algoritmada kullanılan Xp vektörümüzün boyutu değişecektir. Statik çizelgeleme yapan algoritmamızda tüm makinelerin t=0 zamanında boş yani müsait olduğu kabul edilirken tekrarlanan çizelgelemede her makinenin başlama t zamanları değişkenlik gösterecektir. Bu nedenle algoritma adımları uygulanırken yani Gantt şemasını oluştururken bu başlama zamanları dikkate alınarak gerçekleştirilmesi gerekecektir.

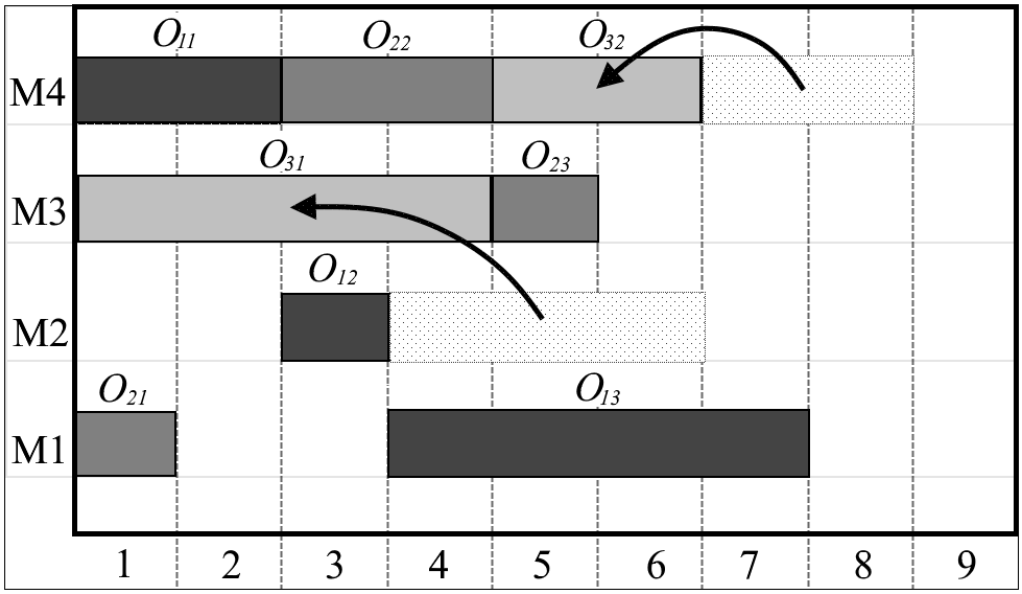
Bu tez çalışmamızda önerdiğimiz yöntemde makine sıralaması ve makine atama yöntemlerinde bir değişiklik olmazken Gantt şeması oluşturma esnasında makinelerin başlangıç zamanları dikkate alınacak ve bu şekilde yeni bir Gantt şeması oluşturulmuş olacaktır.

3.1.5. Yerel Arama (YA)

Çizelgeleme problemlerinde meta sezgisel yöntemler, çözüm uzayındaki tüm sonuçları denemeden, çözüm uzayının sadece bir kısmını, belirli kurallara bağlı kalarak arama yapar ve bulmuş olduğu sonuçların içerisinden optimuma en yakın olan sonucu seçerler. Bu nedenle bulunmuş olan sonucun optimum sonuç olduğu garanti edilemez. YA algoritmaları bulunmuş olan sonuçların iyileştirilmesini sağlarlar. Bu şekilde optimum sonuca biraz daha yaklaşılmış olur.

Şekil 3.7. de 1. Parçacığa ait çözüm hesaplanırken makine atamalarını EKIZ kuralına göre gerçekleştirmiştik. Daha sonra C_{max} değerini hesaplarken, operasyonların makinelere yerleştirilmesini Xp_1 vektöründe verildiği sırayla gerçekleştirdik. Fakat bazı durumlarda EKIZ kuralıyla belirlemiş olduğumuz makine, operasyonu en kısa sürede gerçekleştiriyor olsa bile makinenin operasyonu işlemeye başlama zamanına bağlı olarak, bulmuş olduğumuz C_{max} değerinden daha kısa sürede işleyen bir makine bulunabiliyor. Önermiş olduğumuz algoritmada C_{max} değeri hesaplanırken, operasyon Xp_1 vektöründe belirlenen makinaya atanır. Bu atama yapılırken ilgili makine üzerinde daha önceden atanmış operasyonlar var ise, belirlenen kurallara bağlı kalarak operasyonu mümkün olan ilk boşluğa atar ve bulmuş olduğu C_{max} değerini saklar. Seçilen makinenin en iyi C_{max} değerini verdiğini anlamak için ise aynı işlemleri diğer makinelerde de uygulayıp C_{max} değerlerini karşılaştırır. Karşılaştırma sonucunda daha iyi bir değer bulduğu takdirde atama işlemini ilgili makineye yaparak sonraki operasyonların atamalarına devam eder (Şekil 3.9.). Bu şekilde YA yöntemi kullanılarak 1. Parçacığa (Xp_1 vektörü) ait çözüm iyileştirmiş oluyoruz.

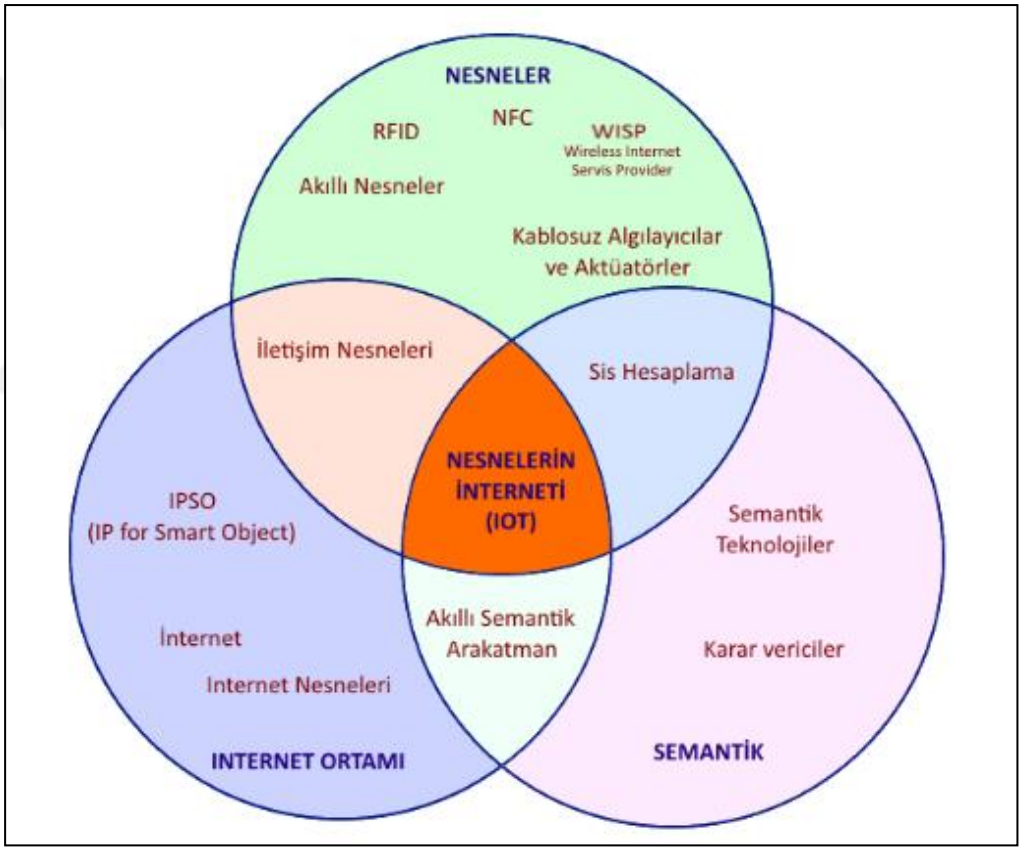
YA stratejimiz Şekil 3.8.'de verilen 1. Parçacığa ait çözüm için uygulanarak sonuç iyileştirilmeye çalışılmıştır. Oluşan yeni çözümün Gantt şeması Şekil 3.9.'da verilmiştir. Gantt şeması incelendiğinde, EKIZ kuralının uygulandığı ilk çözümde 2. Makinaya atanan O_{31}'in 3. Makinaya atanmasıyla en son biten O_{32} operasyonunu 2 birim sola kaydırılarak çözüm iyileştirilmiş oldu. İlk çözümde C_{max}=8 iken iyileştirilen çözümde C_{max}=7 bulunmuştur.



Şekil 3.9. YA stratejisinin uygulandığı 1. Parçacığa ait çözümün Güncellenmiş Gantt şeması

3.2. Nesnelerin İnterneti

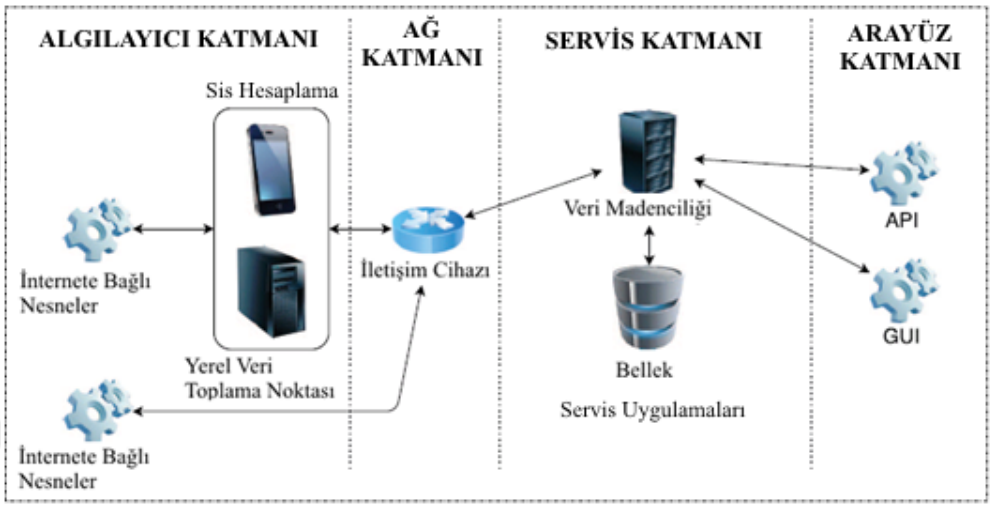
Nesnelerin interneti için "Çevremizde bulunan adreslenebilir (Mac Adresi olan), internete bağlanabilir tüm cihazların (nesnelerin) belirli iletişim protokolleri sayesinde birbirleri arasında oluşturdukları ağ yapısıdır" tanımını yapabilir. IoT kavramı tek bir teknoloji değildir daha ziyade birlikte çalışan çeşitli teknolojilerin bir araya gelmesiyle meydana gelmektedir. Çeşitli teknolojileri sınıflandıracak olursak Nesnelerin İnterneti kavramının tam olarak nereye dek geldiğini Şekil 3.10. i inceleyerek daha iyi anlayabiliriz (Atzori ve ark., 2010).



Şekil 3.10. Nesnelerin İnternetinin Teknolojiler Arasındaki Yeri

3.2.1. Nesnelerin interneti mimarisi

Çok farklı alanlarda farklı şekilde uygulamaları olmasına rağmen internete bağlı olan nesneler benzer mimariye sahiptirler. Veriler işlenir (Algılayıcı Katmanı), taşınır (Ağ Katmanı), depolanır-yorumlanır (Servis Katmanı) ve görüntülenirler (Arayüz Katmanı). Xu ve Arkadaşlarının önermiş olduğu 4 katmanlı IOT mimarisini inceleyeceğiz (Xu ve ark, 2014). Şekil 3.11.'de verilen mimariyi 4 seviyede inceleyecek olursak;



Şekil 3.11. Nesnelerin İnterneti Mimarisi

3.2.1.1. Algılayıcı katmanı

Algılayıcı katmanı, ağa bağlı nesneleri ve bu nesnelerin üretmiş olduğu verilerin toplandığı yerel veri toplama noktalarını içerir. Bu katmanda, RFID etiketleri, RFID Okuyucuları, akıllı algılayıcılar, aktüatörler, WSN (Wireless Sensör Network) vb. donanımlarla fiziksel dünyayı algılama, kontrol etme ve veri toplama işlemleri gerçekleştirilir. Nesneler kablolu veya kablosuz iletişim protokolleriyle veri toplama noktasına bağlanırlar. Veri toplama noktaları akıllı telefon, mini bilgisayar veya başka bir cihaz olabilir. Bu cihazlar bağlı nesnelere ulaşmak için bir ağ geçidi gibi kullanılırlar. İstendiği takdirde bu cihazlar üzerinden de direk olarak bağlı nesnelere ulaşılabilir ve yönetilebilirler. Veri toplama noktamızın bir Raspberry Pi Mini

Bilgisayar olduğunu düşünürsek bu mini bilgisayar üzerinde çalışan bir uygulama aracılığı ile bağlı nesnelere direk olarak ulaşılabilir ve yönetilebilir.

3.2.1.2. Ağ katmanı

Ağ katmanı, nesneler (algılayıcılar) ile verilerin işleneceği ve kaydedileceği sunucu bilgisayarlar arasında iletişim kurulmasını sağlar. Kablosuz veya kablolu ağ üzerinden temel ağ desteği ve veri aktarımı gerçekleştirilir. Bu katmanda Ağ yönetim teknolojileri, ağlardaki enerji verimliliği, QoS (Quality of System) gereksinimleri, sinyal işleme ve veri, güvenlik-gizlilik alanlarındaki konular çalışılmakta ve bu konularda araştırmalar yapılmaktadır.

3.2.1.3. Servis katmanı

Servis katmanı, kullanıcı ve sistem ihtiyaçlarını karşılamak için servislerin oluşturulmasını ve yönetilmesini sağlar. Genel olarak verilerin işlenmesi, analizlerin yapılması, ilgili verilerin çeşitli yöntemlerle saklanması ve bir üst katmanın ihtiyaçlarını karşılayacak her türlü hizmet odaklı işlemlerin gerçekleştirildiği katmandır.

3.2.1.4. Arayüz katmanı

Görüntüleme seviyesinde kullanıcılar veya bazı sistemler tarafından verilere Uygulama Programlama Arayüzleri (API) ve Grafik Kullanıcı Arayüzleri (GUI) ile ulaşılıp görüntülenir ve raporlanırlar.

Sadece Yerel Nesnelerin Ağı Seviyesi Nesnelerin İnternetine özgüdür. Diğer seviyeler diğer alanlarda da bulunurlar. Bu seviye mimariye eklendikten sonra Nesnelerin İnterneti kavramı ortaya çıkmıştır.

3.2.2. Nesnelerin interneti alanında kullanılan teknolojiler

Bu bölümde Nesnelerin İnterneti teknolojilerini 3 ana başlık altında incelemeye çalışacağız. Bunlar, "Tanımlama ve İzleme Teknolojileri", "İletişim Teknolojileri", "Servis ve Uygulama Yazılımları".

3.2.2.1. Tanımlama ve izleme teknolojileri

"Algılayıcı Katmanı" olarak adlandırdığımız katmanda bulunan teknolojileri kapsamaktadır. Bu teknolojiler aracılığı ile fiziksel dünyadan bilgili toplanmasına ve fiziksel dünyada bir eylemin gerçekleştirilmesine olanak sağlayan algılayıcılar ve aktüatörlerin tanımı ve çeşitleri üzerindeki araştırmalar paylaşılacaktır.

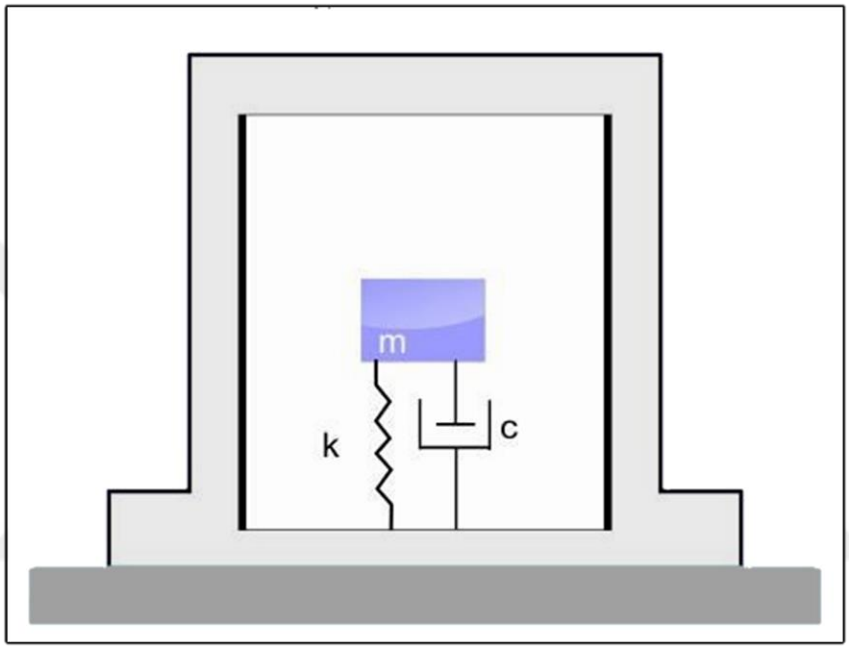
3.2.2.1.1. Algılayıcılar

Tüm Nesnelerin İnterneti uygulamaları, fiziksel dünyadan veri toplamak için kullandıkları bir veya birden fazla algılayıcı teknolojilerine sahiptirler. Akıllı nesnelerin temel bileşenleridirler. Algılayıcılar genellikle küçük ve düşük maliyetli teknolojilerdir ve az miktarda güç tüketirler. Pil kapasitesi ve taşınabilirlik özellikleri açısından sınıflandırılabilirler. Kullanılacak projelerde pil ve taşınabilirlik gibi özellikler göz önünde bulundurularak algılayıcı teknolojisi seçimi yapılmalıdır. Schmıdt ve Laerhoven, bu konuyla ilgili yayımlanmış olan makalelerinde akıllı uygulamalarda kullanılan çeşitli algılayıcı türlerine genele bir bakış açısı sunmaktadır (Schmıdt ve Laerhoven, 2001).

