

T.C.
İSTANBUL AYDIN ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ



**MQTT PROTOKOLÜ İLE ÇALIŞAN NESNELERİN İNTERNETİ
CİHAZLARIN FABRİKA ÜRETİM HATLARINA ENTEGRASYONU**

YÜKSEK LİSANS TEZİ

Ulaş UZUNKAYA

Elektrik - Elektronik Mühendisliği Anabilim Dalı
Elektrik - Elektronik Mühendisliği Programı

Mayıs 2021

T.C.
İSTANBUL AYDIN ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ



**MQTT PROTOKOLÜ İLE ÇALIŞAN NESNELERİN İNTERNETİ
CİHAZLARIN FABRİKA ÜRETİM HATLARINA ENTEGRASYONU**

YÜKSEK LİSANS TEZİ

Ulaş UZUNKAYA
(Y1713.100008)

Elektrik - Elektronik Mühendisliği Anabilim Dalı
Elektrik - Elektronik Mühendisliği Programı

Tez Danışmanı: Dr. Öğr. Üyesi Necip Gökhan KASAPOĞLU

Mayıs 2021

ONAY FORMU

Tez çalışmamı annem, babam ve eşime adıyorum ...

ÖNSÖZ

Bu bitirme çalışmasında, MQTT Protokolü ile çalışan IoT cihazların endüstriyel üretim hatlarına entegrasyonu ve kestirimci bakım çalışmaları üzerinde yaptığım araştırmalar sonucunda elde ettiğim bilgi ve deneyimleri sunmaktayım.

Bu çalışmayı hazırlarken geçirdiğim süreçte benden desteklerini esirgemeyen ENTES Elektronik A.Ş. çalışma arkadaşlarıma, ayrıca manevi desteğini her an yanımda hissettiğim aileme ve eşime, bana bu tez çalışması ile kendimi daha da geliştirmeme katkı sağlayan değerli hocam Dr. Öğr. Üyesi Necip Gökhan KASAPOĞLU'na teşekkürü bir borç bilirim.

Mayıs 2021

Ulaş UZUNKAYA

İÇİNDEKİLER LİSTESİ

	Page
1. GİRİŞ	1
1.1 Çalışma Konusu	1
1.2 Tezin Amacı	1
1.3 Literatür İncelemesi.....	1
1.4 Hipotez	3
2. MQTT TEMELLERİ	5
2.1 MQTT Tanıtımı.....	5
2.2 Publish / Subscribe Yapısı.....	5
2.3 Client ve Broker Bağlantısı	6
2.4 MQTT Kontrol Paket Yapısı	7
2.4.1 Sabit Başlık (<i>Fixed Header</i>).....	7
2.4.1.1 Mesaj Tipi	7
2.4.1.2 Başlık Bayrakları	8
2.4.2 Değişken Başlık (<i>Variable Header</i>)	9
2.4.3 Yük (<i>Payload</i>).....	9
2.5 MQTT Topic Yapısı	9
2.5.1 Joker Karakterler (<i>Wildcards</i>).....	10
2.5.1.1 Tek Düzeyli Joker Karakter: “+”	10
2.5.1.2 Çok Düzeyli Joker Karakter: “#”	10
2.6 QoS Düzeyleri	11
2.6.1 QoS 0.....	12
2.6.2 QoS 1.....	12
2.6.3 QoS 2.....	12
2.7 Kalıcı Oturum ve Mesaj Sıralama	13
2.8 Retained Messages	14
2.9 Last Will and Testament.....	14
2.10 Keep Alive ve Client Take Over.....	15
2.10.1 Keep Alive	15
2.10.2 Client Take Over	15
3. BÜYÜK VERİ “BIG DATA”	17

3.1 Büyük Veri Nedir?	17
3.2 Büyük Veri Bileşenleri (3V)	17
3.2.1 Hacim (Volume)	18
3.2.2 Hız (Velocity)	18
3.2.3 Çeşitlilik (Variety)	18
3.3 Büyük Verinin Önemi	19
3.4 Büyük Veri Nereelerde Kullanılır?	19
3.5 Büyük Veri Nasıl Oluşur?	20
3.6 Yapılandırılmamış Veri (Raw Data) Nedir?	20
4. KESTİRİMCİ BAKIM ÇALIŞMALARI	21
4.1 Kestirimci Bakım Nedir?	21
4.2 Makine Öğrenmesi ile Kestirimci Bakım	21
4.2.1 Support Vector Machines (SVM)	21
4.2.2 Support Vector Regression (SVR)	22
4.2.3 K-Nearest Neighbors (KNN)	23
5. DENEYİN YAPILIŞI	24
5.1 Giriş	24
5.2 İş İçeriği	24
5.2.1 Geliştirme Ortamı	24
5.2.2 Kullanılan Veri Kümeleri	24
5.3 Veri Kümeleri İle Yapılan Çalışmalar	25
5.4 Spektral Analiz Yöntemleri	36
5.5 Rulman Arıza Frekansları	37
5.5.1 Ball Pass Frequency Inner (BPFI)	37
5.5.2 Ball Pass Frequency Outer (BPFO)	37
5.5.3 Ball Spin Frequency (BSF)	37
5.5.4 Fundamental Train Frequency (FTF)	37
5.6 Yapılan Çalışmalar	38
5.6.1 Frekans Alanında Çalışmalar	38
5.6.2 Zaman Alanında Çalışmalar	44
5.6.2.1 Erken Evre	46
5.6.2.2 Normal Evre	47
5.6.2.3 Şüpheli Evre	48
5.6.2.4 İç Yatak Arızası	49
6. SONUÇLAR ve KARŞILAŞTIRMALAR	50

6.1 Literatür Karşılaştırması	51
7. İLERİDE YAPILMASI PLANLANAN ÇALIŞMALAR.....	53
REFERANSLAR.....	55
ÖZGEÇMİŞ.....	59

KISALTMALAR

IoT	: Internet of Things
IIoT	: Industrial Internet of Things
MQTT	: Message Queuing Telemetry Transport
QoS	: Quality of Service
TCP	: Transmission Control Protocol
IP	: Internet Protocol
SSL	: Secure Sockets Layer
TLS	: Transport Layer Security
PUBREC	: Publish Received
PUBACK	: Publish Acknowledgement
PUBREL	: Publish Release
PUBCOMP	: Publish Complete
SUBACK	: Subscribe Acknowledgement
UNSUBACK	: Unsubscribe Acknowledgement
PINGREQ	: PING Request
PINGRESP	: PING Response
LWT	: Last Will and Testament
BI	: Business Intelligence
SVM	: Support Vector Machines
KNN	: K nearest neighbors
SVR	: Support Vector Regression

RUL	: Remaining Useful Life
KNN	: K nearest neighbors
BPFO	: Ball Pass Frequency Outer
BPFI	: Ball Pass Frequency Inner
BSF	: Ball Spin Frequency
FTF	: Fundamental Train Frequency
FFT	: Fast Fourier Transform
RMS	: Root Mean Square
DNN	: Deep Neural Networks

TABLO LİSTESİ

Tablo 2.1: MQTT Kontrol Paket Yapısı	7
Tablo 2.2: MQTT Sabit Başlık Yapısı	7
Tablo 2.3: Mesaj Tipleri	8
Tablo 2.4: Başlık Bayrakları	8
Tablo 2.5: QoS Değerleri	9
Tablo 2.6: “ev/giriskat/+/sicaklik” Konu Mesajları	10
Tablo 2.7: “ev/giriskat/#” Konu Mesajları	11
Tablo 5.1: 1.Rulman Evreleri	26
Tablo 5.2: 2.Rulman Evreleri	26
Tablo 5.3: 3.Rulman Evreleri	27
Tablo 5.4: 4.Rulman Evreleri	27
Tablo 5.5: Rulman Özellikleri	27
Tablo 5.6: Karakteristik Frekanslar	38
Tablo 5.7: Teori ve Ölçüm Farkları	43
Tablo 5.8: Algoritma Başarı Oranı	44

ŞEKİLLER LİSTESİ

Şekil 2.1: MQTT Pub/Sub Yapısı.....	6
Şekil 2.2: MQTT QoS Seviyeleri	13
Şekil 3.1: Büyük Veri Bileşenleri.....	18
Şekil 3.2: Büyük Veri Kullanım Alanları	20
Şekil 4.1: Destek Vektör Makineleri (SVM).....	22
Şekil 4.2: Destek Vektör Regresyonu (SVR)	23
Şekil 4.3: KNN Sınıflandırma Örneği	23
Şekil 5.1: Deney Kurulumu	25
Şekil 5.2: Rulman1 Erken Evre	28
Şekil 5.3: Rulman1 Normal Evre	28
Şekil 5.4: Rulman1 Şüpheli Evre	29
Şekil 5.5: Rulman1 Muhtemel Arıza Başlangıcı	29
Şekil 5.6: Rulman2 Erken Evre	30
Şekil 5.7: Rulman2 Normal Evre	30
Şekil 5.8: Rulman2 Şüpheli Evre	31
Şekil 5.9: Rulman2 Muhtemel Arıza Başlangıcı	31
Şekil 5.10: Rulman3 Erken Evre	32
Şekil 5.11: Rulman3 Normal Evre	32
Şekil 5.12: Rulman3 Şüpheli Evre	33
Şekil 5.13: Rulman3 Muhtemel Arıza Başlangıcı	33
Şekil 5.14: Rulman4 Erken Evre	34
Şekil 5.15: Rulman4 Normal Evre	34
Şekil 5.16: Rulman4 Şüpheli Evre	35
Şekil 5.17: Rulman4 Bilye Arızası	35
Şekil 5.18: Rulman4 2. Aşama Arıza	36
Şekil 5.19: Deney Başlangıcında 3. Rulman	39
Şekil 5.20: 3. Rulman Normal Evre.....	40
Şekil 5.21: 3. Rulman Arıza Şüphesi.....	41
Şekil 5.22: 3. Rulman Arıza Durumu	42
Şekil 5.23: 3. Rulman İvme-Zaman Grafiği	45
Şekil 5.24: a) Dengesizlik b) Yatak Hasarı c) Vuruş d) Gevşeklik	45
Şekil 5.25: Erken Evre a-t Grafiği	46
Şekil 5.26: Normal Evre a-t Grafiği	47
Şekil 5.27: Arıza Şüphesi a-t Grafiği.....	48
Şekil 5.28: Rulman Arıza Durumunda a-t Grafiği.....	49