Özellikle akıllı telefonlar iletişim ve veri işleme özellikleri açısından oldukça kullanışlı ve kullanıcı dostu cihazlardır. Akıllı telefon kullanımının yaygın olmasından dolayı Nesnelerin İnterneti alanında çeşitli projeler gerçekleştirilmiştir (Lane ve ark, 2010; KHAN ve ark, 2013). Anlamlı sonuçlar üretebilmek için akıllı telefon üzerinde bulunan algılayıcı verilerini kullanan mobil uygulamalar geliştirilebilir. Modern bir telefonda bulanan bazı algılayıcılar aşağıdaki gibidir.

(1) İvme Ölçer, bir cep telefonunun hareketini ve hızını algılayan bir donanımdır. Genellikle akıllı telefonun hızındaki değişiklikler üç boyutta ölçülür. Birçok ivmeölçer türü bulunmaktadır (Woodford, 2019)

İvmeölçer tarafından yakalanan veri kalıpları, kullanıcının koşu, yürüyüş ve bisiklet gibi aktivitelerini algılamak için kullanılmaktadır.



Şekil 3.12. İvmeölçerin mekanik modeli

(2) Jiroskop telefonun yönlendirilmesini algılar. Yönlendirme sismik bir kütlenin belirli bir yönde hareket etmesi durumunda kapasitif değişikliklerin hesaplanmasıyla bulunur.

(3) Kamera ve Mikrofon görsel ve işitsel verilerin algılandığı çok yaygın kullanılan algılayıcılardır. Bu algılayıcılardan alınan veriler diğerlerine oranla yüksek boyutlu olduğundan, alınan verilerin direk olarak uygulamalarda kullanılması yerine anlamalı verilen kullanılması için bu verileri yorumlanarak filtrelenmesi gerekebilmektedir.

(4) Manyetometre, manyetik alanların tespitinde kullanılmaktadır. Dijital pusula uygulaması veya metallerin algılanması gibi uygulamalarda kullanılabilirler

(5) GPS (Global Positioning System), telefonun dünya üzerindeki konumunu tespit etmeye yarar. Bu konum bilgisi 3 parametreli bir gösterime sahiptir. Üç veya daha fazla uydu sisteminin kullanılmasıyla (A-GPS durumunda cep telefonu kulelerinden faydalanılır) ölçülür ve koordinatlar tespit edilir.

(6) Işık algılayıcısı, ortam ışığının yoğunluğunu hesaplar. Ekran ışığının miktarını bu algılayıcıdan gelen verilere göre ayarlayabilir. Bir odada bulunan ışıkların devreye alınmasını sağlayan uygulamalar gerçekleştirebiliriz.

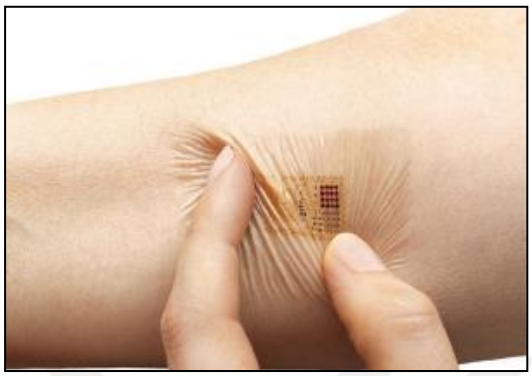
(7) Mesafeölçer, IR ışınlarını yayan kızılötesi bir lamba kullanılır. Bu ışınların cisimlere çarpıp geri dönme süreleri baz alınarak cisimle olan mesafe hesaplanabilmektedir. Mesafeye göre işlem gerçekleştirilen uygulamalar geliştirilebilir.

(8) Samsung Galaxy S4 gibi bazı akıllı telefonlarda sıcaklık, basınç ve nemi ölçmeye yarayan algılayıcılar bulunmaktadır.

Akıllı telefonlarda kullanılan algılayıcılar sayesinde Nesnelerin İnterneti alanında çeşitli uygulamalar yapma fırsatı yakalayabiliriz. Örneğin, Wang ve arkadaşları (Wang ve ark, 2014), Akıllı telefon üzerinde çalışan bir mobil uygulama geliştirmişlerdir. Öğrencilere yönelik geliştirilmiş bu uygulama; öğrencinin ne kadar uyduğunu (İvmeölçer,Işık Algılayıcı), bulunduğu ortamları (İvmeölçer, GPS ,Kamera), Sosyal hayat ve konuşmalarını (Mikrofon) telefon üzerinde bulunan algılayıcılar vasıtasıyla tespit ederek ve öğrenciye anketler yaptırarak veriler toplamaktadır.Uygulama, elde edilen veriler sayesinde öğrencinin Ruh hali, Stress düzeyi ve sosyal yaşam davranışları hakkında bilgi vermektedir.

Tıbbi algılayıcılar, Nesnelerin İnterneti sağlık uygulamalarında kullanılan, insan vücudundaki çeşitli tıbbi parametreleri ölçen ve izleyebilen algılayıcılardır. Mc Grath ve Scanaill, bir insanın sağlığını takip edebilmek için vücuda takılabilen farklı algılayıcıları yazmış oldukları makalelerinde ayrıntılı bir şekilde anlatmaktadırlar (Mc

Grath ve Scanaill, 2013). Kalp hızı, nabız, kan basıncı, solunum hızı ve kan şekerleri seviyeleri gibi farklı parametreleri ölçen tıbbi algılayıcılar piyasada rahatlıkla temin edilebilmektedir.



Şekil 3.13. Tıbbi algılayıcı içeren Geçici Dövme

Kimyasal algılayıcılar, kimyasal ve biyo-kimyasal maddeleri algılamak için kullanılırlar. Bu algılayıcıların genellikle gıda kalitesi kontrolü, akıllı şehirlerin kirlilik seviyesi ölçümü gibi uygulamaları mevcuttur (Manna ve ark, 2014).

3.2.2.1.2. Aktüatörler

Aktüatörler, elektrik enerjisini yararlı elektrik enerji türlerine dönüştürerek fiziksel dünyada değişikliklere neden olabilen cihazlardır. Isıtma ve soğutma elemanları, hoparlörler, ışıklar, monitörler aktüatörlere örnek olarak verilebilirler. Akıllı ev uygulamalarında, Kapıları kilitlemek-açmak, ışıkları veya diğer elektrikli cihazları açıp kapatmak, alarmları, bildirimleri kullanarak kullanıcıları herhangi bir tehdide karşı uyarmak ve bir evin sıcaklığını kontrol etmek için kullanılırlar.

3.2.2.2. İletişim teknolojileri

IoT nesneleri genel olarak internete TCP ve UDP gibi IP tabanlı protokoller ile bağlanırlar. Bu protokoller karmaşıktır ve bağlanan nesnelerden yüksek miktarlarda güç ve bellek talep ederler. IoT nesneleri daha az güç ve bellek tüketen IP dışı ağlar vasıtasıyla yerel olarak bağlanabilir ve bir ağ geçidi üzerinden internete

bağlanabilirler. Bluetooth, RFID, ve NFC gibi IP dışı iletişim kanalları oldukça popülerdir fakat destekledikleri mesafeler oldukça sınırlıdır. Bu yerel ağların destekledikleri mesafeleri arttırmak için IP protokollerinde değişiklik yapmak gerekmektedir. Çözümlerden biri IPv6 yı düşük güç gerektiren kısa mesafeli yerel ağlarla birleştiren 6LoWPAN protokolüdür. Bu ağların menzili yerel ağlarınkine benzer ve güç tüketimi çok daha küçüktür.

IEEE 802.15.4, LoraWAN, Sigfox, 6LoWPAN, RFID, NFC ve benzeri protokoller IoT dünyasında düşük güç kullanan popüler teknolojilerdir.

3.2.2.2.1. TCP

TCP (Transmission Control Protocol.), internet ortamında en yaygın olarak kullanılan ip tabanlı bir iletişim protokolüdür. TCP ile veri gönderimi yapılırken gönderici ve alıcı bilgisayarlar arasında anlaşmalı bir iletim söz konusudur. Gönderici verinin sıralı ve eksiksiz gönderimini garanti eder. Alıcı her bir veri paketini doğru ve eksiksiz aldığını göndericiye geri bildirimde bulunarak bildirir. Gönderici gönderdiği veri paketinin doğru bir şekilde ulaştığına dair alıcıdan mesaj alana kadar aynı paketi göndermeye devam edecektir. Belirli bir süre içerisinde alıcıdan herhangi bir mesaj alınamadığında ise gönderici, bu gönderimi bağlantı hatası koduyla iptal edecektir. Bu nedenle TCP ile hatasız ve doğru bir şekilde veri gönderilebilmektedir.

3.2.2.2.2. UDP

UDP (User Datagram Protocol), TCP ile benzer bir şekilde çalışır. TCP den farklı olarak veri gönderimi yapılırken gönderici ve alıcı bilgisayarlar arasında anlaşmalı bir iletim söz konusu değildir. Gönderici göndermek istediği veriyi sadece göndermeyi amaçlar. Bir veri paketi gönderdiğinde alıcıdan paketin doğru ve eksiksiz gönderildiğine dair bir geri bildirim beklemeden sıradaki veriyi gönderir. Bu nedenle hatalı bir gönderim yaşandığında alıcının bu gönderimi hatalı gönderildiğine dair alıcıyı bilgilendirmesi mümkün değildir. UDP ‘de TCP de olduğu gibi bir kontrol mekanizması ve anlaşma söz konusu olmadığından iletim daha hızlı

gerçekleştirilmektedir. İletimin daha hızlı bir şekilde yapılması gerekliği, iletim hatalarının önemsiz olduğu ve broadcast (canlı yayın) gibi amaçların güdüldüğü durumlarda UDP kullanmak daha avantajlı hale gelmektedir.

Sanayi ortamında üretim yapan cihazlar üzerinde genel olarak yüksek voltajda elektrik enerjisi bunduğundan, kullanacağımız iletişim protokolünün seçiminde enerji tüketiminden çok gerçekleştirilen işe uygunluk ve verimlilik göz önünde bulundurulmuştur. Bu nedenle broadcasting yayın yaparak tüm cihazlara aynı anda senkronize bir şekilde veri gönderebilmemizi sağlayan UDP protokolü kullanılmıştır.

3.2.2.2.3. RFID

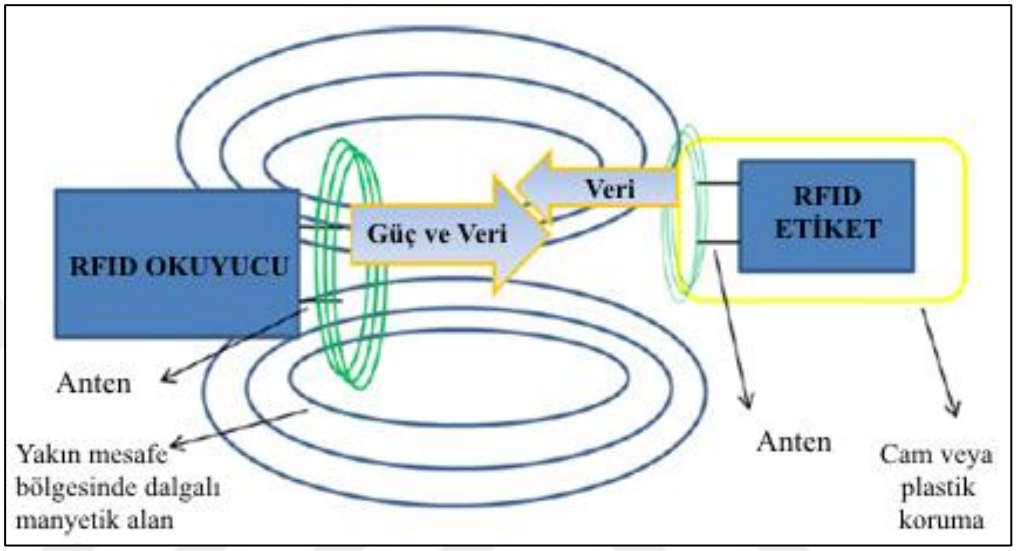
RFID (Radio Frequency Identification) teknolojileri IoT konusunun en önemli teknolojilerinden biridir ve RFID etiketlerinin iletişim teknolojisine entegrasyonu bu konunun temelini oluşturmaktadır. Barkod teknolojisi ile benzedir fakat borkodun aksine etiket ve okuyucunun birbirlerini görmesi gerekli değildir. Mesafe frekansa göre değişiklik gösterebilir. RFID iki bileşenden meydana gelmektedir, bunlar; RFID Okuyucular ve RFID etiketler.

Çizelge 3.2. RFID Sistemlerde kullanılan farklı etiketler

Etiket / Parametre	Pasif	Yarı-Pasif	Aktif
Yakınlık Bilgisi	İyi	Zayıf	Zayıf
Okuma Mesafesi	3 m kadar	30 m kadar	30 m kadar
Güç Kaynağı	İndüksiyon	Pil ve İndüksiyon	Pil
Maliyet/Etiket	25 cent	Geliştirlmekte	2$ - 100$
Depolanan bilgi miktarı	2k (Sadece Okuma)	32K veya daha fazla (Okuma/Yazma)	32K veya daha fazla (Okuma/Yazma)
Frekans Çatışması	Orta	Yüksek	Yüksek

RFID Etiketler nesnelere yerleştirilmiş ve bilgileri saklayan bileşenlerdir. İçerisinde anteni olan bir mikroçip barındırırlar. Okuyucular ise etiket üzerinde saklanan veriyi okumaya yararlar. Kullanılan güç kaynağına göre üçe ayrılırlar, aktif

etiketler, pasif etiketler ve yarı pasif etiketler. Bu etiketlerin detaylı özelliklerini Çizelge 3.2.'ye bakarak inceleyebilirsiniz. Etiket ve anten arasındaki iletişim üzerlerinde bulunan antenler tarafından gerçekleştirilir. Ekstra anten kullanılarak okuma mesafesini arttırmak mümkündür. RFID temel çalışma prensibi aşağıdaki şekilde gösterilmektedir.



Şekil 3.14. Etiketlerin güç ve iletişim mekanizması (100 Mhz'den daha düşük frekanslarda)

3.2.2.2.4. Near Field Communication (NFC)

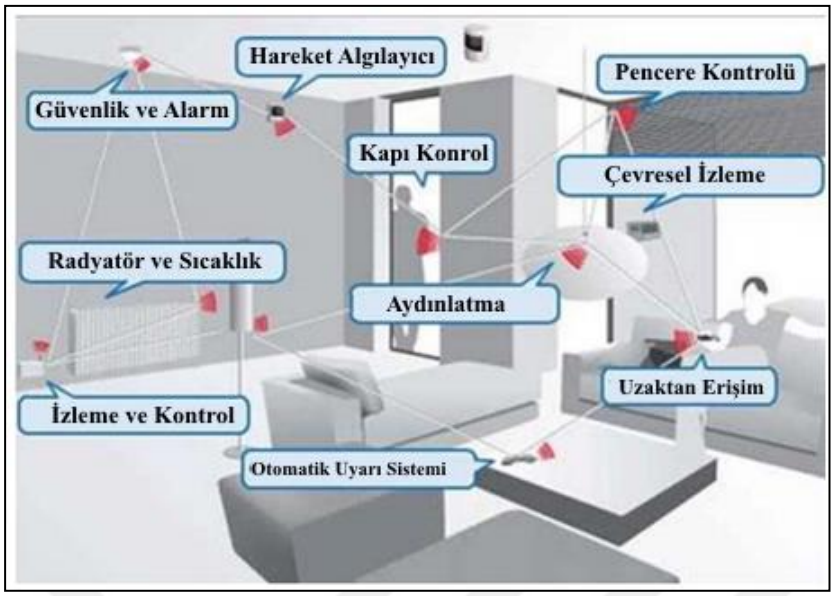
Kısa menzilli kablosuz iletişim teknolojisidir. İki nesne arasında her türlü veri birbirlerine yakınlaştırılarak saniyeler içerisinde aktarılabilir. Bu teknoloji RFID teknolojisi mantığı üzerine kuruludur. İki NFC özellikli nesne arasındaki veri iletişiminde manyetik alan değişiklikleri kullanılır. NFC, RFID ile aynı olan 13.56 Mhz'lik bir frekans bandında çalışır. Aktif ve Pasif olmak üzere iki çalışma durumu vardır. Aktif durumda iki nesne de manyetik alan oluştururken Pasif durumda nesnelerden sadece bir tanesi manyetik alan oluşturur. Pasif durum mobil cihazlarda pil kullanımını optimize etmek için düşünülmektedir. NFC, RFID den farklı olarak iki yönlü iletişim olanağı sağlamaktadır. Ödeme işlemleri gibi yüksek güvenlik gerektiren işlemlerde yakın mesafeden işlem yapılabilmesi bir avantaj olarak düşünülebilir (Agrawal ve ark, 2012;Curran ve ark, 2012),

3.2.2.2.5. Wireless Sensor Network (WSN)

Geniş alanların ve karmaşık etkinliklerin izlenmesinde birden çok kablosuz algılayıcının kullanılması gerekir ve bu algılayıcılarında birbirleriyle kablosuz iletişim kurmaları gerekebilir. RFID, NFC ve Bluetooth gibi IP dışı teknolojilerin, destekledikleri mesafelerin düşük olmasından dolayı geniş alanlara yayılmış birden fazla algılayıcının iletişiminde kullanılması tercih edilmemektedir. WSN kablosuz teknolojileri kullanarak binlerce algılayıcının bulunduğu bir alanda iletişimi sağlayabilir. Algılayıcılardan gelen veriler bir ağ geçidi üzerinden internete aktarılır. WSN de iletişimin çoğu IEEE 802.15.4 standardına göre gerçekleştirilir.

3.2.2.2.6. Zigbee

Kablosuz haberleşmede kullanılan Wi-Fi, Bluetooth, WIMAX ve 3G-4G gibi teknolojiler bilindiği üzere fotoğraf, video, ses gibi büyük boyutlu verilerin hızlı bir şekilde iletilmesini sağlayan teknolojilerdir. Bu teknolojileri kullanarak çalışan cihazlar iletim hızıyla doğru orantılı olarak sistemin kaynaklarını kullanırlar. Algılayıcı ve kontrol cihazları çok büyük boyutlarda veri taşımadıklarından dolayı bu teknolojileri bu alanda kullanmak sistem kaynaklarının kullanımı açısından dezavantaja sahiptirler. Bu durumu göz önünde bulunduran bilim adamları bu tür cihazların veri iletişimlerine kullanmak üzere sistem kaynaklarını çok daha az kullanan teknolojilerin geliştirilmesi için çalışmalar yürütmektedirler. IEEE, 802.15.4. standardı özelliklerini temel alarak Zigbee protokolü ve bu protokol ile çalışan cihazlar üretilmiştir (Şekil 3.15.). Artık günlük hayatımızda Zigbee destekli cihazların kullanımı yaygınlaşmıştır. Zigbee, düşük enerji tüketen ve düşük maliyetli bir teknoloji olması nedeniyle çeşitli sektörlerde, veri iletiminde, kontrol ve izleme uygulamalarında çok yaygın kullanılmaktadır. Zigbee teknolojisi Akıllı Ev, Güvenlik, Sağlık sektörü, otomotiv, tarım ve endüstri uygulamalarında yaygın bir şekilde kullanılmaktadır.



Şekil 3.15. Zigbee'nin bina otomasyonunda kullanımı

Çizelge 3.3. de Zigbee ile diğer popüler olan protokollerin karşılaştırılması gösterilmektedir.

Çizelge 3.3. Zigbee teknolojisinin farklı kablosuz iletişim teknolojileriyle karşılaştırılması

	Zigbee	**Bluetooth**	**Wi-fi**	**GPRS/GSM**
Odaklanma Alanı	İzleme ve Kontrol	Kablo Yerine	Görüntü – Eposta- Web,	Geniş alanda veri ve ses
Pilin Ömrü (Gün)	100 – 1000	1 - 7	0,5 - 5	1 – 7
Ağın Boyutu (Adet)	2^{64} Sınırsız	7	32	1
Sistem Kaynağı	4 - 32 Kb	250 Kb	1 Mb	16 Mb
Ağın Veri Genişliği (kb/s)	20 – 250	720	11000+	64-128+
Başarılı olduğu Alanlar	Maliyet, Dayanıklı olması ve Güç Tüketimi	Maliyet, Rahatlık	Hızı ve Esnekği	Ulaşılabilirlik, Kalite
Kapsama Alanı (metre)	1-100+	1-10+	1-100	1000+

Uzun pil ömrüne sahip olması, esnek bir ağ yapısına sahip olması, sistem kaynaklarını minimum seviye kullanması ve kullandığı frekans bandının lisans gerektirmemesi sebebiyle kontrol ve izleme amaçlı uygulamalar için en uygun teknolojilerden bir tanesidir. Dezavantajı ise büyük diğer teknolojilerdeki gibi yüksek boyutlu veri akışı sağlayamamasıdır.

3.2.2.3. Servis ve uygulama yazılımları

Çevremizde bulunan ve internete bağlanabilir nesneler farklı teknolojilerle üretilmiş olabilirler. Bu yüzden bu nesnelerin birlikte çalışabilmeleri için çok iyi tanımlanmış standartlara ihtiyaç vardır. Fakat standartlaşma farklı uygulamaların ve cihazların farklı gereksinimleri nedeniyle zorlaşmaktadır. Bu türden heterojen uygulamaların ayrıntılarını soyutlayacak farklılıklarının önemsenmeyeceği ortak bir ara katman platformu yani servis sağlayıcıların bulunması gerekmektedir. Nesneler ve uygulamalar arasında bir yazılım köprüsü oluşturacak, uygulamaların temel donanım ile iletişime geçmesi yerine uygulama gereksinimlerine daha fazla odaklanarak uygulama geliştiricilerine gerekli hizmetleri sağlayacak servis yazılımlarına ihtiyaç vardır. Bir servis sağlayıcının oluşturulmasında dikkat edilmesi gerek hususlar;

(1) Birlikte çalışabilirlik; heterojen cihazlar arasında işbirliği ve veri alışverişini kolaylaştırmak için servis yazılımlarının bu ihtiyaçları karşılaması gerekmektedir.

(2) Aygıt keşfi ve yönetimi; bu özellik cihazların çevredeki diğer tüm cihazların ve onlardan sağlanan hizmetlerin farkında olmasını sağlar. Nesnelerin İnterneti cihazları var olduklarını ve ne tür hizmetler sağladıklarını belirtmeleri gereklidir.

(3) Ölçeklenebilirlik; IoT sistem kurulumlarında çok fazla sayıda cihaz kurulması gerekebilir ve artan gereksinimler nedeniyle yeni cihazlar eklenebilir. Bu nedenle alt yapı ölçeklenebilir olmalı ve servis yazılımları tarafında yönetilebilmelidirler.

(4) Büyük Veri; IoT algılayıcıları tipik olarak büyük miktarlarda veri toplamaktadırlar. Bütün bu verilerin detaylı bir şekilde analiz edilmesi ve anlamlı sonuçlar çıkarılmasını sağlayacak servis yazılımları bulundurulması gerekmektedir.

(5) IoT genellikle bir kişinin kişisel yaşamı veya bir endüstri ile ilgili uygulamaları içerir. Tüm bu uygulamalarda güvenlik ve gizlilik konularına dikkat edilmesi gerekmektedir. Servis yazılımlarının kullanıcı kimlik doğrulaması ve erişim

denetimi kontrolleri ile birlikte diğer güvenlik ve gizlilik problemlerini çözmek için mekanizmalara sahip olması gereklidir.

(6) Bulut Hizmetleri; bulut IoT'nin önemli bir parçasıdır. Algılayıcı verilerinin çoğu analiz edilir ve sonrasında bulutta saklanır. Servis yazılımlarının sorunsuzca farklı bulut tipleri üzerinde çalışabilir olmalıdır ve bu bulutlardan elde ettikleri diğer verileri göz önünde bulundurarak daha iyi sonuçlar elde etmelerini sağlayacak yetenekleri olmalıdır.

3.2.3. Nesnelerin interneti uygulama alanları

Çeşitli alanlarda geliştirilmiş IoT uygulamaları mevcuttur. Bunların büyük çoğunluğu yaşam kalitesini arttırmaya yönelik geliştirilmiş uygulamalardan oluşmaktadır. Akıllı Ev, Sağlık, çevre koruma, akıllı şehirler, otomotiv, enerji, imalat, ulaştırma vb. alanlarda uygulama örneklerine rastlanmaktadır.

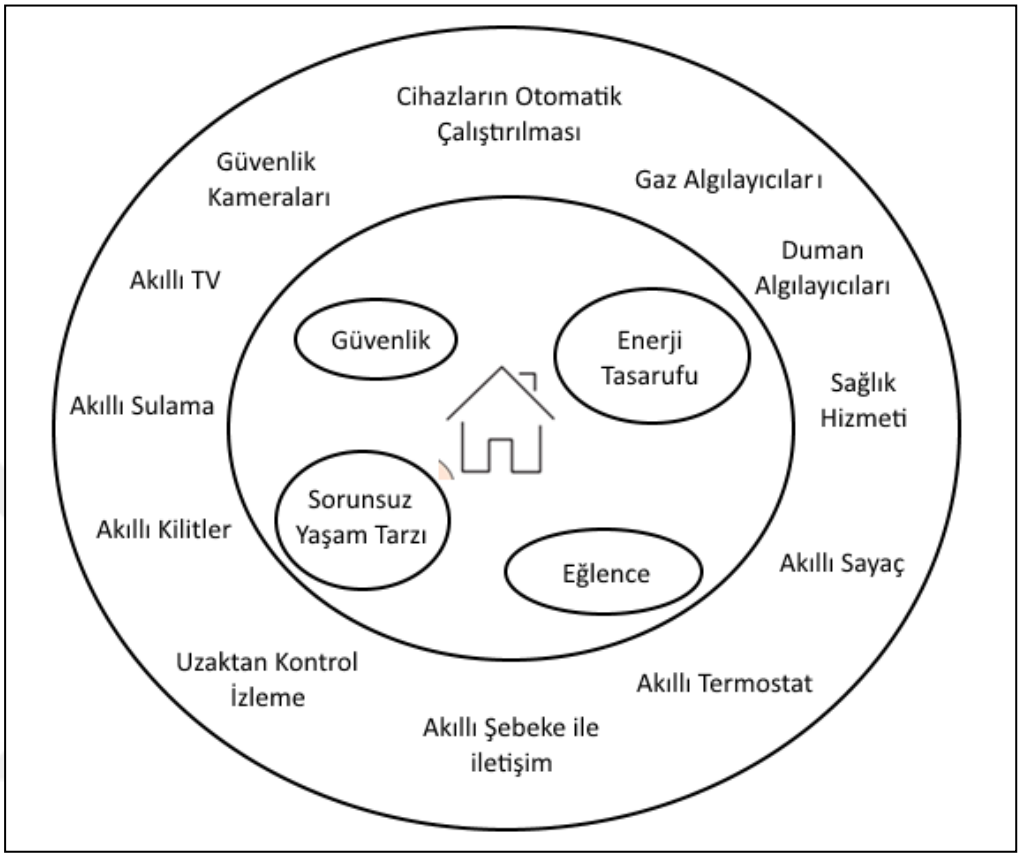
3.2.3.1. Akıllı ev

WSN ve algılayıcı teknolojilerinin hızlı gelişim göstermesi ve insanların yaşam kalitelerini arttırmak için günümüz teknolojilerine güvenmeleri, akıllı ev uygulamalarının son yıllarda daha popüler hale gelmesine neden olmuştur.

Akıllı evlerde yapılan işlerin otomatik hale getirilmesi ve unutkanlık gibi nedenlerle aksayan rutin işlerin devamlılığının sağlanması amacıyla çeşitli algılayıcı ve aktüatörlerin kullanıldığı uygulamalar geliştirilmektedir. Şekil 3.16. da Akıllı evler için bir blok diyagram gösterilmektedir (Sethi ve Sarangi, 2017).

Enerji tasarrufu için geliştirilen uygulamalarda algılayıcılar ve algılayıcılardan gelen verileri işleyen akıllı cihazlar kullanılır (Schmıdt ve Laerhoven, 2001). Algılayıcılar vasıtasıyla ortamdan toplanan veriler (Işık, sıcaklık, nem, gaz vb.) bu verileri işleyip yorumlayan akıllı cihaza iletilirler. Akıllı cihaz gelen verilere göre bir aktüatörü devreye sokar veya kullanıcıyı bilgilendirir. Algılayıcılar bir gaz sızıntısı

tespit ettiklerinde gaz akışını durdurmak için gaz vanası otomatik olarak kapatıp kullanıcıya alarm verebilir. Sıcaklık değişimine göre ısıtıcıyı çalıştırıp durdurabilir.



Şekil 3.16. Akıllı Evler için Blok Diyagram

Özellikle yaşlı insanların yaşadıkları evlerde anormal durumların (Yaşlı insanları düşmesi, hareketsiz kalması vb.) tespiti için uygulamalar geliştirilmiştir. Yu ve arkadaşları, insan vücudunun duruşlarını analiz etmek ve bu analiz sonucunda işlemler gerçekleştirilmesi için bilgisayar tabanlı teknikler kullanmışlardır (Yu ve ark, 2012). Sixsmith ve arkadaşları bir nesnenin konumunu, boyutunu ve hızını belirlemek için düşük maliyetli kızılötesi algılayıcıları kullanarak bilgi veren uygulamalar gerçekleştirmişlerdir. Bu yöntemler ve yapay zekâ teknikleri kullanılarak kişilerin günlük hareketleri kaydedilip önceki etkinlikleriyle karşılaştırıldığında hayati önem taşıyan veriler elde edilip ilgili yerlere iletilebilir (Sixsmith ve Johnson, 2004).

Akıllı ev uygulamalarında diğer IoT uygulamalarında olduğu gibi en önemli sorun güvenlik ve gizlilik konusudur (Robles ve ark, 2010). Evdeki olaylarla ilgili tüm

işlemler kaydedildiğinden ve yapılacak bir dizi işlem bu verilere bağlı gerçekleştirildiğinden, Sistemin güvenliği garanti edilemezse bir saldırgan sistemi ele geçirebilir ve sistemin kötü sonuçlar doğuracak şekilde çalışmasını sağlayabilir. Mahremiyet içeren verilerin farklı insanların eline geçmesi istenmeyen durumların meydana gelmesini ve bu sistemlere karşı güvensizlik duygusunun gelişmesine sebep olabilir. Akıllı ev sistemlerinin bu gibi anormallikleri tespit etmesi ve kullanıcıları bilgilendirmesi için araştırmacılar çalışma yapmaya başlamışlardır.

3.2.3.2. Akıllı ulaşım

Akıllı ulaşım uygulamaları algılayıcılar ve **sis** hesaplama yöntemi aracılığı ile şehir trafiğini yönetirler. Bu sistemler, trafik tıkanıklığını en aza indirmek, park sorunlarını çözmek ve kural ihlallerini tespit etmek amacıyla kullanılabilirler. Hız tespiti için ivmeölçer, araç tanıma için RFID, yolcu ve taşıtları saymak için kızılötesi algılayıcılar ve kameralar gibi pek çok teknoloji bu alanda kullanılmaktadır. Bu alanda gerçekleştirilmiş pek çok uygulama türü bulunmaktadır (Dimitrakopoulos, 2011).

3.2.3.3. Akıllı su sistemleri

Dünyanın pek çok yerinde su tüketimi önemli bir problemdir. Su kaynakları etkili bir şekilde yönetilemediğinde su kıtlığı yaşanabilmektedir. Bu yüzden su tedarik hatları ve kanalizasyonlara akıllı algılayıcılar yerleştirerek su kaynaklarının daha verimli kullanılması ve tüketimin takip edilmesi sağlanabilmektedir. Hauber-Davidson ve E. Idris, Akıllı su sayaçları için çeşitli tasarımları tarif etmektedirler. Bu sayaçlar su giriş ve çıkış miktarlarını ölçüp muhtemel su sızıntılarını tespit etmek ve verimli analizler yapmak için veri toplamaktadırlar (Hauber-Davidson ve Idris, 2006).

.

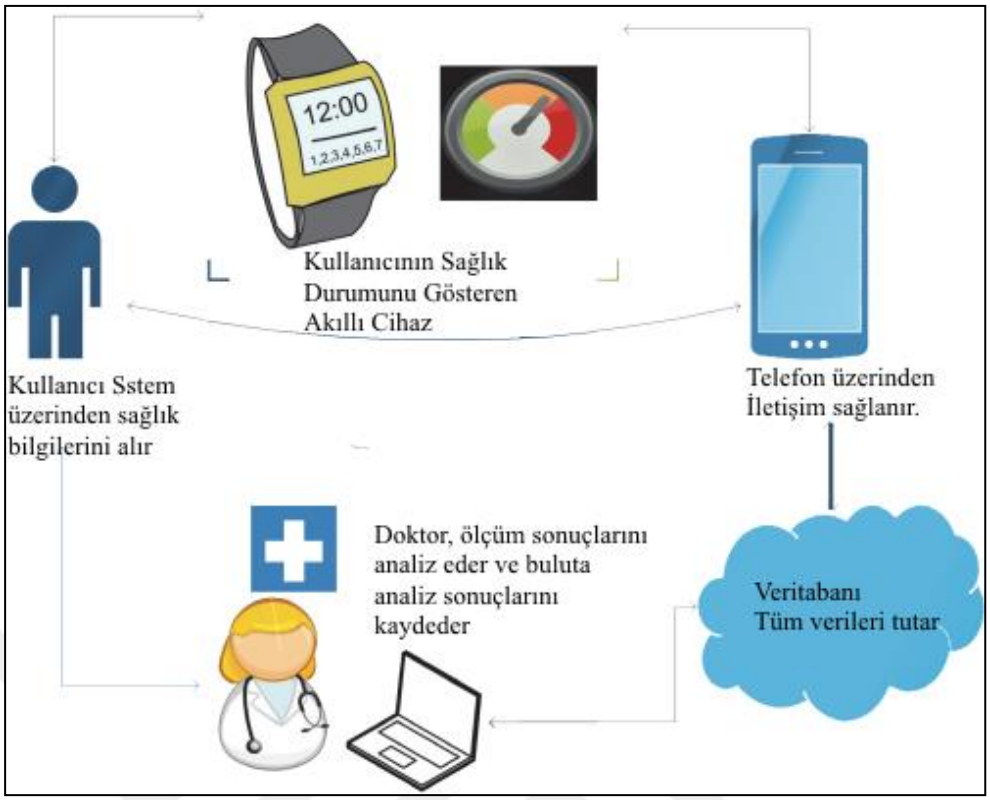
3.2.3.4. Sosyal yaşam ve eğlence

Sosyal yaşam ve eğlence, insanın hayatında çok önemli bir role sahiptir. İnsanların sosyal faaliyetlerini takip eden birçok uygulama geliştirilmiştir. Bu uygulamalardan bir tanesi çeşitli algılayıcı verilerinin yardımıyla bireyin sosyal

faaliyetlerini takip eden "Circle Sense" adlı uygulamadır (Liang, G. ve ark, 2013). Bu uygulama, sosyal faaliyet kalıplarını ve sosyal faaliyetlerde bulunan insanları analiz ederek kişinin sosyal çevresini tanımlar. Bu doğrultuda çeşitli sosyal etkinliklere katılan insanları guruplar. GPS algılayıcı kullanarak kişinin bulunduğu yeri, Bluetooth kullanarak ise çevresindeki benzer faaliyette bulunan insanları tespit eder. Bu sistem makine öğrenme algoritmaları kullanarak sürekli kendini geliştirmektedir. Logmusic uygulaması ise sıcaklık, hava durumu, zaman ve mekân gibi kavramlara göre müzik öneren bir eğlence uygulamasıdır (Lee ve Cho, 2011).

3.2.3.5. Sağlık ve spor

Bireyin sağlık durumunu izlemek için birçok giyilebilir IoT cihaz geliştirilmiştir. Bu şekilde tedavi ve yaşam şekli hakkında daha gerçekçi ve olumlu sonuçlar alınabilmektedir. Bu durum IoT 'nin Sağlık ve Spor alanında ne kadar faydalı ve önemli bir kavram olduğunu göstermektedir. Bu alanda da çeşitli uygulamalar geliştirilmektedir. Şekil 3.17. de gösterildiği gibi özellikle yalnız yaşayan yaşlı insanların sağlık aktivitelerini (tansiyon, şeker vs.) bu alanda geliştirilmiş çeşitli algılayıcılar vasıtası ile sürekli takip ederek anormal bir durum tespit edildiğinde ilgili sağlık kuruluşlarına bildiren sistemler mevcuttur (Sethi ve Sarangi, 2017).