INTEGRATION OF IoT DEVICES WITH MQTT PROTOCOL TO FACTORY PRODUCTION LINES

ABSTRACT

Today, it is possible to encounter internet usage everywhere in our lives. We meet all our needs over the internet, from bilateral communication to intercontinental information sharing. The concept called Internet of Things (IOT), which is the Internet of Things, is the name given to the situation where smart devices and machines are connected with each other or with systems and communicate with each other. With the Internet of Things, it is possible for us to monitor all events in the environment by using various sensors. These sensors can communicate wired or wirelessly, depending on the area of use. With the integration of this system into production lines in factories; In the environment, live tracking of lines can be made with real-time data coming from sensors in machines and devices. These incoming data are stored in cloud storage and recorded in one or more databases. These recorded data can be used for real-time tracking of production lines as well as for big data analysis. Big data analysis is the analysis of data collected in a database and transforming it into a processable and meaningful form. By using the Internet of Things in production lines, the data collected from the sensors are evaluated by prepared algorithms. Thus, by minimizing human errors; It is aimed to increase the life and productivity of production machines and to ensure continuity of production by preventing unplanned and accidental stops. In this study, studies on the integration of the Internet of Things to production lines in factories have been examined and carried out. Studies and researches on database selection, sensors to be used, data to be analyzed and preventive and predictive maintenance have been conducted.

Keywords: *MQTT, Predictive Maintenance, Internet of Things, Internet of Things in manufacturing, Industrial Internet of Things*

MQTT PROTOKOLÜ İLE ÇALIŞAN NESNELERİN İNTERNETİ CİHAZLARIN FABRİKA ÜRETİM HATLARINA ENTEGRASYONU

ÖZET

Günümüzde internet kullanımına hayatımızın her yerinde rastlamak mümkündür. İkili iletişimden kıtalar arası bilgi paylaşımına kadar tüm gereksinimlerimizi internet üzerinden karşılamaktayız. Internet of Things (IOT) yani Nesnelerin İnterneti olarak adlandırılan kavram ise, akıllı cihazların ve makinelerin birbirleriyle veya sistemlerle bağlantılı olup, birbirleriyle iletişim içerisinde olduğu duruma verilen addır. Nesnelerin İnterneti ile, çeşitli sensörler kullanarak, bulunulan çevre içerisindeki tüm olayları izlememiz mümkün olabilmektedir. Bu sensörler kullanım alanına göre kablolu veya kablosuz olarak haberleşebilmektedir. Bu sistemin fabrikalarda üretim hatlarına entegrasyonu ile; ortamda, makine ve cihazlarda bulunan sensörlerden gelen gerçek zamanlı veriler ile hatların canlı takibi yapılabilmektedir. Gelen bu veriler, bulut depolama ile depolanır ve bir veya birden fazla veri tabanına kayıt edilir. Kayıt edilen bu veriler, üretim hatlarının gerçek zamanlı takibinin yanı sıra büyük veri analizi yapmak için de kullanılabilir. Büyük veri analizi, bir veri tabanında toplanan verilerin analiz edilip, işlenebilir ve anlamlı hale dönüştürülmesidir. Nesnelerin internetinin üretim hatlarında kullanılmasıyla, sensörlerden toplanan verilerin hazırlanmış algoritmalar tarafından değerlendirilmesi sağlanmaktadır. Böylece insan hatalarını en aza indirerek; üretim makinelerinin ömürlerini ve verimliliklerini arttırmak, plansız ve arıza duruşların önüne geçerek üretim sürekliliğini sağlamak hedeflenmektedir. Bu çalışmada nesnelerin internetinin fabrikalarda bulunan üretim hatlarına entegrasyonu ile ilgili çalışmalar incelenmiş ve gerçekleştirilmiştir. Veri tabanı seçimi, kullanılacak sensörler, analizi yapılacak verilerin seçimi ve önleyici ve kestirimci bakımlar hakkında çalışmalar ve araştırmalar yapılmıştır.

Anahtar Kelimeler: *MQTT, Kestirimci Bakım, Nesnelerin İnterneti, Üretimde Nesnelerin İnterneti, Endüstriyel Nesnelerin İnterneti*

1. GİRİŞ

1.1 Çalışma Konusu

Tez için seçilen konu, enerji verimliliğinin artırılması ve üretim kayıplarının azaltılması açısından önemlidir. Bu konu, fabrikalarda bulunan üretim hatları ve makinelerin farklı parametreleri hakkında veriler toplayarak bakım ve proses takibi açısından çalıştığım bir konudur.

Tezde ele alacağım konu ve kavramlar şu şekildedir: MQTT, Broker, Publish, Subscribe, QoS, IoT, Endüstri 4.0, büyük veri analizi, veri tabanı ve sensörler.

1.2 Tezin Amacı

Fabrikalarda mevcut üretim hatlarına Nesnelerin İnterneti (IoT) cihazlarının adaptasyonu ile üretim verimliliğinin artması, hatasız üretimin devamlılık sağlaması, daha çevreci ve kaynak dostu üretimin geliştirilmesi, hatların izlenmesi ve oluşabilecek arızaların önceden tahmin edilerek plansız duruşların engellenmesi ve insan hatalarının azaltılması hedeflenmektedir.

Bu çalışmada, üretim makineleriyle entegre şekilde çalışabilecek IoT cihazlar tasarlanacak ve çeşitli sensör ve yazılımlar vasıtasıyla oluşabilecek arızalar önceden tahmin edilmeye çalışılacak ve kestirimci bakım çalışmaları yapılacaktır.

1.3 Literatür İncelemesi

Nesnelerin interneti kavramı ilk defa 1999 yılında Kevin Ashton tarafından P&G firmasında yaptığı bir sunum sırasında kullanılmıştır. Kablosuz haberleşme teknolojilerinin son yıllarda hızla gelişmesiyle birlikte nesnelerin interneti kullanımı gün geçtikçe artmaktadır. Günümüzde bu sistemlerde internet alt yapısı kullanılarak çeşitli sensörler üzerinden gelen kesintisiz ve gerçek zamanlı veriler, taşımacılıkta, güvenlik sistemlerinde, trafik verilerinin takibi ve bunun gibi alanlarda kullanılmaktadır. Bu cihazlardan gelen verileri saklayabilmek için (big data) çok yüksek kapasiteli depolama alanlarına ihtiyaç vardır. Bu verileri anlamlı hale

getirecek yazılımlara, işlenmiş verilere ulaşılabilecek kullanıcı arayüzlerine ihtiyaç vardır. Bu arayüzler web sitesi, mobil aplikasyon veya servis sağlayıcı olabilir.

Nesnelerin interneti (IoT) cihazları için internet bağlantısı bir gerekliliktir. İnternet bağlantısı, cihazların birbirleriyle ve back-end hizmetleriyle çalışmasına izin verir. MQTT protokolü de IoT’de kullanılan protokollerden biridir. TCP / IP yığını üzerine inşa edilen MQTT, cihazlar arasındaki tüm veri iletişiminin şifreli ve güvenli olmasını sağlamak için TCP / IP üzerine kurulu güvenli bir protokol olan SSL / TLS üzerinde de çalışabilmektedir.

[1]’de bahsedildiği üzere, MQTT bir yayınlama ve abone olma şeklinde çalışan istemci – sunucu mesajlaşma protokolüdür. Bu protokol basit, açık, hafif ve uygulanması kolay olacak şekilde tasarlanmıştır. TCP / IP veya diğer ağ protokolleri üzerinden çalışır ve sıralı, kayıpsız ve çift yönlü bağlantı sağlanmaktadır. Bu protokol, mesaj teslimatı için QoS0, QoS1 ve QoS2 olmak üzere üç hizmet kalitesine sahiptir (Quality of Service). Bu hizmet kalitesi seçenekleri diğer IoT protokolleri arasında benzersizdir. MQTT protokolü, ağ trafiğini azaltmak için taşıma yükünü ve protokol değişimini en aza indirmekle kalmaz, aynı zamanda beklenmedik bir bağlantı kesilmesi meydana geldiğinde ilgilenen istemcileri haberdar etmek için özel bir mekanizmaya da sahiptir.