Şekil 3.17. Akıllı Sağlık Sistemi için Blok Diyagram

3.2.3.6. Tarım

Sıcaklık ve nem parametrelerinin ölçüm sonuçları tarımsal üretim alanlarında için önem arz etmektedir. Halı hazırda bazı uygulama örneklerinde bu parametrelerin ölçümü için çiftçiler tarlalarda belirli noktalara ortam şartlarına uygun algılayıcılar koyarak veri toplamakta ve bu verileriler üzerinden analiz gerçekleştirilerek sulama, ilaçlama gibi aktiviteleri daha verimli bir şekilde gerçekleştirmektedirler. Biz bu verileri ortak bir platformda IoT ağında paylaşırsak çok daha büyük veriler oluşacak ve tarımsal üretim sonrasında ürünlerin pazar oluşumu daha verimli bir şekil alacaktır.

Günümüzde IoT teknolojileri tarım alanında en çok seralarda kullanılmaktadır. Sıcaklık, nem, toprak durumu gibi parametreler gerçek zamanlı ölçülür ve sonuçlar analiz edilmek üzere bir sunucu üzerinde kaydedilirler (Zhao ve ark, 2010).

3.2.3.7. Tedarik zinciri ve lojistik

IoT teknolojileri iş süreçlerini basitleştirmeye çalışmaktadır (Ferreira ve ark, 2010). Tedarik zincirindeki ürünler, üretim yerlerinden son dağıtım yerlerine kadar RFID ve NFC gibi algılayıcı teknolojilerini kullanarak izlenebilir. Gerçek zamanlı toplanan veriler analiz edilerek sonraki işlerin daha verimli ve performanslı gerçekleştirilmesi için kullanılabilmektedir. Bo ve Guangwen, bir tedarik zinciri yönetimini gerçek zamanlı gerçekleştirmek için bir uygulama geliştirmişlerdir. RFID etiketleri kullanılarak geliştirilen uygulamada etiketler ürünü otomatik olarak tanımlar ve konum bilgisi ile birlikte IoT ağında paylaşılır. Bu sistemle geçmişteki talepler incelenerek gelecek talepler üzerine çok verimli analizler gerçekleştirilebilir (Bo ve Guangwen, 2009).

3.2.3.8. Akıllı şebeke sistemleri

Akıllı Şebeke, modern elektrik üretimini, iletimini, dağıtımını ve tüketim sistemlerini IoT teknolojilerini kullanarak etkin bir şekilde yönetmemizi sağlayan teknolojidir (Karnouskos, 2010). Elektrik enerjisi üretimi, iletimi ve dağıtımı akıllı hale getirmek için akıllı şebeke sistemi her adımda bil biriktirir ve iki yönlü güç akışına da olanak sağlar. Bu durum ciddi enerji tasarrufları sağladığı gibi tüketici için dinamik faturalandırma gibi olanaklarıyla daha verimli bir enerji kullanması için kolaylık sağlar. Enerji hattında her olayı izlemek için algılayıcılar yerleştirilir.

İki yönlü güç akışı özellikle güneş enerjisi rüzgâr enerjisi gibi teknolojileri kullanarak enerji üreten tüketiciler için bir kolaylık sağlayacak ve artı güç geri gönderilerek faturaya yansıtılabilecektir. Bu sayede artı güç boşa harcanmış olmayacaktır.

Akıllı sayaçlar normal ve en yoğun tüketim zamanlarında alınan verileri analiz eder ve tüketim kalıplarını sunucu üzerinde IoT ağına dâhil eder. Sonraki üretimler elde edilen bu tüketim kalıplarına göre ayarlanır.

3.2.4. Geliştirmeye açık alanlar

IoT teknolojileri ve uygulamaları hala gelişim aşamasındadır. Yönetim, Standardizasyon, Güvenlik ve Mahremiyet gibi alanlarda birçok geliştirilmeye açık alanda çözülmesi gereken konular bulunmaktadır. Bu konuların çözülmesi ve IoT cihazlarının verimli bir şekilde kullanılmasını sağlamak için mevcut sistemleri ve yapılabilecek çalışmalar hakkında çok ciddi çalışmalar yapılması gerekmektedir (Xu, 2011;Atzori ve ark, 2010).

3.2.4.1. Yönetim eksikliği

Yönetim eksikliği bağlığı altında yapılması gerekleri sıralayacak olursak;

(1) IoT cihazlarının birlikte çalışabilirliğini sağlamak için servis yazılımlarını tasarlamak servis temelli nesne performansı ve maliyet kısıtlamaları açısından yakın gelecekte çözülmesi gereken önemli başlıklar arasındadır. Ağa bağlı olan cihazların sayısı arttıkça ölçeklenebilirlik, veri aktarımı, veri işleme ve yönetimi servis yazılımlarının performansı açısından farklı seviyelerde sorunlar çıkaracaktır (Miorandi ve ark, 2012).

(2) IoT içerisinde çok farklı teknolojileri ve standartları barındıran, çeşitli ağ türlerinin birleşmesinden oluşmuş heterojen bir ağdır. Bu heterojenliği gizlemek için çeşitli uygulamalara adlandırma vb. hizmetler sunan yaygın, kabul gören ortak bir platform eksikliği vardır. Faklı ağ türlerinin, mimari ve protokol düzeyinde birlikte çalışabilirliklerini yönetmek önemli bir araştırma konusudur (Miorandi ve ark, 2012).

(3) Yaygın kabul görmüş bir servis tanımlama dili eksikliği servis geliştirmeyi ve fiziksel nesnelerin kaynaklarının diğer servislerle entegrasyonunu zorlaştırmaktadır. Geliştirilen servis yazılımları farklı iletişim ve uygulama ortamlarıyla uyumlu olmayabilir. Bu nedenle IoT teknolojisinin daha etkin kullanımı için güçlü servis bulma yöntemleri ve nesne adlandırma servisleri geliştirilmelidir (Perera ve ark, 2014)

(4) IoT mevcut internet alt yapısına bağlı olarak geliştirildiğinden internete bağlı olan her şeyden etkilenir. IoT cihazlarını bu sistemlere ve bu bilgi altyapısına entegre etmek için çok ciddi çalışmalar yapılması gerekmektedir. Dahası internete bağlı olan çok sayıda cihaz büyük miktarda gerçek zamanlı veri akışlarına sebep olacaklardır. İnsanların bu verilerin anlamlandırmak için etkili yöntemler geliştirmeleri gerekmektedir aksi takdirde bu verilerin bir anlamı olmayacaktır. Mevcut yazılım sistemleri ve web hizmetlerinin IoT cihazlarıyla entegrasyonu için yeni uygulamalar geliştirilmelidir.

3.2.4.2. Standardizasyon

IoT teknolojisinin hızlı bir şekilde büyümesi standartlaşmayı bir hayli zorlaştırmaktadır. Aynı zamanda standardizasyon IoT teknolojilerinin gelişmesi yaygınlaşması için önemli bir oynamaktadır. IoT standardizasyonu, yeni kullanıcılar, hizmet sağlayıcıların sisteme kolayca entegre olmasını, farklı uygulamaların-sistemlerin birlikte çalışabilirliğini iyileştirmeyi, ürünlerin ve hizmetlerin daha performanslı üretilmesini amaçlamaktadır. Farklı ülkelerdeki cihazların ve uygulamaların bilgi alışverişinde bulunabilmelerini sağlamak için çok iyi bir standardizasyon çalışması yapılması gerekmektedir (Miorandi ve ark, 2012). IoT teknolojisinde yaygın kullanılan çeşitli standartlar IoT teknolojisinin yaygınlaşmasında önemli rol oynayacaktır.

Birçok ülke ve kuruluş IoT standartlarının geliştirilmesi konusunda çalışmalar yapmaktadır. Uluslararası Tele-iletişim Birliği, Uluslararası Elektroteknik Komisyonu, Uluslararası Standardizasyon Örgütü, IEEE, Avrupa Elektroteknik Standardizasyon Komitesi, Çin Elektronik Standardizasyon Enstitüsü ve Amerikan Ulusal Standartlar Enstitüsü gibi sayısız organizasyon bu alanda ciddi çalışmalar yürütmektedir.

3.2.4.3. Güvenlik ve gizlilik

Yeni IoT teknolojilerinin ve hizmetlerinin kabul görmesi ve yaygınlaşması, kurulum ve karmaşıklık nedeniyle Güvenlik ve Gizlilik konularında tutarlı yapılan çalışmalara bağlıdır (Wan ve Jones, 2013). Mevcut birçok teknoloji tüketiciler tarafından kullanılmaktadır, ancak yüksek güvenlik ve gizlilik gerektiren endüstri uygulamaları içim uygun değildir. Güvenliği sağlamak için WSN veya diğer teknolojilerden alınan mevcut şifreleme teknolojisi IoT içerinde kullanıldığında dikkatlice gözden geçirilmelidir. IoT birçok günlük aktivitenin izlenmesine ve bağlanmasına izin verdiğinden birçok özel bilgi otomatik olarak toplanabilir. IoT varlıklarının üzerindeki saldırı vektörlerinin sayısı mevcut sistemlerinkinden çok daha fazla olduğundan gizliliğin korunması daha da önemli hale gelmektedir. Örneğin, bir sağlık uygulaması kalp hızı ve kalp şekeri gibi hastanın özel bilgilerini toplar ve ağ üzerinden doktorla paylaşılır. Bu bilgiler ağ üzerinden aktarıldığında hasta bilgileri çalınabilir. Ben zer bir durum endüstride olması durumunda çok ciddi zararlar meydana gelebilir. Mevcut ağ güvenlik teknolojileri, IoT de güvenliği ve gizliliği için bir temel oluştursa da bunun yeterli olmayacağı açık bir şekilde görülmektedir. IoT için güvenilir bir güvenlik mekanizması aşağıdaki yönlerden araştırılmalıdır.

(1) Güvenlik ve gizlilik kavramlarının sosyal ve kültürel açıdan tanımlanması

(2) Güven ve İtibar mekanizmalarının oluşturulması

(3) Uçtan uca şifreleme gibi iletişim güvenlik mekanizmalarının IoT için tekrardan tasarlanması

(4) İletişim Gizliliği ve Kullanıcı verileri

(5) Hizmetler ve Uygulama üzerindeki güvenlik

3.2.5. Araştırma konuları

IoT teknolojilerinin gelişimi büyük olasılıkla aşamalı bir şekilde ilerleyecektir. IoT teknolojilerindeki eksiklikleri ve zorlukları aşmak için uluslararası bir işbirliklerine ihtiyaç olduğu görülmektedir. Bu bölümde mevcut sorunların eksikliklerin yanı sıra farklı araştırma konularını ele alacağız.

3.2.5.1. Sosyal ağ entegrasyonu

IoT çözümleri ve sosyal ağın entegrasyonu ile farkı IoT nesneleri arasındaki iletişimi arttırmak için yenilikçi yaklaşımlar mevcuttur. Atzori ve arkadaşları tarafından yakın zamanda Sosyal Nesnelerin İnterneti (SIoT- Social Internet of Things) adlı yeni bir paradigma önerildi. Nesneler arasındaki ilişkileri insanlar arasındaki sosyal ağ (Facebook,twitter vb.) ilişkisine benzeterek bir sosyal ağ oluşturulmasını öngörmektedirler (Atzori ve ark, 2012).

3.2.5.2. Yeşil IoT (Nesnelerin İnterneti) teknolojileri

Yeşil IoT teknolojilerinin geliştirilmesiyle kablosuz ağ üzerinden iletişim kuran milyarlarca algılayıcının düşük güç tüketmesi amaçlanmaktadır. Algılayıcılar tarafından tüketilen enerjiyi minimum düzeye getirecek teknikler veya yaklaşımların geliştirilmesi gerekmektedir (Yaacoub ve ark, 2012).

3.2.5.3. Bağlam farkındalığı

Bağlam Farkındalığı, milyarlarca algılayıcı internete bağlandığında insanların bu nesnelerden gelen tüm verileri işleyebilmesi çok mantıklı olmadığından önemli araştırma konularından bir tanesi durumundadır. Algılayıcı verilerini daha iyi anlamak ve hangi verilerin işlenmesi gerektiğine karar veren hesaplama teknikleri önerilmektedir (Wang ve ark, 2013). Günümüzde çoğu IoT çözümleri bağlam farkındalığı yeteneklerine sahip değiller. Avrupa birliği bağlam farkındalığını önemli bir IoT araştırma alanı olarak belirlemiştir.

3.2.5.4. Yapay zekâ tekniklerinin kullanımı

Yapay Zekâ Tekniklerinin IoT teknolojilerinde kullanılmasıyla ilgili çeşitli araştırmalar mevcuttur. Arsénio ve arkadaşları, yapay zekâ kavramlarını IoT nesnelerinde ve iletişim ağlarında kullanarak Akıllı Nesneler geliştirmeyi önermişlerdir. Gelecekteki IOT sistemlerinin "kendiliğinden konfigürasyon, kendini optimize etme, kendini koruma ve kendini iyileştirme" gibi özelliklerinin olması planlanmaktadır (Arsénio ve ark, 2014).

3.2.5.5. Bulut bilişim ile entegrasyon

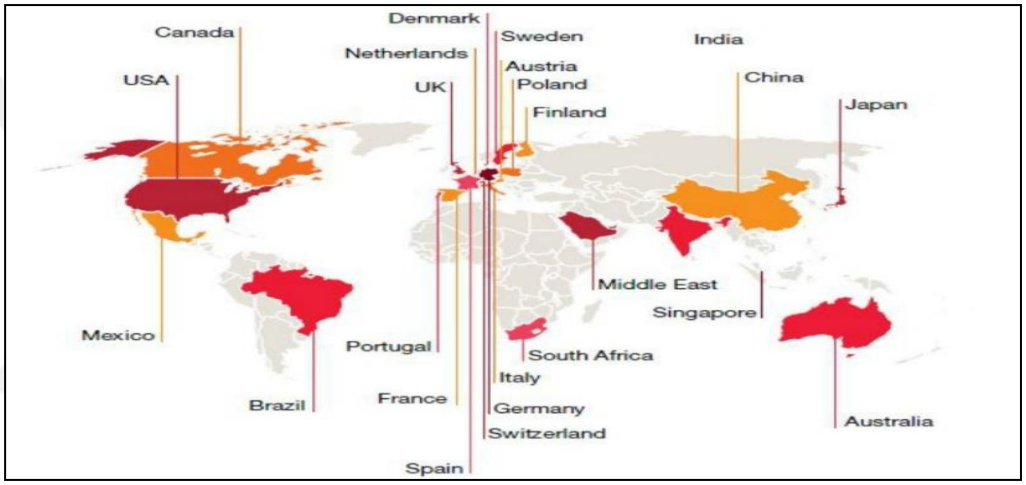
Bulut Bilişiminin IoT teknolojilerinde kullanılmasıyla çok farklı nesnelerin birbirlerine bağlanmasıyla çok çeşitli nesnelere erişebilmemiz sağlanmış olacaktır. Bu anlamda bu nesnelere erişebilmeyi sağlayacak uygulama platformlarına odaklanmak gerekecektir.

3.3. Endüstri 4.0

Endüstri 4.0 kavramının esas amacı Nesnelerin İnterneti (IoT) teknolojileri ile Endüstri teknolojilerini buluşturmaya çalışmaktır. Endüstrinin gelişimi Endüstri 4.0 ile birlikte 4 evrede tanımlanmaktadır. Üretimde Makinelerin kullanılmaya başlaması Endüstri 1.0, sonrasında elektrikli makinelerle seri üretim bantlarının kullanılmaya başlaması Endüstri 2.0, bilgisayar teknolojilerinin gelişmesiyle birlikte bu teknolojilerin sanayide üretim makinelerinde kullanılmaya başlaması ise Endüstri 3.0 olarak adlandırılmıştır. IoT teknolojilerinin ve İnternet altyapısının hızlı bir şekilde gelişmesi bu teknolojilerin endüstride kullanımını hızlandırmıştır. IoT teknolojilerinin Endüstride kullanılmasıyla da günümüzde Endüstri 4.0 kavramı ortaya çıkmış ve en popüler araştırma konularından biri haline gelmiştir.

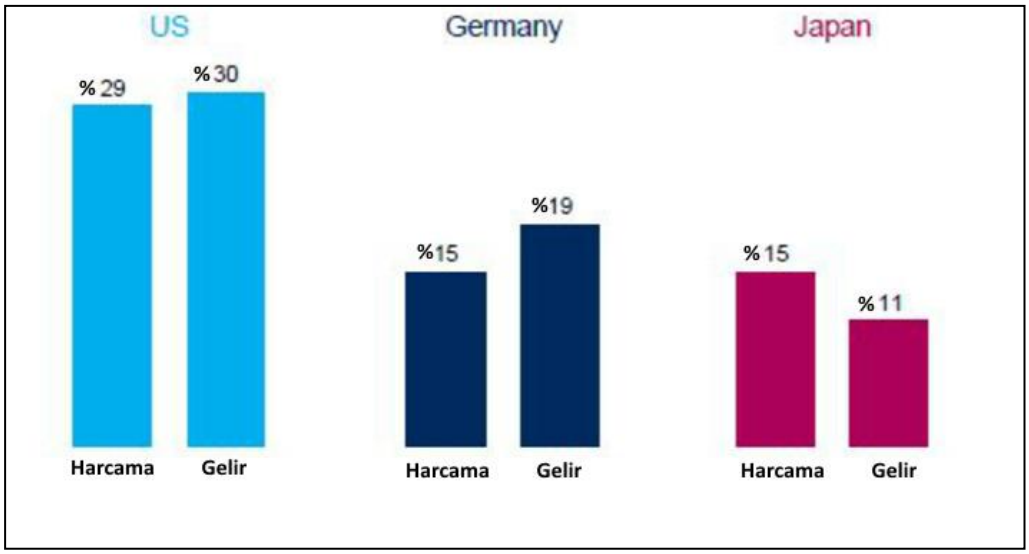
3.3.1. Dünyada endüstri 4.0

2016 yılı içerisinde PwC araştırma şirketinin 9 farklı sektörde ve 26 ülkede 2000 şirketle Endüstri 4.0 başlığı adı altında bir anket çalışması gerçekleştirmiştir. Araştırmaya katılan bu şirketlerin %33 ü kendilerini ileri düzeyde teknolojiyi kullanan, dijitalleşmiş bir şirket olarak görmektedir. Şirketlerin %72 si ise 5 yıl içerisinde ileri seviyede dijitalleşmiş olmayı planlamaktadırlar. Endüstri 4.0 a ilk uyum sağlayan şirketlerin %27 si bu teknolojiye uyum sayesinde %30 gider tasarrufu ve %30 gelir artışı sağlayacaklarını düşünmektedirler.



Şekil 3.18. Anket çalışmasının gerçekleştirildiği ülkeler

BCG (Boston Consulting Group) tarafından 2015 yılında Almanya İmalat Sanayi de gerçekleştirilen bir diğer çalışmaya göre, Endüstri 4.0 ile birlikte gelir, yatırım, istihdam ve verimlilik gibi konularda önemli gelişmeler yaşanacaktır. İmalat sanayisinde 91-150 milyar Euro luk bir kazanç sağlamayı düşünmektedirler.

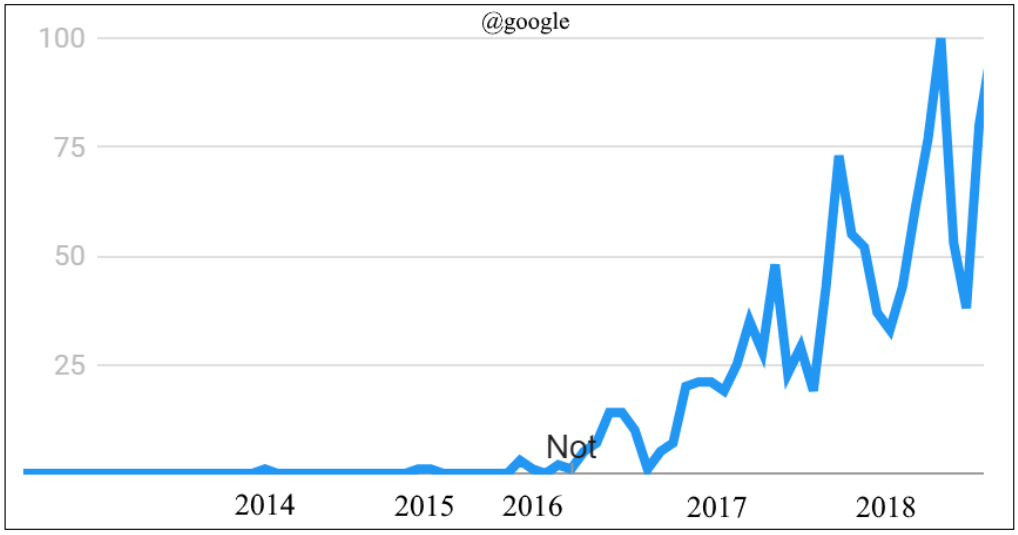


Şekil 3.19. ABD, Japonya ve Almanya'da 2015 yılında gerçekleştirilen çalışmaya göre toplam gelire göre harcama payları ve toplam gelire göre gelir payları (McKinsey)

Birçok ülke ve kuruluş IoT standartlarının geliştirilmesi konusunda çalışmalar yapmaktadır. Uluslararası Tele-iletişim Birliği, Uluslararası Elektroteknik Komisyonu, Uluslararası Standardizasyon Örgütü, IEEE, Avrupa Elektroteknik Standardizasyon Komitesi, Çin Elektronik Standardizasyon Enstitüsü ve Amerikan Ulusal Standartlar Enstitüsü gibi sayısız organizasyon bu alanda ciddi çalışmalar yürütmektedir.

3.3.2. Türkiye ve endüstri 4.0

Google ‘de Türkiye üzerinden yapılan aramalarda Endüstri 4.0 kavramının Türkiye’de nasıl bir popülerliğe sahip olduğu Şekil 3.20. verilmektedir. Grafikteki veriler bilimsel bir veri olmamakla birlikte Endüstri 4.0 kavramının Türkiye'de ne zaman konuşulmaya başlandığının bir göstergesi olabilir.



Şekil 3.20. Google' de Türkiye üzerinden Endüstri 4.0 kavramının aranması

Türk Sanayicileri ve İş İnsanları Derneği, Deloitte Türkiye, Samsung Türkiye, ve GFK Türkiye bünyesinde hizmet veren 58 şirkette görev yapmakta olan üst yöneticilerle yaptıkları görüşmeleri derledikten sonra 2016 yılında "Türkiye'deki Dijital Değişime CEO Bakışı" adıyla bir rapor hazırlayarak kamuoyuna sunmuştur. Perakende satış yapan şirketler, telekomünikasyon alanında hizmet veren kurumlar ve bankacılığın bulunduğu sektörlerde hizmet veren şirketlerle çeşitli araştırmalar gerçekleştirmişlerdir. Birlikte çalışma yapılan şirket yöneticilerinin aktardığı verilere göre, şirket çalışanları teknolojik gelişmeleri ve bu doğrultuda hazırlanan dijital stratejileri anlama oranları %66'dır. Aynı zamanda bu teknolojik değişimlere yön vermesi gereken yöneticilerin oranı ise %38 seviyelerindedir. Araştırma yapılan şirket yöneticilerine dijitalleşme yani teknolojiye uyum sağlama konusunda seviyeleri sorulduğunda; %7'sinin giriş seviyesinde olduğu, %59'unun gelişmekte olduğu ve %34'ünün ise dijital alanda gelişmiş oldukları gözlemlenmiştir (TUSIAD, 2016).

Endüstri 4.0 teknolojilerinin yaygınlaşmasıyla birlikte veri analizi ve veri madenciliği konularında çalışmaların yapılması ülkemiz açısından oldukça olumlu bir gelişmedir. Nesnelerin İnterneti teknolojilerinin kullanımının artmasıyla birlikte artan verilerin korunması oldukça önemlidir. Bu verilerin korunması amacıyla şirketlerin yatırım yapma niyetinde oldukları görülmektedir.

2016 yılında Bilim Teknoloji Yüksek Kurulu tarafından düzenlenen toplantıda Endüstri 4.0'a geçişin kolaylaşması açısından önemli kararlar alınmıştır. Toplantıda, dijital robot teknolojisi, Nesnelerin İnterneti (IoT), yapay zeka, siber güvenlik, bulut bilişim ve büyük veri (Big Data) konularında şirketlerin kabiliyetlerini arttırmayı hedefleyen Ar-Ge çalışmalarının desteklenmesi ve bu konularda ivedi şekilde çalışmalar yapılması kararı alınmıştır. Aynı zamanda, ülkenin geleceğini etkileyebilecek önemli teknolojilerin Türk firmalar tarafından üretilmesini kolaylaştıracak ve teşvik edecek destek sistemlerinin tekrardan ele alınması ve takip edilerek geliştirilmesi konularında alınan kararlar Türkiye'deki yöneticilerin Endüstri 4.0 konusunda farkındalıklarının yüksek seviyelerde olduğunu göstermektedir.

Türkiye Bilimsel ve Teknolojik Araştırma Kurumu (TÜBİTAK), 2016 yılı içerisinde özel sektörde hizmet veren 1000 ayrı firma ile yapmış oldukları çalışmalar sonrasında, ülkemiz sanayisinin teknoloji kullanımı Endüstri 2.0 ile Endüstri 3.0 arasında olduğunu sonucuna varmışlardır. Yapılan bu araştırmaların gösterdiği bir diğer önemli sonucu, otomasyon sistemlerinin, siber güvenlikle ilgili fiziksel sistemlerin ve endüstride kullanılan üretim robotlarının üretim sürecinin sadece bir kısmında yer almasıdır. Yapılan bu çalışmaya katılan şirketlerin ortalama %50' sinin önümüzdeki yıllarda Endüstri 4.0 teknolojilerinin, hali hazırda kullanılan üretim makinelerine entegresi ile ilgili eylem planlarının olması, ülkemiz adına geç kalınmış olsa da yakın gelecekte bu teknolojilerin sanayimizde daha yüksek oranlarda kullanılacağı umudunu vermektedir.

Türkiye İstatistik Kurumu'ndan alınan 2015 verilerine göre, 10-249 çalışanı bulunan işletmelerin yalnızca yüzde 12'si bilgisayar ağı vasıtasıyla ürün veya hizmet siparişlerini kontrol ediyor. Üretim imkânlarımız, AB ortalamasının dörtte birinden az, Alman imalatının ise yarattığı katma değerin onda birinden daha azını oluşturmaktadır.

4. ARAŞTIRMA BULGULARI ve TARTIŞMA

Çizelgeleme problemlerinde önerilen algoritmanın verimliliğini ve literatüre katkısını ölçmek için kullanılan tekniklerden bir tanesi bulunan sonuçların bu alanda yayımlanan diğer çalışmalarla karşılaştırılmasıdır. DEATÇ problemlerinde hem statik çizelgeleme sonucu, hem de dinamik olaylardan sonra elde edilen yeniden çizelgeleme sonucunun ilk elde edilen statik çizelgeleme sonucuna göre Yüzde Sapması (YS) hesaplanarak elde edilen sonuçlar yayımlanan diğer çalışmalar ile karşılaştırılır. Bu çalışmada, statik çizelgeleme sonucunda elde ettiğimiz C_{max} ile yeniden çizelgeleme sonucunda elde edilecek olan YS hesaplanarak yayımlanan diğer çalışmalarla karşılaştırmalar yapılmıştır.

4.1. Çizelgeleme İşlemleri ve Sonuçlar

Tez çalışmamızda, çizelgeleme yapılacak olan veri setleri iki guruba ayrılmıştır. Birinci gurupta Tam esnekliğe sahip Atölye Tipi çizelgelemeler için olan veri setleri, ikinci gurupta ise Kısmi Esnek Atölye Tipi çizelgelemeler için olan veri setleri vardır.

Birinci gurup; 4 iş, 5 makine ve 12 alt operasyon (Ex1: 4x5x12), 10 iş, 7 makine ve 29 alt operasyon (Ex2: 10x7x29), 10 iş, 10 makine ve 30 alt operasyon (Ex3: 10x10x30), 15 iş, 10 makine ve 56 alt operasyon (Ex4: 15x10x56) boyutlarındaki veri setlerinden oluşmaktadır (Kacem ve ark, 2002a).

İkinci grup; 8 iş, 8 makine ve 27 alt operasyondan (Ex5: 8x8x27) oluşan veri seti (Kacem ve ark, 2002b) ve 5 iş, 6 makine ve 25 alt operasyondan (WA: 5x6x25) oluşan veri setlerinden oluşmaktadır (Wang ve ark, 2017).

Her iki gruptaki veri setlerini DEATÇ problemleri için önermiş olduğumuz hibrit algoritma (HA) ile çözüp yine bu alanda yayımlanmış ve veri setlerinin aynı dinamik olaylarla çözüldüğü çalışmalarla karşılaştırmalar yapılmıştır. Karşılaştırma yaptığımız yayımlarda paylaşılan dinamik olaylar Çizelge 4.1.'de detaylı bir şekilde

verilmiştir. Çizelgede verilen dinamik olaylar (i, t, r) şeklinde gösterilmektedir. Burada; i arızalanan makine numarasını, t arızanın geçekleştiği zamanı ve r arızalı makinenin tamir edilip tekrar çizelgeleme işlemine dâhil edilme süresini temsil etmektedir.

Çizelge 4.1. Dinamik olay (Makine arızası) senaryoları

Veri Seti	Problem Boyutu (n x m x o)	Dinamik Olay (i, t, r)
Ex1	4 x 5 x 12	(4, 9, 3)
Ex2	10 x 7 x 29	(3, 5, 3)
Ex3	10 x 10 x 30	(4, 3, 6)
Ex4	15 x 10 x 56	(5, 3, 5)
Ex5	8 x 8 x 27	(4, 5, 3)
WA	5 x 6 x 25	(6, 20, 10)

Önermiş olduğumuz hibrit algoritma (HA) ile elde ettiğimiz yeniden çizelgeleme sonuçlarının verimliliğini gösteren Yüzde Sapma (YS)'nın hesaplanma formülü denklem (4.1) de verilmiştir. Denklemde C_{maxs} hiçbir dinamik olayın olmadığı ilk hesaplanan statik çizelgeleme sonucunu, C_{maxy} ise dinamik olay meydana geldikten sonra hesaplanan dinamik çizelgeleme sonucunu temsil etmektedir.

$$YS = (C_{maxy} - C_{maxs}) / C_{maxs} \times 100\% \quad (4.1)$$

Önermiş olduğumuz HA'yı kullanarak geliştirdiğimiz yazılım aracılığı ile elde ettiğimiz Statik Çizelgeleme (SÇ), Çizelge 4.1. de verilen dinamik olaylar sonrası elde edilen Yeniden Çizelgeleme (YÇ) ve Yüzde Sapma (YS) sonuçlarını aynı dinamik olayları kullanarak elde edilen sonuçların paylaşıldığı yayımlar ile karşılaştık. Bu karşılaştırmaya ait sonuçlar Çizelge 4.2. de verilmiştir. Bu yayımlardan bir tanesi, Oyun teorisi yöntemi ve yuvarlama mantığına dayalı yeniden çizelgeleme yöntemlerinin birleşiminden oluşan hibrit bir metodu (HGR) kullanarak (Tian ve ark., 2019), diğeri ise değişken aralıklı yeniden çizelgeleme stratejisi metodunu (VIRS) kullanarak aşağıdaki sonuçları elde etmişlerdir (Wang ve ark., 2017).

Karşılaştırma yapılan makalelerde WA veri seti için HGR yöntemiyle elde edilen sonuç, Ex1, Ex2, Ex3, Ex4 ve Ex5 veri setleri için VIRS ile elde edilen sonuçlar

bulunmamaktadır. Veri setleri ve dinamik olaylar farklı makalelerden alındığı için böyle bir tablo oluşmuştur.

Çizelge 4.2. HA ile elde edilen sonuçların HGR ve VIRS ile karşılaştırılması

Veri Seti	HA			HGR			VIRS		
	SÇ	YÇ	YS(%)	SÇ	YÇ	YS(%)	SÇ	YÇ	YS(%)
Ex1	**11**	**12**	**9.09**	11	13	18.18	-	-	-
Ex2	**11**	**11**	**0**	11	13	18.18	-	-	-
Ex3	**7**	**7**	**0**	7	9	28.57	-	-	-
Ex4	12	**12**	**0**	12	15	25.00	-	-	-
Ex5	14	**15**	**7.14**	14	16	14.29	-	-	-
WA	**27**	**27**	0	-	-	-	35	35	0

Tian ve arkadaşları önermiş oldukları HGR algoritmasıyla elde ettikleri YS sonuçlarını yine kendi yayımlamış oldukları çalışmalarında üç farklı çalışma ile elde edilen YS sonuçlarıyla karşılaştırmışlardır. HGR (Tian ve ark, 2019), RSS (Wang ve ark., 2013), AAR (Kutanoğlu ve Sabuncuoğlu, 2001) ve SGR (Hasan ve ark., 2011) ile elde edilen YS sonuçlarının önermiş olduğumuz HA ile elde edilen YS sonuçları ile karşılaştırılması Çizelge 4.3. de verilmiştir. Yine bu tabloda da elde edilen veriler önermiş olduğumuz HA algoritmasının verimliliğini göstermektedir.

Çizelge 4.3. HA ile elde edilen sonuçların farklı çalışmalar ile karşılaştırılması

Veri Seti	Yüzde Sapma (%)				
	HA	HGR	RSS	ARR	SGR
Ex1	**9.09**	18.18	18.18	18.18	18.18
Ex2	**0**	18.18	36.36	36.36	18.18
Ex3	**0**	28.57	57.14	42.86	57.14
Ex4	**0**	25.00	33.33	25.00	25.00
Ex5	**7.14**	14.29	35.71	21.43	14.29

Yeniden çizelgeleme süreçlerinin nasıl gerçekleştiğini adına WA ve Ex4 veri setlerinin çizelgeleme işlem adımları ve sonuçları (Gantt Şemaları) detaylı bir şekilde paylaşılacaktır.

4.1.1. WA veri seti için işlem adımları

Her operasyonun farklı makinelerde farklı işlenme süreleri vardır. Aynı anda iki farklı iş (J_1,J_2) üretilebilmekteyken her işin alt operasyonları kendi aralarında işlem

sıraları bulunmaktadır (Ör: $O_{1.1}$ üretilmeden $O_{1.2}$ üretilemez). Çizelge 4.4.'de her operasyonun makineler üzerindeki işlem süreleri ve operasyonların işlenme sırası verilmiştir.

C# yazılımı dili ile geliştirmiş olduğumuz çizelgeleme ve üretim simülasyonu programına Çizelge 4.4. de verilen WA' ya ait veri seti girilip ayarlandıktan sonra program çalıştırılır. Program açıldıktan sonra DEATÇ problemleri için geliştirmiş olduğumuz HA algoritmasının istemiş olduğu makine sayısı, parçacık sayısı ve iterasyon sayısı parametreleri girilir. Ardından çizelgeleme işlemini gerçekleştir butonu tıklanarak çizelgeleme gerçekleştirilir (Şekil 3.1.).

Çizelge 4.4. 5x6x25 Boyutlu DEATÇ (WA) problemi makine işlem süreleri

İşler	Operasyonlar	İşlem Süreleri					
		M_6	**M_5**	**M_4**	**M_3**	**M_2**	**M_1**
J_1	O_{11}	—	—	—	4	3	2
	O_{12}	—	4	2	—	3	—
	O_{13}	—	—	—	5	4	1
	O_{14}	—	5	3	—	—	4
	O_{15}	9	6	7	8	6	—
J_2	O_{21}	—	2	—	5	—	3
	O_{22}	—	6	—	—	3	4
	O_{23}	11	7	—	4	—	—
	O_{24}	5	—	7	—	5	—
	O_{25}	—	5	—	7	5	4
J_3	O_{31}	—	—	—	—	6	5
	O_{32}	—	5	3	—	4	—
	O_{33}	12	9	—	13	—	—
	O_{34}	—	8	4	7	5	6
	O_{35}	8	7	—	—	6	8
J_4	O_{41}	—	—	9	7	—	9
	O_{42}	5	—	4	—	6	—
	O_{43}	3	—	—	3	—	1
	O_{44}	4	5	7	9	—	6
	O_{45}	—	8	8	7	8	—
J_5	O_{51}	6	3	9	7	3	4
	O_{52}	5	—	4	—	6	5
	O_{53}	3	—	—	4	—	6
	O_{54}	7	—	7	—	5	—
	O_{55}	—	8	7	8	—	7

DEATÇ sonuçlarını sağlıklı bir şekilde karşılaştırabilmek amacıyla veri setlerini aldığımız yayımlanmış çalışmalarda yer alan dinamik olaylarla aynı olması sağlanmıştır. Bizim karşılaştırma yaptığımız birinci makalede WA veri seti kullanılmış ve bu veri seti üzerinde 4 farklı dinamik olay gerçekleştirilerek sonuçlar paylaşılmıştır (Wang ve ark., 2017).