[2]’de anlatıldığı gibi, MQTT protokolü abone, yayın ve mesaj aracısından (broker) oluşur. Belirli bir konuya ait olan mesajı almak isteyen bir istemci, konuya bir broker vasıtasıyla abone olur. Başka bir istemci mesajı belirli bir konu ile yayınlarsa, bir broker mesajı belirli konuya halihazırda abone olan istemcilere yeniden yayımlar. Konu, her istemci için mesajları filtrelemek için bir mesaj dizisidir ve bir veya daha fazla konu düzeyinden oluşur. Her seviye eğik çizgi “/” ile ayrılır.

[3]’e göre MQTT, bugün IoT tarafından kullanılan en popüler ikinci IoT mesajlaşma protokolüdür. MQTT’nin bir dezavantajı ise, keşif yeteneklerine sahip olmamasıdır. Bu, kablosuz sensörler bağlamında bir sorun yaratmamaktadır, çünkü IIoT senaryolarının çoğu, tüm düğümlerin özelliklerinin zaten bilindiğini veya ilgili uç yönlendiriciden kolayca keşfedilebileceğini varsayar. MQTT’nin diğer dezavantajı, senkronize bir iletişim protokolü olmamasıdır. Bu düğümler arasında senkronize iletişim için özel bir ihtiyaç olmadığından, bir ağ geçidinde algılanan verileri periyodik olarak raporlayan IIoT sensörleri bağlamında bunun bir önemi yoktur,

ancak bazı durumlarda IIoT aktüatörleri ile iletişim kurulurken bir sorun olabilir. Bu durum abone olunan aktüatörlere iletilen ve daha sonra onları bir eylemi tetiklemeye zorlayan bir konu yayınlanarak üstesinden gelinebilir.

[4]'te bahsedildiği üzere, üretim endüstrisi daha rekabetçi hale gelmektedir. Rekabet arttıkça endüstriye son teknolojiyi uyarlamak, ciddi başarılarla ulaşmanın kesin yollarından biri olarak görülmektedir. IoT, teknolojik alanda son yıllarda oldukça revaçta olan bir terim. 2020 yılının sonunda, dünya çapında 22 milyara yakın aktif bağlı cihaz olduğu görülmektedir. Bu cihazların 12 milyara yakını nesnelerin interneti cihaz bağlantılarıdır. Bağlı cihazların önümüzdeki beş yıl içerisinde 30 milyara çıkması beklenmektedir. IoT, akıllı ağlarda yüzlerce akıllı cihazı birbirine bağlar ve üreticilerin proseslerini geliştirmelerine ve kurumsal kazanç sağlanmasına yardımcı olur.

Özellikle üretim sektörünün; yüksek verimli süreçlere, kontrol ve denetime ihtiyacı vardır. Endüstriyel Nesnelerin İnterneti (IIoT), endüstrinin üretim bilgisi, veri analizleri, otomasyon, uzaktan yönetim, enerji yönetimi, azaltılmış duruş süreleri, kestirimci bakımlar ve yüksek güvenilirlik gibi unsurları elde etmesini sağlar. IIoT çözümleri, akıllı sensör ve ağ geçitlerinden bulut ve akıllı analitiğe kadar tüm döngüyü kapsamaktadır. Bu sayede endüstrinin üretim süreçlerinin iyileşmesi ve artırılmasının yanı sıra dünya çapındaki müşterilerin farklı ihtiyaçlarını karşılayacak ürünlerin ortaya çıkmasına da olanak sağlamaktadır. IIoT, üretim sürecinin geliştirilmesi ve sektörü ileri başarı seviyelerine taşımak için büyük bir potansiyele sahiptir.

1.4 Hipotez

MQTT haberleşme protokolü kullanan IoT cihazları vasıtasıyla sensörlerden gelen veriler toplanacak ve toplanan verileri analiz eden bir yazılım geliştirilecektir. Toplanan bu bilgiler ve geliştirilen yazılım ile oluşabilecek arızalar önceden tahmin edilecek ve kestirimci bakım çalışmaları yapılmış olacaktır. Kestirimci bakım, makine veya çalışan bazı ekipmanların, çeşitli sensörler vasıtasıyla titreşim ve sıcaklık gibi bazı verilerinin işlenmesiyle oluşabilecek çeşitli arızaların önceden belirlenmesi ve buna bağlı olarak gerekli bakım işlemlerinin yapılmasıdır. Uzaktan algılama ve kontrol için uygun olması, güvenlik seviyelerinin olması ve diğer protokollere kıyasla düşük ağ bant genişliğinde daha fazla veri transfer edilebilmesi

sebebi ile MQTT protokolü, kestirimci bakım çalışmalarında daha hızlı ve güvenilir veri alışverişi sağlayabilir.

2. MQTT TEMELLERİ

2.1 MQTT Tanıtımı

Nesnelerin İnterneti (IoT) cihazlarının birbirleriyle haberleşebilmeleri ve arkayüz hizmetleriyle çalışabilmeleri için mutlaka internete bağlanmaları gerekmektedir. Bilindiği üzere, internetin temeli TCP / IP ağ protokolüdür. Message Queuing Telemetry Transport (MQTT) protokolü de TCP / IP ağ protokolü üzerine inşa edilmiştir ve IoT hizmetleri için bir standart haline gelmiştir. Veri aktarımının güvenli ve şifreli olmasının istendiği durumlarda MQTT, SSL / TLS üzerinde de çalışabilmektedir.

MQTT ilk olarak 90'lı yılların sonlarında IBM tarafından keşfedilmiştir. Bu protokolün ilk olarak, petrol boru hatlarındaki sensörlerin uydulara bağlanması için kullanılmıştır. Bu protokol, taraflar arasında asenkron mesajlaşmayı destekleyen bir iletişim protokolüdür. Asenkron bir mesajlaşma protokolü olduğu için, mesajı gönderen ve alıcı alan ve zamanda birbirinden ayrıştırılmış durumdadır. Bu sayede, güvensiz ağ ortamlarında ölçeklenebilmektedir.

İsminin aksine mesajlaşma kuyruklarıyla bir ilgisi bulunmamaktadır, yayınlama / abone olma modeli (Publish/Subscribe) kullanmaktadır. Kullanılan çoğu programlama dillerinde açık kaynak uygulamaları ile desteklenmektedir.

2.2 Publish / Subscribe Yapısı

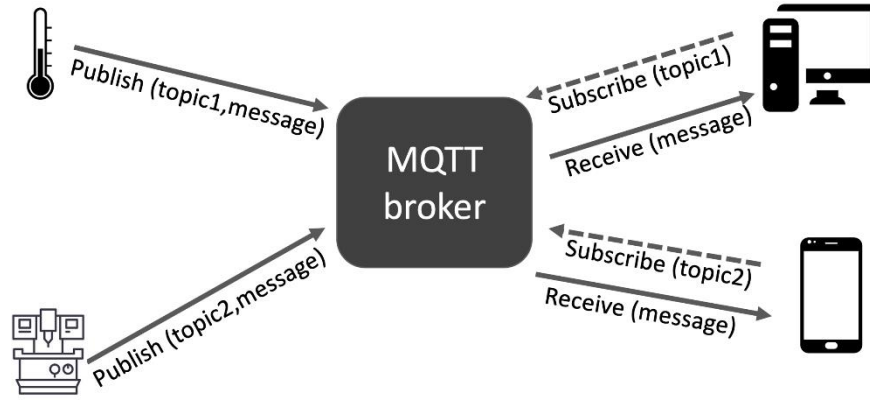
MQTT protokolünde mesajı yayınlayan yayıncı ve konulara abone olan istemcilerin oluşturduğu modele Publish/Subscribe (Pub/Sub) modeli denir. Bu yapıda istemciler, ilgili oldukları konulara abone olurlar ve abone oldukları konularda yayınlanan tüm mesajları alırlar.

Pub/Sub modelinde, mesaj gönderen (yayıncı) istemcilerden mesajları alan istemciler (aboneler) ayrılır. Bu ayrıştırma birkaç boyuttan oluşur:

- Alan Ayrışması: Yayıncı ve abonelerin birbirlerini bilmesine gerek yoktur.

- Zaman Ayrışması: Yayıncı ve abonelerin aynı anda çalışmalarına ihtiyaç yoktur.
- Senkronizasyon Ayrışması: İki bileşendeki işlemlerin yayınlama ve alma sırasında kesilmesine gerek yoktur.

Yayıncılar ile aboneler asla birbirleriyle doğrudan iletişim kurmazlar, hatta birbirlerinin varlığından haberleri yoktur. Yayıncı ile aboneler arasındaki bağlantı üçüncü bir bileşen tarafından sağlanır. Bu bileşen Broker olarak adlandırılır. Broker'ın görevi, gelen tüm mesajları filtrelemek ve abonelere doğru şekilde dağıtmaktır (Şekil 2.1).



Şekil 2.1: MQTT Pub/Sub Yapısı

2.3 Client ve Broker Bağlantısı

MQTT protokolü TCP / IP paketiyle çalışmaktadır. Hem istemcilerin hem de Broker'ın TCP / IP yığınına sahip olması gerekmektedir.