Dinamik olaylara ait bilgiler;

DO1: Üretimin t=20 zamanında Makine 6 (M_6) arızalanmış. T=30 zamanında ise tamir edilmiş ve üretime hazır hale getirilmiştir (6, 20, 10).

DO2: Üretimin t=20 zamanında 4. İş (J_4) bozulur. Bu işin tekrar üretilmesi gerekecektir.

DO3: Üretimin t=20 zamanında 4. İşin 4. Operasyonu bozulur. Bu operasyonun tekrar üretilmesi gerekecektir.

DO4: Üretimin t=20 zamanında acil üretilmesi gereken bir iş (J_6) gelecektir (Çizelge 4.5. de yeni gelecek olan işe ait işlem süresi bilgileri verilmiştir).

Çizelge 4.5. Üretim sırasında acil gelen işe (J_6) ait makine işlem süreleri

İş	Operasyon	İşlem Süreleri					
		M_6	M_5	M_4	M_3	M_2	M_1
J_6	O_{11}	—	—	—	4	3	2
	O_{12}	—	4	2	—	3	—
	O_{13}	—	—	—	5	4	1
	O_{14}	—	5	3	—	—	4
	O_{15}	9	6	7	8	6	—

Önermiş olduğumuz HA algoritması ile yeniden çizelgeleme yapılırken, 4 farklı dinamik olay, 3 farklı strateji kullanılarak gerçekleştirildi. Kullanılan hangi stratejinin daha verimli olduğunu anlamak için ise elde edilen SÇ, YÇ ve YS değerleri birbirleriyle karşılaştırıldı (Çizelge 4.6.).

DO1, DO2, DO3 ve DO4 dinamik olayları için ayrı ayrı test ettiğimiz stratejiler;

S1: Aktif operasyonların bitiş süresini hesaba katarak çizelgele

S2: Aktif Operasyonların bitmesini bekleyerek çizelgele

S3: Aktif Operasyonları iptal ederek çizelgele

Çizelge 4.6. HA ile gerçekleştirilen yeniden çizelgeleme stratejilerinin karşılaştırılması

Dinamik Olay	SÇ (C_{max})	S1		S2		S3	
		YÇ(C_{max})	YS(%)	YÇ(C_{max})	YS(%)	YÇ(C_{max})	YS(%)
DO1	27	**27**	**0**	31	14.81	33	22.22
DO2	27	**43**	**59.25**	50	85.18	46	70.37
DO3	27	**27**	**0**	34	25.92	29	7.41
DO4	27	**34**	**25.92**	40	48.15	37	37.04

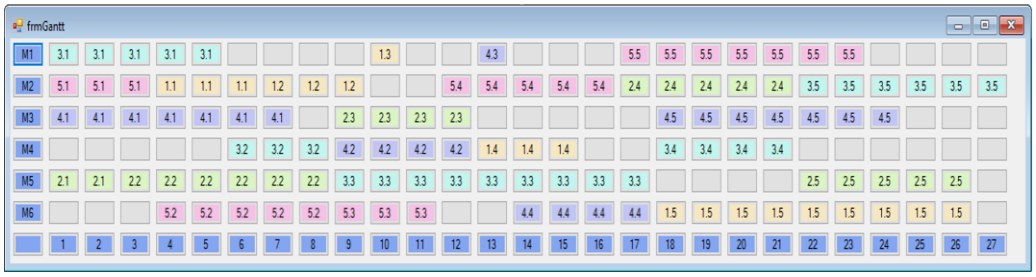
Tüm dinamik olaylar için ayrı ayrı elde ettiğimiz sonuçlar göz önünde bulundurulduğunda en iyi C_{max} değerini S1 stratejisinde elde edebildik. Bu durumda S1 stratejisi tez çalışmamızda önerdiğimiz yeniden çizelgeleme stratejisi olacaktır.

Wang ve arkadaşları tarafından 2017 yılında yayımlanan makalede VIRS yöntemi kullanılarak Çizelge 4.4. de verilen WA veri seti için sonuçlar elde edilmiştir (Wang ve ark, 2017). Geliştirmiş olduğumuz HA algoritması ve S1 yeniden çizelgeleme stratejisi ile WA veri seti için elde ettiğimiz sonuçlar ile Wang ve arkadaşlarının VIRS yöntemi ile elde ettiği sonuçları karşılaştırdık (Çizelge 4.7.). VIRS yöntemi ile elde edilen C_{max} değeri 35 iken, önermiş olduğumuz HA algoritması ve yeniden çizelgeleme yöntemi (S1) ile elde ettiğimiz C_{max} değeri 27 olmuştur. Bu durum önermiş olduğumuz algoritma ve yöntemin verimliliğini göstermektedir.

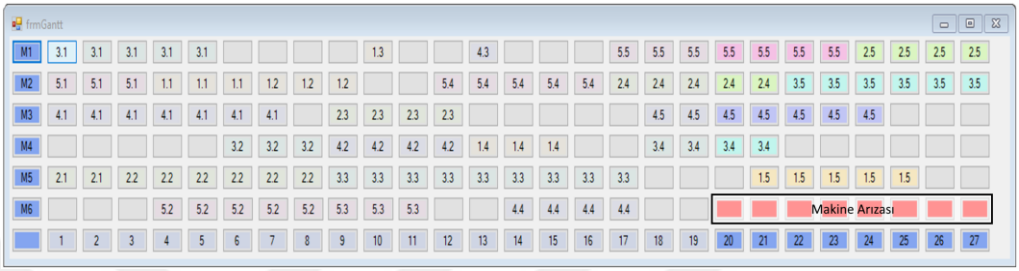
Çizelge 4.7. WA veri seti için HA ve S1 ile elde edilen sonuçların VIRS ile karşılaştırılması

Veri Seti	HA		VIRS	
	Statik Çizelgeleme SÇ	Yeniden Çizelgeleme YÇ	SÇ	YÇ
DO1	**27**	**27**	35	35
DO2	**27**	**43**	35	43
DO3	**27**	**27**	35	35
DO4	**27**	**34**	35	37

WA veri seti için elde ettiğimiz SÇ çözümüne ait Gantt şeması Şekil 4.1. de, DO1 dinamik olayından sonra YÇ çözümüne ait Gantt şeması ise Şekil 4.2. de gösterildiği gibidir.



Şekil 4.1. WA veri seti için HA ile elde edilen Gantt şeması



Şekil 4.2. DO1 Dinamik olayından sonra WA veri seti için HA ve S1 stratejisi kullanılarak elde edilen Gantt şeması

4.1.2. EX4 veri seti için işlem adımları

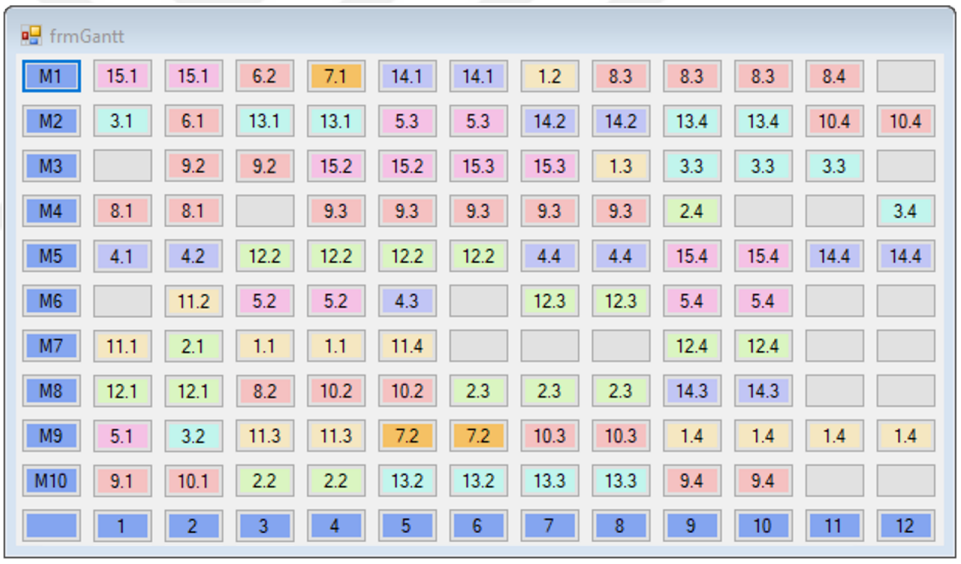
Bu veri setimizde de her operasyonun farklı makinelerde farklı işlenme süreleri vardır. Aynı anda iki farklı iş (J_1,J_2) üretilebilmekteyken her işin alt operasyonları kendi aralarında işlem sıraları bulunmaktadır. EK 1'de her operasyonun makineler üzerindeki işlem süreleri ve operasyonların işlenme sırası verilmiştir.

Geliştirmiş olduğumuz program üzerinde EK 1'de verilen Ex4' e ait veri seti girilip ayarlandıktan sonra program çalıştırılır. Program açıldıktan sonra DEATÇ problemleri için geliştirmiş olduğumuz HA algoritmasının istemiş olduğu makine sayısı, parçacık sayısı ve iterasyon sayısı parametreleri girilir. Ardından çizelgeleme işlemini gerçekleştir butonu tıklanarak çizelgeleme gerçekleştirilir (Şekil 3.1.).

Ex4 veri seti ise karşılaştırma yaptığımız ikinci makalede kullanılmış ve bu veri seti üzerinde 1 dinamik olay gerçekleştirilerek sonuçlar paylaşılmıştır (Tian ve ark., 2019). Bu dinamik olay; Üretime başlandıktan bir süre sonra (t=3) 5. makine arızalanır. Belirli bir süre sonra (t=8) makine arızası giderilip tekrar çizelgeleme sürecine dâhil edilir (Çizelge 4.1.).

Bu veri setinde de S1 stratejisi kullanılarak yeniden çizelgeleme işlemi gerçekleştirilmiştir. Tian ve arkadaşları tarafından 2019 yılında yayımlanan makalede HGR yöntemi kullanıldığında, SÇ için bulunan C_{max} değeri 12 ve yeniden çizelgeleme sonrası hesaplanan YS değeri %25 olmuştur. Bizim önermiş olduğumuz HA algoritması ve yeniden çizelgeleme yöntemi (S1) kullanıldığında SÇ için bulunan C_{max} değeri 12 ve yeniden çizelgeleme sonrası hesaplanan YS değeri %0 olmuştur (Çizelge 4.2.). Bu durum önermiş olduğumuz algoritma ve S1 yönteminin verimliliğini göstermektedir.

Ex4 veri seti için elde ettiğimiz SÇ çözümüne ait Gantt şeması Şekil 4.3. de, Dinamik olay (5, 3, 5) gerçekleştikten sonra YÇ çözümüne ait Gantt şeması ise Şekil 4.4. de gösterildiği gibidir.



Şekil 4.3. Ex4 veri seti için HA ile elde edilen Gantt şeması



Şekil 4.4. (5, 3, 5) Dinamik olayından sonra Ex4 veri seti için HA ve S1 stratejisi kullanılarak elde edilen Gantt şeması

Arıza içeren çizelgeleme işlemleri için gerçekleştirdiğimiz üretim süreci simülasyonun işlem adımları Çizelge 4.8. de gösterildiği gibidir. Bu simülasyon sürecinde iki kez çizelgeleme yapılmaktadır. 1. Çizelgeleme çalışmamızda önerdiğimiz HA ile yapılan ve herhangi bir dinamik olayın öngörülmediği çizelgelemedir. 2. Çizelgeleme ise üretim süreci içerisinde herhangi bir zamanda oluşan (Örneğimizde t=20) makine arızası sonrası gerçekleştirilen çizelgelemedir.

Çizelge 4.8. Arıza İçeren Üretim Süreci

Adım 1	Çizelgeleme işlemi ana sunucu üzerinde gerçekleştirilir.
Adım 2	Başla komutu ve çizelgeme sonucuna göre makinelerin yapacakları iş tabloları, ilgili makinelere **nesnelerin İnterneti** teknolojileri kullanılarak iletilir.
Adım 3	Üretime başlandıktan bir müddet sonra makinelerden biri her hangi bir nedenden dolayı devre dışı kalırsa sunucu bilgisayara üretimin durduğunu belirten bir arıza komut iletir.
Adım 4	Sunucu bilgisayar, diğer makineler üzerinde devam etmekte olan üretimin durmasını sağlayacak komutu göndererek üretimi durdur.
Adım 5	Sunucu bilgisayar, aktif üretimde olan operasyonların bitiş sürelerini, tamamlanan ve tamamlanmayan operasyonların durumları göz önünde bulundurularak tekrarlanacak olan çizelgeleme için gerekli giriş parametrelerini hazır hale getirilir.
Adım 6	Sunucu bilgisayar, üretimi gerçeklememiş olan tüm operasyonları tekrar çizelgeleme işlemine tabi tutarak yeni bir sonuç üretir (Fahmy ve ark, 2009).
Adım 7	Adım 2 deki işlem tekrarlanarak yeni çizelgeleme sonucuna göre üretimin devam etmesi sağlanır.
Adım 8	Yeni bir arıza olmaması durumunda üretim planlandığı şekilde tamamlanır
Adım 9	Üretim süreci ve sonucuyla ilgili veriler sunucu bilgisayarın ekranına yansıtılır.

Çizelgeleme işlem süresi, kullanılan sunucu bilgisayarının özelliklerine, parçacık sayısına ve iterasyon sayısına bağlı olarak değişkenlik gösterecektir. Bu nedenle çizelgeleme işlem süreleri göz ardı edilmektedir.

4.1.3. Donanım özellikleri ve çizelgeleme süresi

Tez çalışmamızda C# yazılım diliyle geliştirmiş olduğumuz simülasyon programının çalıştırılabilmesi için kullandığımız sunucu bilgisayarın özellikleri Çizelge 4.9. de gösterildiği gibidir. İlk olarak parçacık sayısı 100, iterasyon sayısı ise 50 olarak belirlenmiş (Kaya, 2014) ve karşılaştırma yaptığımız tüm veri setleri için bulduğumuz en iyi değerler elde edilmiştir. Programın vermiş olduğu çizelgeleme raporları incelendiğinde ilgili veri setlerinin tümünde ilk 20 iterasyonun ve her iterasyonun ilk 20 parçacığı içerisinde en iyi sonuç elde edildiği anlaşılmıştır. Bu nedenle çizelgeleme sürelerini azaltmak adına iterasyon ve parçacık sayıları 20 olarak belirlenmiş ve çizelgeleme sonuçları bu parametreler ile elde edilmiştir.

Çizelge 4.9. Çizelgeleme işlemleri için kullanılan sunucu bilgisayar özellikleri

Özellik Adı	Açıklama
İşletim Sistemi	Windows 10 Education 64 bit
İşlemci Modeli	Intel (R) Core (TM) i5-4210U
İşlemci Hızı	1.70 GHz (4 CPU), ~2.4 GHz
İşlemci Ön Bellek	3 MB
RAM	8 GB 1600 MHz
Ekran Kartı	AMD Radeon R5 M255 2GB

Programımız 100 Parçacık, 50 iterasyon ve 20 parçacık, 20 iterasyon ile çalıştırıldığında oluşan çizelgeleme süreleri Çizelge 4.10. da verilmiştir. Bu tabloda belirtilen değerler milisaniye (ms) olarak verilmiştir. Bu sayede, parçacık ve iterasyon sayılarının çizelgeleme sürelerine etkisini görebilmiş olduk.

Çizelge 4.10. Veri setlerinin çizelgeleme işlem süreleri (milisaniye)

Veri Seti	Parçacık:100, Iterasyon:50			Parçacık:20, Iterasyon:20		
	SÇ Süresi (ms)	YÇ Süresi (ms)	Toplam Süre (ms)	SÇ Süresi (ms)	YÇ Süresi (ms)	Toplam Süre (ms)
Ex1	9523 ms	3111 ms	12634 ms	621 ms	207 ms	828 ms
Ex2	53835 ms	50400 ms	104235 ms	3980 ms	3100 ms	7080 ms
Ex3	65242 ms	52956 ms	118198 ms	5550 ms	4326 ms	9876 ms
Ex4	172850 ms	165010 ms	337860 ms	16234 ms	16002 ms	32236 ms
Ex5	42000 ms	36520 ms	78520 ms	2387 ms	1971 ms	4358 ms
WA	25731 ms	5600 ms	31331 ms	1500 ms	273 ms	1773 ms

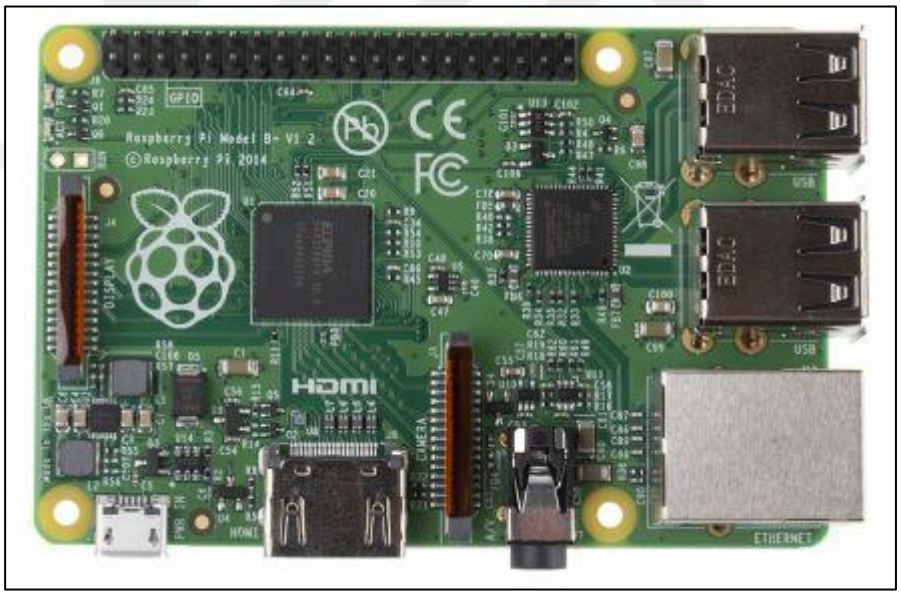
4.2. IoT deney ortamı

Üretim sürecinin makineler üzerinde başlatılması, herhangi bir dinamik olayın meydana gelmesi sonrası yapılması planlanan işlem adımlarının uygulanabilmesi için sunucu bilgisayar ile sahada bulunan makinelerin bir birleriyle haberleşmesi ve veri paylaşımı yapması gerekmektedir. Tez çalışmamızda, cihazlar arasında veri paylaşımı yapılabilmesi için Nesnelerin İnterneti teknolojilerinin kullanılması önerilmektedir. Geliştirmiş olduğumuz çizelgeleme ve üretim simülasyon programında oluşturulan sanal makineler birbirleriyle nesnelerin interneti teknolojilerini kullanarak haberleşmekte ve üretim sürecinin tamamı gerek duyulmadıkça insan müdahalesi olmadan otomatik olarak gerçekleştirilmektedir.