MQTT bağlantıları, her zaman istemci ile Broker arasındadır. İstemciler hiçbir zaman birbirleriyle doğrudan bağlantı kuramazlar. İstemci bir bağlantı başlatmak istediği zaman Broker'a CONNECT mesajı gönderir. Broker, CONNACK mesajı ve durum kodu göndererek istemciye yanıt verir. Bağlantı sağlandıktan sonra, istemci bağlantı kesme komutu gönderene kadar veya bağlantı kesilene kadar Broker bağlantıyı açık tutar.

Eğer istemci, Broker'a bağlantı başlatmak için hatalı biçimlendirilmiş bir CONNECT mesajı gönderirse veya ağ soketi açılmasıyla CONNECT mesajının

gönderilmesi arasında çok fazla zaman geçerse Broker bu bağlantıyı kapatır. Böylece Broker, kendini yavaşlatabilecek kötü niyetli istemcileri engellemiş olur.

2.4 MQTT Kontrol Paket Yapısı

Tablo 2.1: MQTT Kontrol Paket Yapısı

Sabit Başlık, 2 bayt boyutunda (Fixed Header)	Değişken Başlık, değişken boyut (Variable Header)	Yük, değişken boyut (Payload)
--	--	----------------------------------

2.4.1 Sabit Başlık (Fixed Header)

Fixed Header yani sabit başlık, her MQTT paketinde her zaman mevcuttur. İki bayt uzunluğundadır. Sabit başlık yapısı aşağıda gösterilmiştir (Tablo 2.1).

Tablo 2.2: MQTT Sabit Başlık Yapısı

bit	0	1	2	3	4	5	6	7
1. Bayt	Başlık Bayrakları				Mesaj Tipi			
2. Bayt	Kalan Uzunluk							

2.4.1.1 Mesaj Tipi

Mesaj tipi, bağlantı isteği türünü belirtir. Mesaj tipinin her bir değeri farklı bağlantı talebini temsil eder ve 4 bit uzunluğundadır.

Tablo 2.3: Mesaj Tipleri

Mesaj Tipi	Değeri	Açıklama
Rezerve	0	Rezerve
CONNECT	1	İstemciden Broker'a bağlanma isteği
CONNACK	2	Bağlantı isteği kabulü belirtilir
PUBLISH	3	İstemci mesaj yayınlar
PUBACK	4	İstemcinin gönderdiği mesajın Broker tarafından onaylandığını belirtir
PUBREC	5	İstemcinin gönderdiği mesajın Broker tarafına iletildiğini belirtir
PUBREL	6	İstemcinin mesaj aldığını belirtir
PUBCOMP	7	İstemci ile Broker arasındaki haberleşme tamamlandı
SUBSCRIBE	8	İstemci Broker'a bir konuya bone olmak için mesaj gönderir
SUBACK	9	İstemciye mesajın ulaştığı bildirilir
UNSUBSCRIBE	10	İstemci Broker'a abone olduğu bir konudan ayrılmak için mesaj gönderir
UNSUBACK	11	İstemciye mesajın ulaşmadığı bildirilir
PINGREQ	12	İstemciden Broker'a haberleşme kontrolü için paket gönderilir
PINGRESP	13	Broker'dan istemciye haberleşme kontrolü için paket gönderilir
DISCONNECT	14	İstemci ile Broker arasındaki bağlantı sonlandırılır
Rezerve	15	Rezerve

2.4.1.2 Başlık Bayrakları

MQTT'de 3 başlık bayrağı vardır. Bunlar Retain, QoS ve DUP tur. Retain 1 bit, QoS 2 bit ve DUP 1 bittir (Tablo 2.3).

Tablo 2.4: Başlık Bayrakları

bit	0	1	2	3
Başlık Bayrakları	Retain	QoS	QoS	DUP

- DUP Flag

Yayınlanan kontrol paketinin yinelenen teslimini belirtir. DUP=0 ise, istemcilerin ilk defa MQTT yayınlama kontrol paketi göndermeye çalıştığı belirtilir. DUP=1 ise, istemcinin önceden gönderilen paketi tekrar göndermeye çalıştığını belirtir.

- QoS Seviyesi

QoS seviyeleri, mesaj tesliminin garantisini gösterir. 3 adet QoS seviyesi vardır ve 2 bit uzunluğundadır (Tablo 2.4).

Tablo 2.5: QoS Değerleri

QoS Değeri	Bit 1	Bit 2
0	0	0
1	1	0
2	0	1
Rezerve	1	1

- Retain

Eğer RETAIN=1 ise, Broker'ın depolanan pakette aynı konu başlığına sahip bir abone yoksa paketi tutacağı anlamına gelir. Abone olunur olunmaz, Broker depolanan paketi teslim edecektir. RETAIN=0 ise Broker'ın paketi tutmayacağı anlamına gelir.

Remaining length yani kalan uzunluk, değişken başlık ve yük dahil olmak üzere geçerli pakette kalan bayt sayısını belli eder.

2.4.2 Değişken Başlık (Variable Header)

Tüm MQTT kontrol paketlerinde mevcut değildir. Farklı MQTT istekleri için farklı yapıları vardır. PINGREQ, PINGRESP ve DISCONNECT paketleri haricindeki paketlerde geçerlidir.

2.4.3 Yük (Payload)

Yük, gönderilecek gerçek verilerdir ve tüm MQTT kontrol paketlerinde mevcut değildir. CONNECT, PUBLISH, SUBSCRIBE, SUBACK ve UNSUBSCRIBE paketlerinde kullanımı gereklidir.

2.5 MQTT Topic Yapısı

Topic, MQTT'de Broker'ın bağlı tüm istemciler için iletileri filtrelemek için kullandığı bir UTF-8 string dizisidir. Konular basit yapıda ve hafiftir.

Topic, bir veya daha fazla konu düzeyinden oluşmaktadır. Konu düzeyleri, eğik çizgi (/) ile ayrılır. İstemcilerin, mesaj yayınlamadan veya konuya abone olmadan önce istenen konuyu oluşturması gerekmemektedir.

Her konu en az bir harf içermelidir. Konular boşluklara ve küçük-büyük harflere duyarlıdır. Örneğin, “ev/giriskat/salon/sicaklik” ile “Ev/Giriskat/Salon/Sicaklik” veya “ev/giris kat/salon sicaklik” birbirinden farklı konulardır.

MQTT konuları, bir istisna haricinde, istenen her şekilde adlandırılabilir. \$ sembolü ile başlayan konular farklı bir amaçtır. \$ sembolü ile başlayan konular MQTT Broker’ın dahili istatistikleri için ayrılmıştır. İstemciler bu konularda mesaj yayınlamazlar.

2.5.1 Joker Karakterler (Wildcards)

Bir istemci bir konuya abone olacağı zaman, yayımlanan mesajın tam konusuna abone olabilir veya aynı anda birden fazla konuya abone olmak için Joker Karakterleri (Wildcards) kullanabilir. Joker karakterler sadece konulara abone olmak için kullanılabilir, mesaj yayınlamak için kullanılamazlar. Tek düzeyli (Single Level) ve Çok düzeyli (Multi Level) olmak üzere iki farklı joker karakter vardır.

2.5.1.1 Tek Düzeyli Joker Karakter: “+”

Tek düzeyli (Single Level) bir joker karakter, bir konu seviyesinin yerini alır. Artı simgesi (+), bir konudaki tek düzeyli joker karakteri temsil eder.

Örneğin “ev/giriskat+/sicaklik” konusuna abone olan bir istemcinin alabileceği ve alamayacağı mesajlar aşağıdaki tabloda belirtilmiştir (Tablo 2.6).

Tablo 2.6: “ev/giriskat+/sicaklik” Konu Mesajları

Mesaj Durumu	Konu
Mesaj Alınır	ev/giriskat/salon/sicaklik
Mesaj Alınır	ev/giriskat/mutfak/sicaklik
Mesaj Alınır	ev/giriskat/oda1/sicaklik
Mesaj Alınmaz	ev/giriskat/salon/ nem
Mesaj Alınmaz	ev/ kat1 /mutfak/sicaklik
Mesaj Alınmaz	ev/giriskat/mutfak/ buzdolabi /sicaklik

2.5.1.2 Çok Düzeyli Joker Karakter: “#”

Çok düzeyli (Multi Level) joker karakter, birçok konu seviyesini kapsar. Diyez işareti (#), konudaki çok düzeyli joker karakterini temsil eder. Broker’ın hangi

konuların eşleştiğini belirleyebilmesi için, çok düzeyli joker karakter, konudaki son karakter olmalı ve önünde eğik çizgi (/) olmalıdır.

Bir istemci çok düzeyli joker karakterli bir konuya abone olduğu zaman, konu ne kadar uzun olursa olsun, joker karakterden önceki modelle başlayan konuların tüm mesajlarını alır. Konu olarak sadece çok düzeyli joker karakterine abone olunursa (#), MQTT Broker'a gönderilen tüm mesajlar alınır. \$ sembolü ile başlayan konular ile yayınlanan mesajlar, istemci “#” konusuna abone olunduğu zaman dahi alınmazlar.

Örneğin “ev/giriskat/#” konusuna abone olan bir istemcinin alabileceği ve alamayacağı mesajlar aşağıdaki tabloda belirtilmiştir (Tablo 2.7).

Tablo 2.7: “ev/giriskat/#” Konu Mesajları

Mesaj Durumu	Konu
Mesaj Alınır	ev/giriskat/salon/sicaklik
Mesaj Alınır	ev/giriskat/mutfak/sicaklik
Mesaj Alınır	ev/giriskat/mutfak/buzdolabi/sicaklik
Mesaj Alınmaz	ev/kat1/mutfak/sicaklik
Mesaj Alınmaz	garaj/giriskat/ortam/sicaklik

2.6 QoS Düzeyleri

Quality of Service (QoS), yani hizmet kalitesi seviyesi, MQTT protokolünün temel bir özelliğidir. QoS, istemcilere ağ güvenilirliği ve uygulama mantığına göre bir hizmet düzeyi seçme şansı verir. Kısaca QoS, bir mesajın teslim garantisini sağlayan, gönderici ile alıcı arasındaki anlaşmadır.

MQTT’de QoS0, QoS1 ve QoS2 olmak üzere 3 QoS seviyesi bulunmaktadır. MQTT mesajların yeniden iletimini yönettiği ve mesaj teslimatını garanti ettiği için QoS, güvenilirmez ağlarda iletişimi çok daha kolay hale getirir.

Mesajı yayınlayan istemci, mesajı Broker’a gönderdiğinde mesajın QoS seviyesini de tanımlar. Broker bu mesajı, abone olan istemcilerin abonelik işlemi sırasında tanımladıkları QoS seviyelerini kullanarak abonelere iletir. Abone olan istemci, mesajı yayınlayan istemciden daha düşük bir QoS seviyesi ile abone olmuşsa, Broker, mesajı istemcilerin en düşük QoS seviyesi hangisiyse o kalitede yayınlar.

2.6.1 QoS 0

En düşük QoS seviyesi QoS 0 dır. Bu hizmet seviyesi en hızlı mesaj teslimatını sağlar fakat mesaj teslimatını garanti etmez. Alıcı, mesajı aldığını onaylamaz ve mesaj gönderici tarafından saklanmaz, yeniden iletilmez. QoS 0 seviyesi genellikle gönder ve unut (fire and forget) olarak adlandırılır. Mesaj en çok bir kez gönderilir.

2.6.2 QoS 1

QoS 1 seviyesi, bir mesajın alıcıya en az bir kez iletilmesini garanti eder. Gönderen, alıcıdan mesajın alındığını onaylayan bir PUBACK paketi alana kadar mesajı saklı tutar. Bu seviyede mesajın birden çok kez gönderilmesi veya teslim edilmesi mümkündür.

Gönderen, PUBLISH paketini karşılık gelen PUBACK paketiyle kıyaslamak için her pakette bulunan paket tanımlayıcısını kullanır. Gönderen yeterli süre içinde bir PUBACK paketi almazsa, PUBLISH paketini yeniden gönderir.

Bir alıcı QoS 1 seviyesi ile bir mesaj aldığı anda, hemen işleyebilir. Örneğin, alıcı bir Broker ise, Broker, mesajı tüm abone istemcilere gönderir ve ardından bir PUBACK paketi ile yanıt verir.

Yayıncı istemci mesajı tekrar gönderirse, bir kopya bayrağı (DUP Flag) oluşturur. QoS 1'de, bu DUP bayrağı yalnızca dahili amaçlar için kullanılır ve Broker veya istemciler tarafından işlenmez. Mesajın alıcısı, DUP bayrağına bakılmaksızın bir PUBACK paketi gönderir.

2.6.3 QoS 2

QoS 2 MQTT'deki en yüksek hizmet seviyesidir. QoS 2, her mesajın hedeflenen alıcılara yalnızca bir kez iletilmesini garanti eder. Bu hizmet seviyesi en güvenli fakat en yavaş hizmet seviyesidir. Telimat garantisi, gönderici ile alıcı arasında en az iki istek / yanıt akışı ile sağlanır. Gönderen ve alıcı, mesaj teslimini koordine etmek için orijinal PUBLISH mesajının paket tanımlayıcısını kullanır.

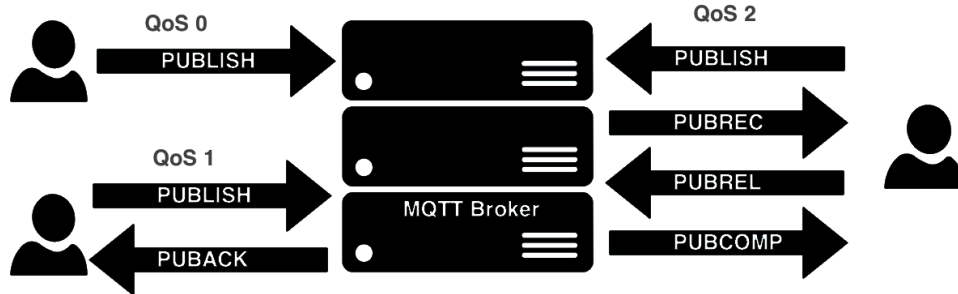
Alıcı, göndericiden bir QoS 2 PUBLISH paketi aldığı anda, yayınlama mesajını buna göre işler ve göndericiye PUBLISH paketini kabul eden PUBREC paketi ile yanıt verir. Gönderici alıcıdan bir PUBREC paketi almazsa, alındı bildirimini alana kadar PUBLISH paketini bir DUP Flag ile yeniden gönderir.

Gönderen alıcıdan bir PUBREC paketi aldığıında, gönderen ilk PUBLISH paketini yok eder. Gönderen alıcıdan gelen PUBREC paketini saklar ve PUBREL paketi ile yanıt verir.

Alıcı, PUBREL paketini aldıktan sonra, depolanan tüm durumları yok edebilir ve bir PUBCOMP paketi ile yanıtlayabilir. Alıcı, işlemeyi tamamlayana ve PUBCOMP paketini gönderene geri gönderene kadar, orijinal PUBLISH paketinin paket tanımlayıcısına bir referansını saklar. Bu adım, mesajı ikinci kez işlemeyi önlemek için önemlidir. Gönderen PUBCOMP paketini aldıktan sonra, yayınlanan mesajın paket tanımlayıcısı yeniden kullanım için uygun hale gelir.

QoS 2 mesaj akışı tamamlandığında, her iki taraf da mesajın teslim edildiğinden ve gönderenin teslimatın onayını aldığıından emin olur.

Eğer bu işlem süresince bir paket kaybolursa, gönderen mesajı kısa süre içinde yeniden iletmek durumundadır. Gönderen bir MQTT istemcisi veya MQTT Broker'ı olsa bile aynı durum geçerlidir. Alıcının her komut mesajına uygun yanıtı verme zorunluluğu vardır.



Şekil 2.2: MQTT QoS Seviyeleri

2.7 Kalıcı Oturum ve Mesaj Sıralama

Bir MQTT broker'dan mesaj alabilmek için, bir istemci broker'a bağlanır ve ilgilendiği konulara abonelikler oluşturur. Eğer istemci ile broker arasındaki bağlantı kalıcı olmayan bir oturum sırasında kesilirse, istemci broker'a yeniden bağlandığı zaman ilgilendiği konulara yeniden abone olması gerekir. Her bağlantı kesildiğinde konulara yeniden abone olmak, kaynakları kısıtlı olan istemciler için ekstra bir yüküdür. Bu ekstra yükü engellemek için istemci, broker'a bağlandığında kalıcı oturum talep edebilir.

Kalıcı oturumlarda istemci ile ilgili tüm bilgiler broker'da kaydedilir. İstemci broker'a bağlandığında, istemcinin sağladığı clientId oturumu tanımlar.

İstemci, broker'a ne tür oturuma ihtiyacı olduğunu bildirmek için cleanSession flag kullanır. CleanSession (temiz oturum) bayrağı "true" olarak belirtilmişse, istemci kalıcı oturum istemediğini belirtir. Bu durumda istemci herhangi bir nedenle bağlantıyı keserse, önceki kalıcı oturumda kuyruğa alınan tüm bilgiler ve mesajlar kaybolur.

Temiz oturum bayrağı "false" olarak belirtildiğinde, broker istemci için kalıcı bir oturum oluşturur. Tüm bilgiler ve mesajlar, istemcinin bir sonraki temiz oturum talebine kadar saklanır. Temiz oturum bayrağı "false" olarak ayarlanmış ve broker'ın istemci için kullanılabilir bir oturumu varsa, mevcut oturum kullanılır ve önceden sıraya alınmış iletiler istemciye teslim edilir.

İstemci çevrimdışı olsa bile abone olduğu konulardaki tüm mesajları almak zorundadır. Broker, iletileri sıraya koyar ve istemci çevrimişi olur olmaz tüm iletileri teslim eder.

2.8 Retained Messages

Retained message (saklanan mesaj), Retained flag "true" olarak işaretlenmiş normal bir MQTT mesajıdır. Broker, saklanan mesajı ve bu mesaja karşılık gelen QoS seviyesini depolar.

Saklanan mesajın konusuna abone olan her istemci, abone olduktan hemen sonra saklanan mesajı alır. Broker, konu başına yalnızca bir mesaj saklayabilir.

Abone olan istemci abone olduğu konu modelinde joker karakterler içeriyorsa, saklanan mesajın konusu tam olarak eşleşmese bile saklanan bir mesaj alır.

Saklanan mesaj, yeni bağlanan istemcilerin, broker bir istemcinin bir sonraki mesajı göndermesini beklemeden hemen mesaj almasını istendiğinde kullanılmaktadır.