Üretim sürecinin gerçek sanayi ortamına uygunluğunu test etmek için bu çalışma kapsamında bir deney ortamı oluşturulmuştur. Bu deney ortamında kullandığımız donanım ve yazılım teknolojileri aşağıdaki gibidir.

- Windows işletim sistemini sorusuz çalıştıran bir sunucu bilgisayar.
- Sunucu bilgisayar üzerinde çalışacak olan, belirlemiş olduğumuz işletim sistemi (Deney ortamımızda Windows 10 işletim sistemi kullanılmıştır).
- Çizelgeleme algoritmasını çalıştıracak ve temsili sanal makinelerin başlatılıp durdurulmasını sağlayacak olan C# yazılım dili ile geliştirmiş olduğumuz çizelgeleme ve üretim simülasyonu programı
- 6 Ayrı akıllı üretim makinesini temsil edecek olan 6 Adet Raspberry Pi Model B+ Mini Bilgisayar (Şekil 4.5.)

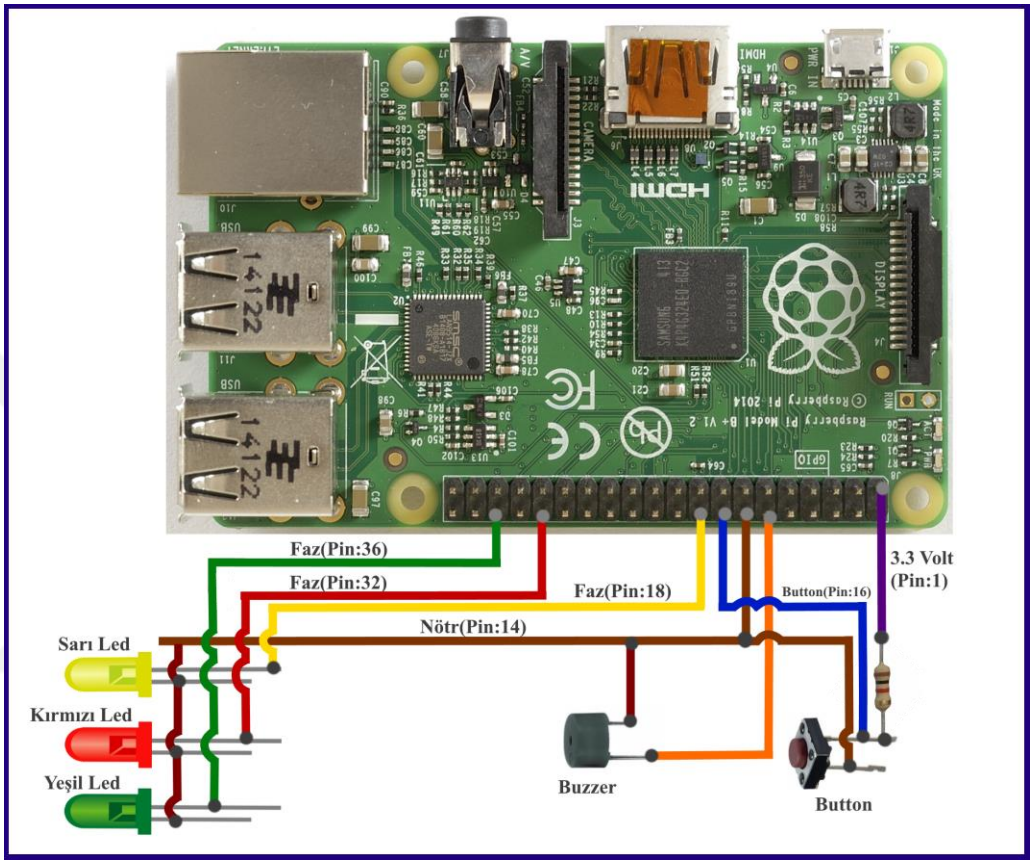
- Raspberry Pi ile birlikte kullandığımız, makinelerin farklı durumlarını gösterecek olan üç farklı renkte Led, herhangi bir makine üzerinde arıza oluşturmamızı sağlayacak olan button ve butona basıldığında sesli uyarı verecek olan buzzer (Şekil 4.6.).
- Raspberry Pi üzerinde çalışan "Rasbian" İşletim Sistemi (Linux tabanlı bir işletim sistemidir.)
- Raspberry Pi cihazlar içerisinde iş süreçlerini gerçekleştirecek, haberleşmeyi sağlayacak ve girdiler sonucunda sonuç eylemlerini gerçekleştirecek, Python yazılım dili kullanılarak geliştirmiş olduğumuz IoT servis uygulaması.
- Akıllı cihazların UDP protokolünü kullanarak birbirleriyle iletişim kurmasını sağlayacak olan yeterli sayıda porta sahip bir switch (Deney setimiz 6 adet makine olarak planlandığından en az 7 cihazın bağlanabileceği 8 porta sahip bir switch kullanılmıştır (Şekil 3.5.).



Şekil 4.5. Makineleri temsil etmek için kullanılan mini bilgisayar

Makineleri temsil etmek amacıyla son zamanlarda Nesnelerin İnterneti teknolojilerinde sıkça kullanılan Raspberry Pi mini bilgisayarı kullanılmıştır (Zhong ve Liang, 2016). Bu kısımdan sonra Raspbery Pi mini bilgisayarı makine olarak adlandırılacaktır. Sunucu bilgisayar üzerinde geliştirmiş olduğumuz uygulamaya gerekli veriler ve parametreler girildikten sonra çizelgeleme işlemini

gerçekleştiriyoruz. Sonrasında sunucu bilgisayar herhangi bir makinenin gerçekleştirmesi gereken operasyonları UDP (3. Bölümde UDP protokolü ile ilgili detaylı bilgi verilmiştir.) iletişim protokolünü kullanarak sanal makinelere mesaj gönderir ve üretimi başlatır.



Şekil 4.6. Deney düzeneği için hazırlamış olduğumuz makine tasarımı, Ledler, button ve buzzer

Her bir sanal makinenin üzerinde, Şekil 4.6. da gösterildiği gibi, sanal makinanın t zamandaki durumunu gösteren yeşil, kırmızı ve sarı renklerde toplam 3 adet led, sanal makine her tetiklendiğinde sesli uyarı verecek olan 1 adet buzzer ve herhangi bir sanal makinenin arızalanmasını sağlayacak olan 1 adet buton bulunmaktadır. Çizelge 4.11. farklı renklerdeki ledlerin ne anlama geldiğini göstermektedir.

Üretim başladıktan sonra herhangi bir makinenin üzerinde bulunan butona basıldığında makine arızalandığını ve yeniden çizelgeleme yapılması gerektiğini sunucuya mesaj olarak gönderecektir. Sunucu tarafında çalışan uygulama ise üretimin

durdurulması için diğer tüm makineleri bilgilendirecek, çizelgelemeyi tamamlanmayan operasyonlar için tekrar çalıştıracak ve makinelere kaldıkları yerden devam etmesi için bilgi mesajı gönderecektir (Şekil 3.2., Şekil 3.4.).

Çizelge 4.11. Makine üzerindeki Ledlerin Anlamları

LED DURUMLARI			AÇIKLAMA
Yeşil	Kırmızı	Sarı	
—	—	—	Cihaz Kapalı (Hazır Değil)
◯	—	—	Cihaz Açık (Sunucuyla bağlantı kurulamıyor).
1,5	—	—	Cihaz Açık, Sunucuyla Bağlantı kurulu, çizelgeleme için beklemede
1,5	1,5	—	Çizelgeleme alındı ve üretime başladı, aktif üretilen operasyon yok
1,5	1,5	◯	Çizelgeleme alındı ve üretime başladı, aktif üretilen operasyon var
1,5	—	1,5	Üretim tamamlandı
1,5	◯	—	Cihaz arızası
1,5	1,5	1,5	Yeniden çizelgeleme yapılıyor ve beklemede (Rescheduling)
—	1,5	1,5	Üretim Sorumlu tarafından durduruldu

◯ : Sürekli yanan led

1,5 : 1500 milisaniyede bir yanıp sönen led

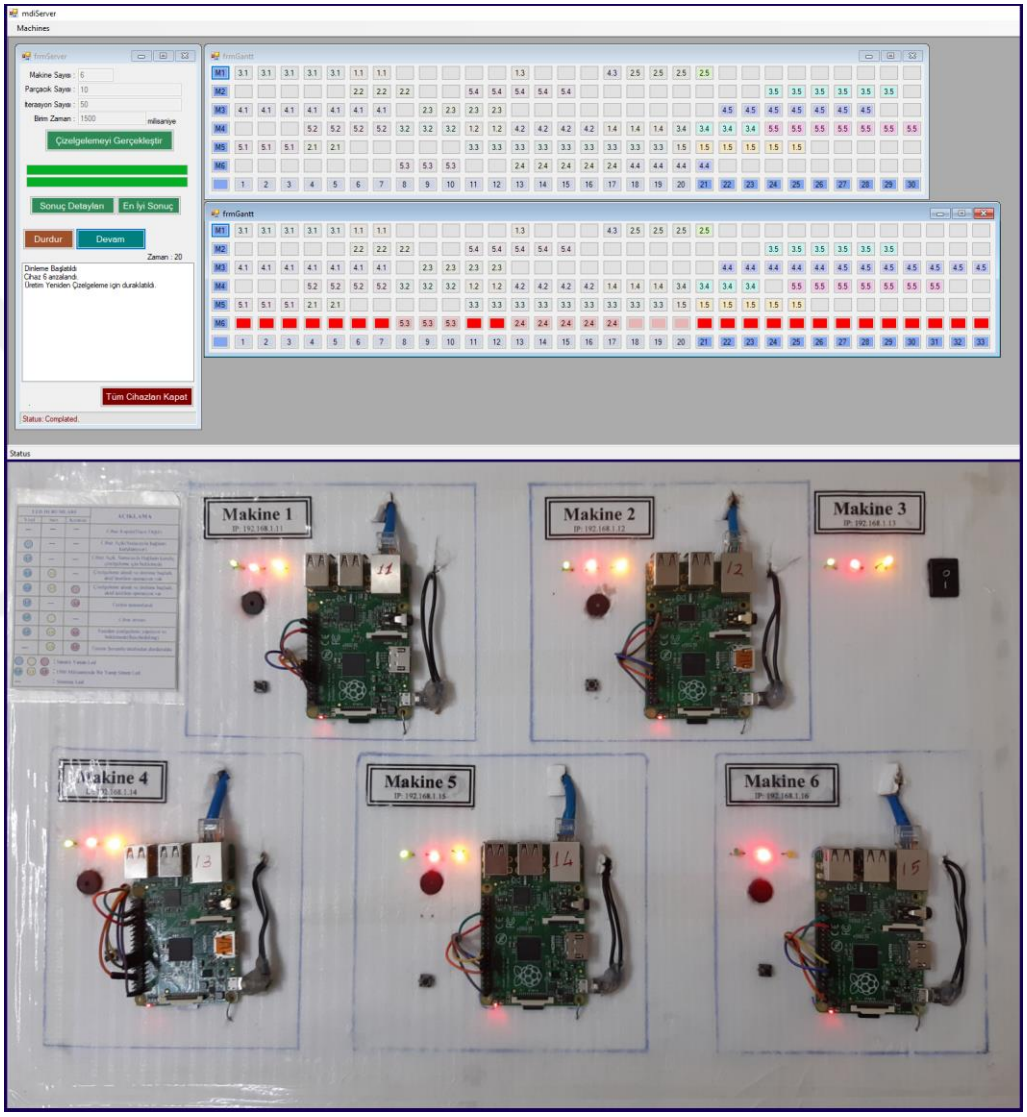
— : Sönmüş Led

Şekil 4.7. de hazırlamış olduğumuz deney düzeneğinin bir görseli bulunmaktadır. Sunucu bilgisayarda çalışan C# yazılım dili ile geliştirmiş olduğumuz yazılımı çalıştırmadan önce çizelgeleme yapmak istediğimiz veri seti bilgilerini yapılandırma dosyasına giriyoruz. İlgili veriler girildikten sonra program çalıştırılır. Program çalıştırıldıktan sonraki süreci anlatacak olursak;

1. Karşımıza çıkan ekranda HA algoritmasının çalıştırılabilmesi için gerekli olan iterasyon sayısı ve parçacık sayısı parametreleri belirlenir.
2. Çizelgeleme işleminden sonra çalışacak olan simülasyonun hızını belirleyen birim zaman değeri milisaniye olarak girilir ve ardından "Çizelgelemeyi Gerçekleştir" butonu tıklanır. Çizelgeleme işlemi belirtilen parçacık sayısı, iterasyon sayısı ve kullanılan bilgisayarın özelliklerine bağlı olarak değişkenlik gösterecektir.
3. Çizelgeleme algoritması başarılı bir şekilde çalıştırıldıktan sonra elde edilen en iyi C_{max} değerine ait makine atamasının Gantt şeması görsel olarak oluşturulur.
4. Gantt şemasıyla birlikte ekranda "Üretimi Başlat" butonu çıkmış olacaktır. "Üretimi Başlat" butonu tıklandıktan sonra üretim simülasyonu çalışmaya başlayacaktır.
5. Üretimin başlaması için sunucu bilgisayarı makinelere UDP protokolünü kullanarak üretim planını gönderir. Bu üretim planını alan cihazlar üretime başlar. Üretimin hangi zamanda ve hangi aşamada olduğu makinelerin üzerlerindeki led durumlarıyla birlikte Gantt şeması üzerinden de dinamik olarak izlenebilmektedir.
6. Yeniden çizelgeleme işleminin olabilmesi için dinamik bir olayın üretilmesi gerekmektedir. Bunun için ise üretimin herhangi bir aşamasında herhangi bir makinenin (Raspberry Pi) üzerinde bulunan arıza butonuna basılır.
7. Arıza butonuna basıldıktan sonra makine sunucuya arızalandığını bildiren bir "ARIZA" komutu gönderir.
8. Arıza komutunu alan sunucu üretimin sağlam makineler üzerinde devam edebilmesi için makinelere "TEKRARLANIYOR" komutu gönderir.
9. "TEKRARLANIYOR" komutu alan sağlam makineler yeniden çizelgelemenin sonucu bekler.
10. Yeniden Çizelgeleme işlemi sonucunda sunucu üzerinde yeni bir Gantt şeması oluşturulur ve makinelere "DEVAM" komutu gönderilerek üretimin devam etmesi sağlanır.
11. Üretimi tamamlayan cihazlar üretimi tamamladıklarını belirten "TAMAMLANDI" komutunu sunucuya gönderirler.

12. Üretimin makinelerde bittiğini gösteren tamamlandı mesajları ekranda gösterilerek süreç bitirilir.

Yukarıda anlatılan tüm süreç adımlarını detaylı bir şekilde gösteren akış diyagramları Şekil 3.1., Şekil 3.2. ve Şekil 3.3. de verilmiştir.



Şekil 4.7. Hazırlanan deney düzeneğinin görseli

5. SONUÇLAR ve ÖNERİLER

EATÇ yönteminde makinenin arızalanması veya devre dışı kalması gibi dinamik olaylar göz önünde bulundurulmazken önerdiğimiz modelde (HA) dinamik bir olayın gerçekleşmesi sonrası çizelgeleme yöntemimiz, önerdiğimiz S1 stratejisi doğrultusunda tekrarlanarak, ilk yapılan çizelgeme sonucunda oluşan yüzde sapmayı (YS) en aza indirmektedir (EK 4).

Üretim sürecinde meydana gelmesi muhtemel olan dinamik olayların tespiti çizelgeleme sonucunun uygulanabilirliği açısından önem arz etmektedir. Klasik yöntemlerde bu tür dinamik olaylar meydana geldikten sonra sahada çalışan personelin olayı tespit etmesi, tamamlanmış, yarıda kalmış ve tamamlanmamış operasyonların raporlanarak çizelgelemenin tekrarlanması, C_{max} değerinin artmasına sebep olmaktadır. Özellikle makine arızası gibi dinamik olayların geç fark edilmesi üretimin başarıya ulaşmasını büyük oranda tehdit edecektir.

Dinamik olayın gerçekleşmesinden sonra planlamada meydana gelecek sapmayı en aza indirmek öncelikli amacımız olmuştur. Özellikle bu süreçlerde yaşanan, insan faktöründen kaynaklı hataları minimize etmek ve raporlama sürecini hızlandırmak gerekmektedir. Dinamik olayların tespiti (Makine arızası vb.) ve sonrasında çizelgelemenin hatasız tekrarlanması, çizelgeleme sonucunun tekrar uygulanması ve bu süreçlerin doğru bir şekilde analiz edilerek raporlanması aşamalarını otomatik hale getirilmesi bu tez çalışmamızın esas amacıdır. Bahsedilen süreçlerin otomatik hale getirilmesi için nesnelerin interneti (IoT) teknolojileri kullanılmıştır. Bu sayede hem arızanın fark edilmesi ivedi bir şekilde sağlanmış hem de üretimin hataları minimize edilerek üretimin devamlılığı sağlanmıştır.

Gelecek çalışmalarda bu makineler üzerinde gerçekleşen üretim miktarı, makine kullanım oranı, belirli bir zaman aralığında bir makinenin kaç kez arızalandığı gibi işlemlerin kayıtları tutularak ana sunucu üzerinde anlık raporlar oluşturulabilir. Bu raporlar oluşturulurken elde edilen veriler, veri madenciliği ve sezgisel yöntemler

kullanılarak analizleri gerçekleştirilebilir. Analiz sonuçları, üretim ortamının verimliliğini ölçmek ve üretim sırasında yaşanan sıkıntıların sebeplerini anlamak için karar vericilere yol gösterecektir.