2.9 Last Will and Testament

MQTT'de, diğer istemcilere beklenmedik bir şekilde bağlantısı kesilmiş bir istemciyi bildirmek için Last Will and Testament (LWT) özelliği kullanılır. İstemciler, broker'a bağlandıktan sonra LWT mesajını belirleyebilirler. Son istek mesajı (LWT),

konu, saklanan mesaj bayrağı, QoS seviyesi ve mesaj içeren normal bir MQTT mesajıdır.

Broker, istemcinin bağlantısının beklenmedik bir şekilde kesildiğini algılayana kadar iletiyi saklar. Beklenmedik bir bağlantı kesilmesi olduğu zaman, broker son istek mesajını abone olan istemcilere gönderir.

İstemci, doğru bir “DISCONNECT” mesajıyla sorunsuz bir şekilde bağlantıyı keserse, broker depolanan LWT mesajını atar.

2.10 Keep Alive ve Client Take Over

2.10.1 Keep Alive

MQTT’de, yarı açık bağlantılar için geçici bir çözüm sağlayan veya bağlantının hala açık olup olmadığını denetleyen bir Keep Alive (canlı tutma) işlevi bulunmaktadır. Canlı tutma, broker ile istemci arasındaki bağlantının hala açık olmasını denetler ve broker ile istemcinin bağlı olduklarının farkında olmasını sağlar.

İstemci, broker ile bir bağlantı kurduğunda, broker’a saniye cinsinden bir zaman aralığı bildirir. Bu aralık, broker ve istemcinin birbiriyle iletişim kuramayabileceği maksimum süreyi tanımlar.

Mesajlar karşılıklı olarak alınıp verildiği ve canlı tutma aralığı aşılmadığı sürece, bağlantının hala açık olup olmadığını belirlemek için fazladan bir mesaj göndermeye gerek yoktur. İstemci canlı tutma süresi boyunca bir ileti göndermezse, broker’ın kullanılabilir olduğunu doğrulamak ve broker’ın da hala kullanılabilir olduğundan emin olmak için broker’a bir PINGREQ paketi gönderir. Broker, canlı tutma aralığının bir buçuk katı içinde bir ileti veya PINGREQ paketi göndermeyen bir istemcinin bağlantısını keser. Benzer şekilde, istemcinin broker’dan makul bir süre içinde yanıt almaması durumunda bağlantıyı kapatması beklenir.

2.10.2 Client Take Over

Genellikle bağlantısı kesilen bir istemci, broker’a yeniden bağlanmaya çalışır. Bazen, broker’ın istemci için hala yarı açık bir bağlantısı vardır. Broker yarı açık bir bağlantı tespit ederse, bir Client Take Over (İstemci Devralma) gerçekleştirir. Aracı, aynı istemciyle olan önceki bağlantıyı kapatır ve istemciyle yeni bir bağlantı kurar. Bu

şekilde, yarı açık bağlantının, bağlantısı kesilen istemcinin yeniden bağlantı kurmasının engellenmemesi sağlanır.

3. BÜYÜK VERİ “BIG DATA”

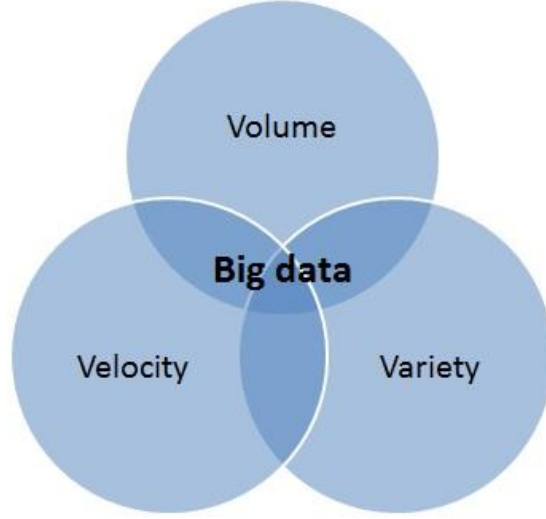
3.1 Büyük Veri Nedir?

Boyutu veya türü, geleneksel veritabanlarının düşük gecikmeyle verileri yakalama, yönetme ve işleme yeteneğinin ötesinde olan veri kümeleri olarak tanımlanabilmektedir. Büyük Veri, bilgi için madencilik yapma, makine öğrenme projelerinde ve diğer gelişmiş analitik uygulamalarında kullanılma potansiyeline sahip büyük hacimli yapılandırılmış, yarı yapılandırılmış ve yapılandırılmamış verileri tanımlayan bir terimdir.

Büyük Verinin özellikleri arasında yüksek hacim, yüksek hız ve yüksek çeşitlilik yer almaktadır. Veri kaynakları, yapay zekâ, mobil cihazlar, sosyal medya ve Nesnelerin İnterneti (IoT) tarafından yönlendirildiği için geleneksel verilere göre daha karmaşık bir hale gelmektedir.

3.2 Büyük Veri Bileşenleri (3V)

Büyük Veride üç tanımlayıcı özellik vardır ve “3V” olarak adlandırılmaktadır. Hacim (Volume), hız (Velocity) ve çeşitlilik (Variety), bunlar büyük veriyi nasıl ölçebileceğimizi ve büyük verininin geleneksel verilerden ne kadar farklı olduğunu göstermektedir.



Şekil 3.1: Büyük Veri Bileşenleri

3.2.1 Hacim (Volume)

Çeşitli operasyonel işlemler, sensör veya makineden makineye verilerden gelen bilgiler gibi çeşitli kaynaklardan veriler toplanır. Geçmişte, bu verileri depolamak büyük bir sorundu fakat, gelişen teknolojiler sayesinde bu sorun hafifledi. Büyük Veri, muazzam bir boyutla ilgilidir. Verilerin boyutu, verilerin değerinin belirlenmesinde önemli bir rol oynar. Belirli bir verinin gerçekten Büyük Veri olarak kabul edilip edilemeyeceği, verinin hacmine bağlıdır.

3.2.2 Hız (Velocity)

Veri alışverişi çok yüksek hızda olmaktadır ve veriler tam zamanında işlenmelidir. Sensörler ve akıllı ölçümler, neredeyse gerçek zamanlı veri akışlarıyla başa çıkma ihtiyacını artırmaktadır.

Buradaki “Hız”, veri üretme hızını ifade etmektedir. Verilerin talepleri karşılamak için ne kadar hızlı üretildiği ve işlendiği, verilerdeki gerçek potansiyeli ortaya çıkarmaktadır. Veri akışı çok büyük ve sürekli.

3.2.3 Çeşitlilik (Variety)

Veriler, yapılandırılmış veri kümelerinden, geleneksel veritabanlarındaki sayısal verilerden yapılandırılmamış metin belgelerine, e-postaya, videoya, sese kadar her tür biçimde gelebilir. Çeşitlilik hem yapılandırılmış hem de yapılandırılmamış verilerin doğasını ifade eder. Geçmişte, tablolar ve veritabanları, uygulamaların çoğu

tarafından dikkate alınan tek veri kaynaklarıydı. Artık analiz uygulamalarında e-posta, fotoğraf, video, izleme cihazları, PDF, ses vb. veriler de dikkate alınmaktadır.

3.3 Büyük Verinin Önemi

Büyük veri analizi, kuruluşların verilerinden yararlanmasına ve yeni fırsatları belirlemek için kullanılmasına yardımcı olabilmektedir. Bu da daha akıllı iş hamlelerine, daha verimli operasyonlara ve daha yüksek karlara sebep olabilmektedir.

Büyük veri teknolojileri, büyük miktarda verinin depolanması söz konusu olduğunda önemli maliyet avantajları sağlamaktadır, ayrıca iş yapmanın daha verimli yollarını belirleyebilirler.

Analitiğin hızı, yeni veri kaynaklarını analiz etme becerisiyle birleştiğinde, işletmeler bilgileri anında analiz edebilmekte ve bu analizlere göre hızlı kararlar alabilmektedirler.

3.4 Büyük Veri Nerelerde Kullanılır?

Büyük veri analitiğinin amacı, veri bilimcilerinin, tahmine dayalı modelleyicilerin ve diğer analitik uzmanlarının büyük hacimli işlem verilerini ve daha geleneksel yöntemlerle kullanılmayan diğer veri biçimlerini analiz etmesini sağlayarak şirketlerin daha bilinçli iş kararları almalarına yardımcı olmaktır. İş Zekâsı (BI) programları. Bu, web sunucusu günlüklerini ve web sitesi akışı verilerini, sosyal medya içeriğini ve sosyal ağ etkinlik raporlarını, müşteri e-postalarından ve anket yanıtlarından gelen metinleri, cep telefonu araması ayrıntı kayıtlarını ve sensörler tarafından yakalanan ve Nesnelere İnternetine bağlı makine verilerini içerebilir.

4. KESTİRİMCİ BAKIM ÇALIŞMALARI

4.1 Kestirimci Bakım Nedir?

Kestirimci bakım, imalattan toplu taşımaya, madencilğe ve lojistiğe kadar çeşitli sektörlerdeki işletmelerin arasında güncel bir konudur. Son teknoloji bir bakım stratejisi olan kestirimci bakım, bir işletmenin makinelerinin durumunu izlemek ve ölçmek için IoT'nin, verilerin ve gelişmiş analitik algoritmaların gücünü kullanır. Makinelerin ne zaman bakıma ihtiyaç duyacağını kesin olarak tahmin edebilir ve bakım faaliyetlerini planlayarak, gerçekleştirmek için en uygun zaman aralığını önerir.

Sayıları gittikçe artan işletme, kestirimci bakıma dayalı planlama ve operasyonları keşfetmektedir. Bu onların makine arızalarını önceden görmelerine ve yalnızca gerekli olduğunda bakım yapmalarına olanak tanıyor. Kestirimci bakım sayesinde makine duruş sürelerini ve bakım maliyetlerini en aza indirmek mümkün olmaktadır.

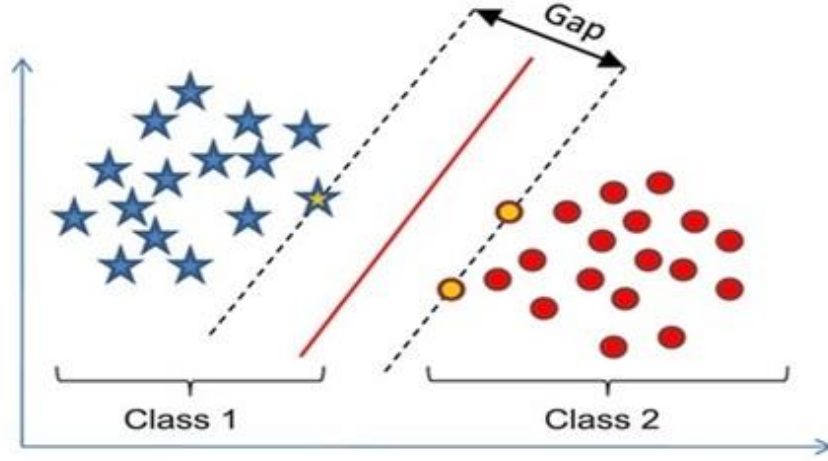
4.2 Makine Öğrenmesi ile Kestirimci Bakım

Kestirimci Bakım görevlerinin üstesinden gelmek için hem geleneksel makine öğreniminden hem de derin öğrenmeden çok sayıda yöntem kullanılmaktadır.

4.2.1 Support Vector Machines (SVM)

Farklı kategorilere ait örneklerin geniş bir boşlukla ayrıldığı bir şekilde veri nesnelerinin uzayda noktalar olarak temsillerini oluşturan, yaygın olarak kullanılan denetimli bir yöntemdir. Yeni veri noktaları tanıtıldığında, algoritma bunları içine düştikleri boşluğun tarafına göre pozitif veya negatif olmak üzere iki sınıftan birine atar.

SVM'nin uygulanması kolaydır. Genellemeye başarılıdır ve yüksek sınıflandırma doğruluğu sağlar. Bu da onu hata tespiti ve kalan faydalı ömür (RUL) tahmini için başarılı kılar.



Şekil 4.1: Destek Vektör Makineleri (SVM)

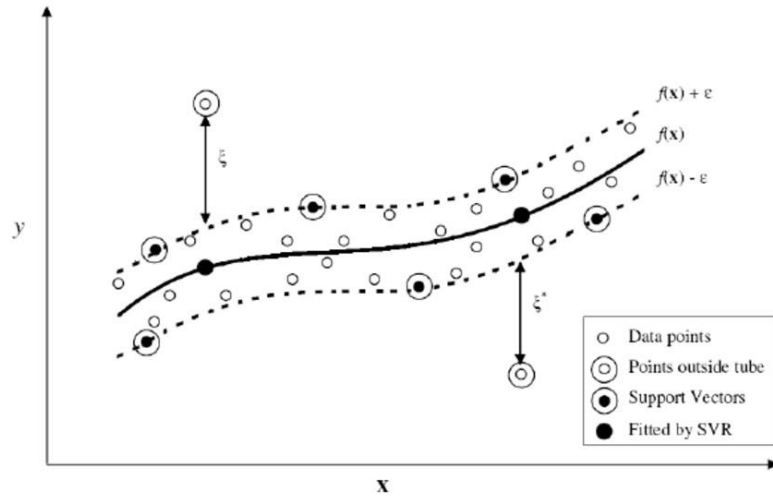
4.2.2 Support Vector Regression (SVR)

Destek Vektör Regresyonu (SVR), çekirdek kullanımı ve marjin VC kontrolü ile SVM'ler gibi karakterize edilen, endüstriyel amaçlar için de kullanışlıdır.

Firmaların modelde ne kadar hatanın kabul edilebilir olduğunu anlamalarına ve daha yüksek boyutlarda hiper düzlemi veya verilerine uyan bir çizgiyi belirlemelerine yardımcı olabilir.

SVM yöntemi kadar popüler olmasada, SVR de hata prognozu için yaygın olarak kullanılmaktadır. Sensörlerden gelen verilere dayanan ve bozulma durumları, sağlık göstergeleri veya arıza eşikleri için ön tahminler gerektirmeyen, SVR'ye dayalı bir kalan faydalı ömür tahmin yöntemi önermektedir.

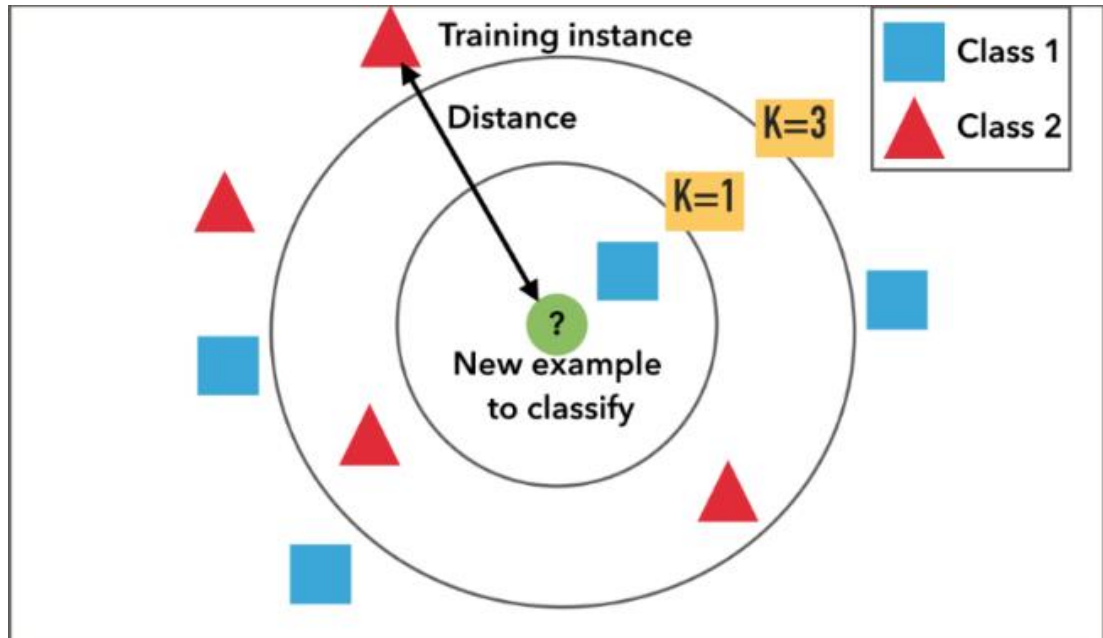
SVR, sensörlerin değerleri ve sağlık göstergeleri arasındaki ilişkileri açıkça modellemek için kullanılır ve bozulma sürecinin herhangi bir noktasında RUL'un tahmin edilmesine imkân sağlar.



Şekil 4.2: Destek Vektör Regresyonu (SVR)

4.2.3 K-Nearest Neighbors (KNN)

Makine öğreniminde en yaygın kullanılan tekniklerden biri olan K-en yakın komşular, benzer K veri nesnelerinin bulunduğu bir kategoriye görünmeyen veri örneklerini atar. Benzerlik, belirli bir veri kümesi için neyin daha uygun olduğuna bağlı olarak Öklit, Manhattan, Chebyshev veya diğer mesafe ölçümleriyle uygulanabilmektedir.



Şekil 4.3: KNN Sınıflandırma Örneği

5. DENEYİN YAPILIŞI

5.1 Giriş

Bu bölümde, tezde tasarlanan deneyler ve deney sonuçları verilmektedir. Deney ortamını sağlayamadığım için rulmanlar üzerinden alınan titreşim verilerinin olduğu veri kümeleri [41] ile çalışma yapılmıştır. İlk olarak veri kümeleri zaman ve frekans alanlarında incelenerek muhtemel arızalar erkenden teşhis edilerek kestirimci bakım çalışmaları yürütülecektir. Bu çalışma frekans alanında spektral analiz çalışmalarına odaklanacaktır. Zaman ve frekans alanlarında yapılan kesitirmci bakım çalışmalarının faydaları, avantajları ve dezavantajları üzerinde tartışılacak ve karşılaştırma yapılacaktır.