Endüstri 4.0'ın esas amaçlarından birisi nesnelerin interneti teknolojilerini kullanarak sanayide gerçekleşen süreçlerin otomatik hale getirilmesiyle birlikte, üretim süreçlerinde yaşanan tüm adımların veri kayıtlarını tutmak, arşivlemek ve elde edilen verilerin ülke genelinde tek bir merkezde toplanmasını sağlayarak büyük veriyi (Big Data) oluşturmaktır. Oluşturulan bu verilerin analiz edilmesiyle birlikte ülke genelinde üretim faaliyetlerinin birbiriyle olan ilişkileri daha iyi anlaşılmış ve eylem planlarını daha verimli bir şekilde uygulanabilir olacaktır. Bu nedenle tasarladığımız üretim ortamından elde ettiğimiz raporların, internet üzerinde belirlenen veri merkezlerine uygun bir şekilde gönderilmesi ve yayımlanması için gerekli çalışmalar yapılabilir.

Ülkemiz genelinde üretimde kullanılan çoğu makinenin bilgisayarlı (akıllı) makine değildir. Akıllı üretim makinelerinin maliyetlerinin yüksek olması nedeni ile mevcut makinelerin akıllı hale getirilmesinde yaşanan mali sıkıntılar, üretim yapan firmaların Endüstri 4.0 a geçişini zorlaştırmaktadır. Tez çalışmamızda kullanılan Raspberry Pi Mini bilgisayarı ile akıllı olmayan makineler akıllı hale getirilebilirler. Bu konuda yapılacak çalışmalarla akıllı olmayan makinelerin de nesnelerin interneti ağına dâhil edilebilmesi sağlanmış olacaktır. Bu sayede küçük çaplı üretim firmalarının Endüstri 4.0 a geçişişini kolaylaştıracak uygulamalar geliştirilebilecektir.

KAYNAKLAR

AGRAWAL, P. and BHURARIA S., 2012. Near field communication. SETLabs Bridfings, 10: 67–74.

AKTAŞ, B., 2018. Üretim Etkinliğinin Artırılmasında Anlık Veri Analizi Yaklaşımı Ve Bir Uygulama. Dokuz Eylül Üniversitesi, Sosyal Bilimler Enstitüsü, Yüksek Lisans Tezi, İzmir, 71s.

ALADAĞ, A., 2010. Tekrar İşlemeli Esnek Atölye Tipi Çizelgeleme Problemi için Yapay Bağışıklık Sistemi ile Bir Çözüm Yaklaşımı. Osman Gazi Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, Eskişehir, 97s.

ARSÉNİO, A., SERRA, H., FRANCİSCO, R., NABAİS, F., ANDRADE, J. and SERRANO, E., 2014. Internet of Intelligent Things: Bringing artificial intelligence into things and communication networks. Inter-cooperative Collective Intelligence: Techniques and Applications, 495:1–37.

ATZORI L., LERA A. and MORABITO, M., 2010. The Internet of Things: A survey. Computer Networks, 54:2787–2805.

ATZORI, L., IERA, A., MORABIOTO, G. and NİTTİ, M., 2012. The social internet of things (SIoT)-when social networks meet the internet of things: Concept, archi- tecture and network characterization. Computer Network, 56:3594–3608.

BO, Y., and GUANGWEN, H., 2009. Supply chain information transmis- sion based on RFID and internet of things. Proceedings of the Second ISECS International Colloquium on Computing, Communication, Control, and Management (CCCM '09), August 2009, Sanya, China,166–169.

BAKER, K. R., TRIETSCH, D., 2009. Principles of Sequencing and Scheduling. A John Wiley & Sons, Inc Publication, Hoboken, New Jersey, 510s.

BORAY, M. B., 2007. Paralel Tezgahlarda Arıza Halinde Çok Amaçlı Çizelgeleme. Yıldız Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, İstanbul, 142s.

BÜYÜKSÜNETÇİ, A. S., 2006. Tepkin Çizelgeleme Yaklaşımının Akış Tipi Atölye Ortamında Etkinliğinin Analizi. Osman Gazi Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, Eskişehir.

COWLING, P. and JOHANSSON, M., 2002. Using real time information for effective dynamic scheduling. European Journal of Operational Research, 139: 230–244.

CURRAN, K., MILLAR, A. and GARVEY, C., 2012. Near Field Communication. International Journal of Electrical and Computer Engineering (IJECE), June 2012, University of Ulster, Magee Campus, Northern Ireland, 371-382.

ÇÖREKÇİ, C., 2014. Atölye Tipi Üretimde Simülasyon Teknikleri İle Dinamik Çizelgeleme Ve Atölye Simülasyonu. Kırıkkale Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, Kırıkkale, 123s.

DİMİTRAKOPOULOS, G., 2011. Intelligent transportation systems based on internet-connected vehicles: fundamental research areas and challenges. Proceedings of the 11th International Conference on ITS Telecommunications (ITST '11), August 2011, Saint Petersburg, Russia, 145–151.

FAHMY, S. A., BALAKRISHNAN, S., ELMEKKAWY, T. Y., 2009. A generic deadlock-free reactive scheduling approach, International Journal of Production Research, 47:5657-5676.

FERREIRA P., MARTINHO, R. and DOMİNGOS, D., 2010. Iot-aware business processes for logistics: limitations of current approaches. Proceedings of the Inforum Conference, 3:612–613.

HAUBER-DAVİDSON, G. and IDRİS, E., 2006. Smart water metering, Water. 33: 56–59.

JIANG, Y., DING, Q. and WANG, X., 2016. A Recovery Model for Production Scheduling: Combination of Disruption Management and Internet of Things. Hindawi Publishing Corporation Scientific Programming, 2016(8264879): 9s.

KARNOUSKOS, S., 2010. The cooperative internet of things enabled smart grid. Proceedings of the 14th IEEE International Sym- posium on Consumer Electronics (ISCE ’10), June, 7–10.

KAYA, S., 2014. Çok Amaçlı Esnek Atölye Tipi Çizelgeleme Problemlerinin Geliştirilmiş Parçacık Sürü Optimizasyonu ile Çözümüne Yönelik Model Önerileri. Kocaeli Üniversitesi, Fen Bilimleri Enstitüsü, Doktora, Kocaeli, 183s

KHAN, W. Z., XIANG, Y., AALSALEM, M. Y. and ARSHAD, Q., 2013 Mobile phone sensing systems: a survey. IEEE Communications Surveys & Tutorials, 15: 402–427.

KUNDAKÇI, N., 2013. Üretim Sistemlerinde Dinamik İş Çizelgeleme Problemlerinin Sezgisel Yöntemlerle Çözülmesi. Pamukkale Üniversitesi, Sosyal Bilimler Enstitüsü, Doktora, Denizli, 210s.

LANE, N. D., MILUZZO, E., LU, H., PEEBLES, D., CHOUDHURY, T. And CAMPBELL, A.T., 2010. A survey of mobile phone sensing. IEEE Communications Magazine, 48:140–150.

LEE M., and CHO D., 2014. Logmusic: context-based social music recommendation service on mobile device. Proceedings of the ACM International Joint Conference on Pervasive and Ubiq- uitous Computing (UbiComp ’14), September 2014, Seattle, Wash, USA, 95–98.

LİANG, G., CAO, J. and ZHU,W., 2013. CircleSense: a pervasive comput- ing system for recognizing social activities. Proceedings of the 11th IEEE International Conference on Pervasive Computing and Communications (PerCom ’13), March 2013, IEEE, San Diego, Calif, USA, 201–206.

LÓPEZ, I., 2005. ITU Internet Report 2005 The Internet of Things. International Telecominication Union, Geneva, 28s.

MANNA, S., BHUNIA, S. and MUKHERJEE, N., 2014. Vehicular pollution monitoring using IoT. International Conference on Recent Advances and Innovations in Engineering, (ICRAIE-2014), 9-11 May 2014, Jaipur, India, 5s.

MCGRATH, M. J. and SCANAILL C. N., 2013. Body-worn, ambient, and consumer sensing for health applications. Sensor Technologies, Springer, 181–216.

MIORANDI, D., SICARI, S., PELLEGRINI, F. D. and CHLAMTAC, I., 2012. Internet of things: Vision, applications and research challenges. Ad Hoc Netw., 10:1497–1516.

MUTLU, Ö., 1993. Tek Makine Çizelgeleme Probleminde En İyi Teslim Zamanlarının Belirlenmesi. Osman Gazi Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, Eskişehir, 76s.

PERERA, C., ZASLAVSKY, A., CHRISTEN, P. and GEORGAKOPOULOS, D., 2014. Context aware computing for the internet of things: A survey. IEEE COMMUNICATIONS SURVEYS & TUTORIALS, First Quarter, 16:414-452.

PİNEDO, M. L., 2008. Scheduling Theory Algorithms and Systems, Springer, USA, 622s.

ROBLES, R. J., KIM, T. H., COOK, D. and DAS, S., 2010. A review on security in smart home development. International Journal of Advanced Science and Technology, 15:.13-22.

SCHMIDT, A. and LAERHOVEN, K., 2001. How to build smart appliances?. IEEE Personal Communications, August 2001, 10.1109/98.944006.

SETHİ, P. and SARANGİ, S. R., 2017. Internet of Things Architectures, Protocols, and Applications. Journal of Electrical and Computer Engineering, 2017 (9324035): 5-30.

SIXSMITH, A. and JOHNSON, N., 2004. A smart sensor to detect the falls of the elderly. IEEE Pervasive Computing, 3: 42– 47.

TRSTENJAKA M. and COSİCA P., 2017. Process planning in Industry 4.0 environment. 27th International Conference on Flexible Automation and Intelligent Manufacturing, FAIM2017, 27-30 June 2017, Modena, Italy, 1744 – 1750.

TIAN S., TAIYONG W., ZHANG L., WU X., 2019. The Internet of Things enabled manufacturing enterprise information system design and shop floor dynamic scheduling optimisation, Enterprise Information Systems, DOI: 10.1080/17517575.2019.1609703

TUSIAD, 2016. Türkiye'deki Dijital Değişime CEO Bakışı, https://tusiad.org/tr/yayinlar/raporlar/item/8867-tu-rkiye-deki-dijital-deg-is-ime-ceo-bakis-i-raporu 05.05.2018

VERMESAN, O., FRIESS P., 2011. Internet of Things Global Global Technological and Societal Trends. River Publishers, Denmark, 297s.

WAN, J. and JONES, J., 2013. Managing IT service management implementation complexity from the perspective of the Warfield version of systems science. Enterp. Inf. Syst., 7: 490–522.

WANG, L., LUO, C, and CAI, J., 2017. A Variable Interval Rescheduling Strategy for Dynamic Flexible Job Shop Scheduling Problem by Improved Genetic Algorithm. Hindawi- Journal of Advanced Transportation, 2017: 1527858, 12s.

WANG, R., CHEN, F. and CHEN Z., 2014. Studentlife: assessing mental health, academic performance and behavioral trends of college students using smartphones. Proceedings of the ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp '14), September 2014, Seattle, Wash, USA, 3–14.

WANG, S., ZHANG, Z., YE, Z., WANG, X., LİN, X. and CHEN, S., 2013. Application of environmental internet of things on water quality management of urban scenic river. International Journal of Sustainable Development & World Ecology, 20:216–222.

WOODFORD, C., 2019. Accelerometers, https://www.explainthatstuff.com/accelerometers.html 14.02.2018

WU, Z., HE, H. and HUANG, C., 2015. Flexible job shop dynamic scheduling problem research with machine fault [J]. Machine Design and Research, 3:31 94–98.

XU, L., HE, W. and LI, S., 2014. Internet of Things in Industries: A Survey. IEEE Transactıons on Industrıal Informatıcs, 10: 2233-2243.

XU, L., 2011. Enterprise Systems: State-of-the-art and future trends. IEEE Trans. Ind. Informat, Nov, 7: 630–640.

YAACOUB, E., KADRİ, A. and ABU-DAYYA, A., 2012. Cooperative wireless sensor networks for green internet of things. Proceedings of 8th ACM Symp. QoS Security Wireless Mobile Netw., Paphos, Cyprus, 79–80.

YU, M., RHUMA, A., NAQVI, S. M., WANG, L. and CHAMBERS, J., 2012. A posture recognition-based fall detection system for monitoring an elderly person in a smart home environment. IEEE Transactions on Information Technology in Biomedicine, 16:1274–1286.

ZHAO, C., ZHANG, F., FENG, Y. and GUO, X., 2010. The study and application of the IOT technology in agriculture. Proceedings of the 3rd IEEE International Conference on Computer Science and Information Technology (ICCSIT ’10), July 2010, Chengdu, China, 462–465.

ZHONG, X. and LIANG, Y., 2016. Raspberry Pi: An Effective Vehicle in Teaching the Internet of Things in Computer Science and Engineering. Electronics, 5:56.

ÖZGEÇMİŞ

KİŞİSEL BİLGİLER

Adı Soyadı : Mehmet Emin AYDOĞDU
Uyruğu : T.C.
Doğum Yeri ve Tarihi : Şanlıurfa – 20/07/1984
e-mail : meayd@harran.edu.tr

EĞİTİM

Derece	Adı, İlçe, İl	Bitirme Yılı
Lise	: Orhan Gazi Lisesi, Merkez / Şanlıurfa	2002
Üniversite	: Sakarya Üniversitesi Mühendislik Fakültesi Bilgisayar Mühendisliği	2009
Yüksek Lisans	: Harran Üniversitesi Fen Bilimleri Enstitüsü Elektrik Elektronik Mühendisliği A.B.D	-

İŞ DENEYİMLERİ

Yıl	Kurum	Görevi
2000 - 2004	İnci Bilgisayar Yazılım Donanım ve İnternet Hiz. Tic.Ltd.Şti.	Teknik Servis
2009 - 2010	Avukat Pro Yazılım Firması	Yazılım Uzmanı
2010 - 2015	Şanlıurfa Belediyesi	Mühendis
2010 - 2015	IMG HASAR DANIŞMANLIK	Yazılım Uzmanı
2015 - --	Harran Üniversitesi Bilgi İşlem Daire Bşk.	Yazılım Uzmanı
2015 - --	Şanlıurfa Teknik Bilimler MYO	Öğr. Görevlisi

UZMANLIK ALANI

Yazılım Geliştirme, Nesnelerin İnterneti, Gömülü Sistemler

YABANCI DİLLER

İngilizce

YAYINLAR

AYDOĞDU, M.E., BEŞLİ, N., KAYA, S. 2018, **"**Çizelgeleme Problemlerinin Nesnelerin İnterneti Açısından İncelenmesi ve Geliştirilmesi", 6th International GAP Engineering Conference, GAP2018, Şanlıurfa. (Tam Metin Bildiri/Sözlü Sunum – Yayın No:4735740).

EKLER

EK 1: 15x10x56 Boyutlu DEATÇ (Ex4) problemi makine işlem süreleri

		Makineler									
İşler	Operasyonlar	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10
J1	O11	1	4	6	9	3	5	2	8	9	4
	O12	1	1	3	4	8	10	4	11	4	3
	O13	2	5	1	5	6	9	5	10	3	2
	O14	10	4	5	9	8	4	15	8	4	4
J2	O21	4	8	7	1	9	6	1	10	7	1
	O22	6	11	2	7	5	3	5	14	9	2
	O23	8	5	8	9	4	3	5	3	8	1
	O24	9	3	6	1	2	6	4	1	7	2
J3	O31	7	1	8	5	4	9	1	2	3	4
	O32	5	10	6	4	9	5	1	7	1	6
	O33	4	2	3	8	7	4	6	9	8	4
	O34	7	3	12	1	6	5	8	3	5	2
J4	O41	6	2	5	4	1	2	3	6	5	4
	O42	8	5	7	4	1	2	36	5	8	5
	O43	9	6	2	4	5	1	3	6	5	2
	O44	11	4	5	6	2	7	5	4	2	1
J5	O51	6	9	2	3	5	8	7	4	1	2
	O52	5	4	6	3	5	2	28	7	4	5
	O53	6	2	4	3	6	5	2	4	7	9
	O54	6	5	4	2	3	2	5	4	7	5
J6	O61	4	1	3	2	6	9	8	5	4	2
	O62	1	3	6	5	4	7	5	4	6	5
J7	O71	1	4	2	3	3	6	9	8	5	4
	O72	2	1	4	5	2	3	5	4	2	5
J8	O81	2	3	6	2	5	4	1	5	8	7
	O82	4	5	6	2	3	5	4	1	2	5
	O83	3	5	4	2	5	49	8	5	4	5
	O84	1	2	36	5	2	3	6	4	11	2
J9	O91	6	3	2	22	44	11	10	23	5	1
	O92	2	3	2	12	15	10	12	14	18	16
	O93	20	17	12	5	9	6	4	7	5	6
	O94	9	8	7	4	5	8	7	4	56	2
J10	O101	5	8	7	4	56	3	2	5	4	1
	O102	2	5	6	9	8	5	4	2	5	4
	O103	6	3	2	5	4	7	4	5	2	1
	O104	3	2	5	6	5	8	7	4	5	2
J11	O111	1	2	3	6	5	2	1	4	2	1
	O112	2	3	6	3	2	1	4	10	12	1
	O113	3	6	2	5	8	4	6	3	2	5
	O114	4	1	45	6	2	4	1	25	2	4
J12	O121	9	8	5	6	3	6	5	2	4	2
	O122	5	8	9	5	4	75	63	6	5	21
	O123	12	5	4	6	3	2	5	4	2	5
	O124	8	7	9	5	6	3	2	5	8	4
J13	O131	4	2	5	6	8	5	6	4	6	2
	O132	3	5	4	7	5	8	6	6	3	2
	O133	5	4	5	8	5	4	6	5	4	2
	O134	3	2	5	6	5	4	8	5	6	4
J14	O141	2	3	5	4	6	5	4	85	4	5
	O142	6	2	4	5	8	6	5	4	2	6
	O143	3	25	4	8	5	6	3	2	5	4
	O144	8	5	6	4	2	3	6	8	5	4
J15	O151	2	5	6	8	5	6	3	2	5	4
	O152	5	6	2	5	4	2	5	3	2	5
	O153	4	5	2	3	5	2	8	4	7	5
	O154	6	2	11	14	2	3	6	5	4	8