5.2 İş İçeriği

5.2.1 Geliştirme Ortamı

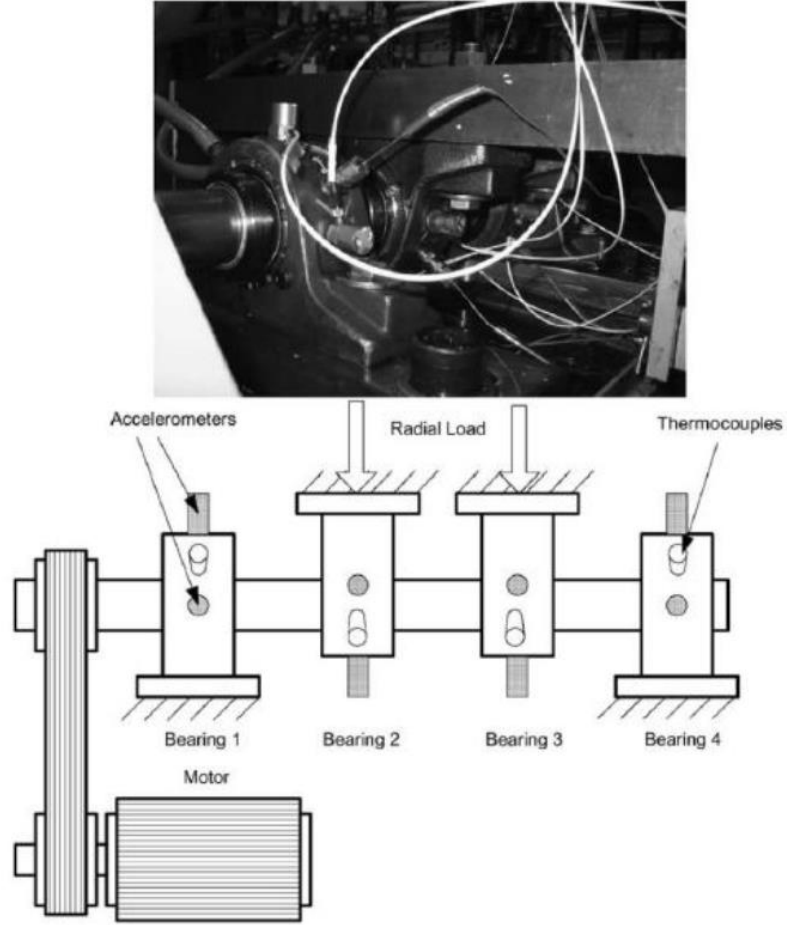
Çalışmaların uygulanması MATLAB kullanılarak gerçekleştirilmiştir. MATLAB mühendislerin ve bilim adamlarının sistemleri ve ürünleri analiz etmesi ve tasarlaması için özel olarak tasarlanmış bir dördüncü nesil programlama platformudur.

Dördüncü nesil programlama platformları; kolay, daha az kod yazarak yönergeler ve hazır şablonlar kullanılarak belirli ihtiyaçlarda uzmanlaşmış pratik çözümler geliştirmeye yönelik platformlara verilen isimdir.

5.2.2 Kullanılan Veri Kümeleri

Veriler, 2000 devir/dakika sabit bir hızda dönen yaklaşık 3 ton (6000 lbs) yüklü bir shaft üzerindeki dört rulmanı içeren bir “run to failure” deneyden alınmıştır ve üç farklı veri kümesinden oluşmaktadır: Birinci deney setinde, x ve y eksenleri için 2 yüksek hassasiyetli ivmeölçer kullanılmıştır. Her bir rulmanın üzerine iki adet ivme ölçer monte edilirken, ikinci ve üçüncü veri kümelerinde rulman başına bir ivme ölçer kullanılmıştır. Her veri kümesi, belirli aralıklarla kaydedilen 1 er saniyelik titreşim sinyalinin anlık görüntüsünü içeren ayrı dosyalardan oluşmaktadır. Her dosya, 20 kHz'lik bir örnekleme hızı ile 20.480 noktadan oluşmaktadır. Dosya adı,

verilerin ne zaman toplandığını göstermektedir. Veri dosyasındaki her kayıt bir veri noktasıdır. Bu çalışmada, iki sinyal içerdiği için daha zengin bilgi sağlayacağından dolayı birinci veri kümesine odaklanılmıştır. Deney kurulumu Şekil 5.1'deki gibidir [41].



Şekil 5.1: Deney Kurulumu

5.3 Veri Kümeleri İle Yapılan Çalışmalar

Kullanılan veri kümeleri [41], 5 er dakika aralıklarla kaydedilmiş toplamda 2156 adet dosyadan, her bir dosya da 20480 ölçümden oluşmaktadır. Örnekleme hızının 20 kHz olduğunu düşünürsek, bir ölçümün süresi 1.024 saniye olduğunu görmekteyiz.

Bu veri kümesi üzerinde yapılan çalışmada, yedi farklı sağlık durumu gözlenmiştir.

Sağlık durumları sırasıyla şu şekildedir:

- Erken (düzeneğin ilk çalıştırması)
- Normal
- Şüpheli (durum kötüye gidiyor)
- Muhtemel arıza başlangıcı
- İç yatak arızası
- Yuvarlanma elemanı (bilye) arızası
- 2 aşama arıza

Veri kümeleri içerisindeki dosyalar “yyyy.MM.dd.hr.mm.ss” şeklinde “yıl.ay.gün.saate.dakika.saniye” formatında kayıt edilmiştir.

Çalışılan veri kümeleri ile ilgili aşağıdaki bilgiler edinilmiştir:

Tablo 5.1: 1.Rulman Evreleri

Rulman 1	
Erken Evre	2003.10.22.12.06.24 - 2013.1023.09.14.13
Normal Evre	2003.11.08.12.21.44 - 2003.11.19.21.06.07
Şüpheli Evre	2003.11.19.21.16.07 - 2003.11.24.20.47.32
Muhtemel Arıza Evresi	2003.11.24.20.57.32 - 2003.11.25.23.39.56

Tablo 5.2: 2.Rulman Evreleri

Rulman 2	
Erken Evre	2003.10.22.12.06.24 - 2003.11.01.21.41.44
Normal Evre	2003.11.01.21.51.44 - 2003.11.24.01.01.24
Şüpheli Evre	2003.11.24.01.11.24 - 2003.11.25.10.47.32
Muhtemel Arıza Evresi	2003.11.25.10.57.32 - 2003.11.25.23.39.56

Tablo 5.3: 3.Rulman Evreleri

Rulman 3	
Erken Evre	2003.10.22.12.06.24 - 2003.11.01.21.41.44
Normal Evre	2003.11.01.21.51.44 - 2003.11.22.09.16.56
Şüpheli Evre	2003.11.22.09.26.56 - 2003.11.25.10.47.32
İç Yatak Arızası	2003.11.25.10.57.32 - 2003.11.25.23.39.56

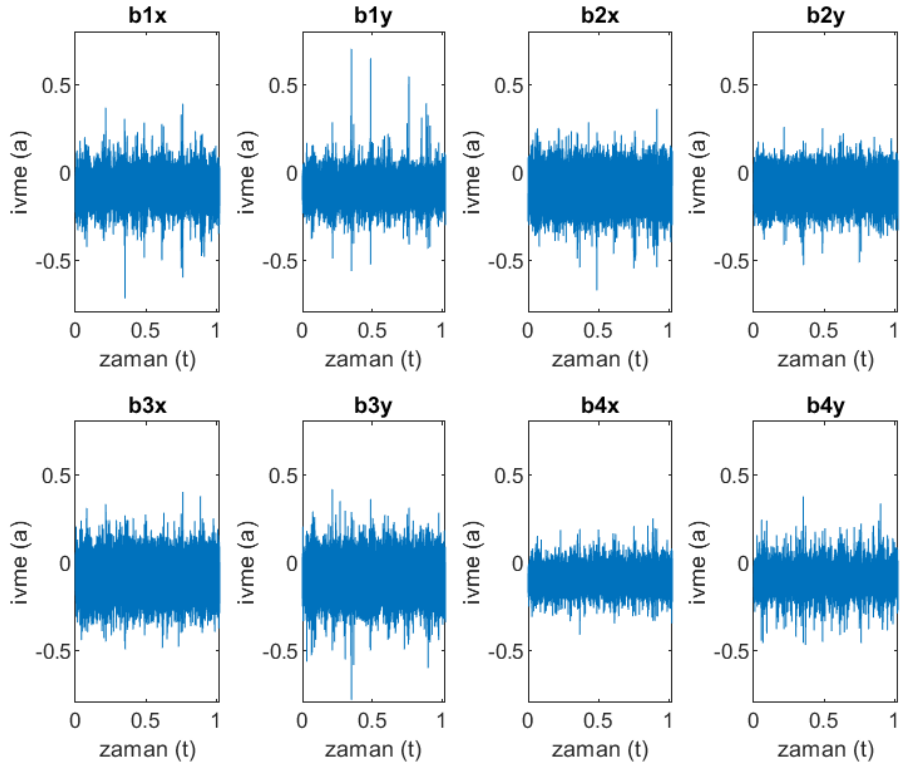
Tablo 5.4: 4.Rulman Evreleri

Rulman 4	
Erken Evre	2003.10.22.12.06.24 - 2003.10.29.21.39.46
Normal Evre	2003.10.29.21.49.46 - 2003.11.15.05.08.46
Şüpheli Evre	2003.11.15.05.18.46 - 2003.11.18.19.12.30
Bilye Arızası	2003.11.19.09.06.09 - 2003.11.22.17.36.56
2. Aşama Arıza	2003.11.22.17.46.56 - 2003.11.25.23.39.56

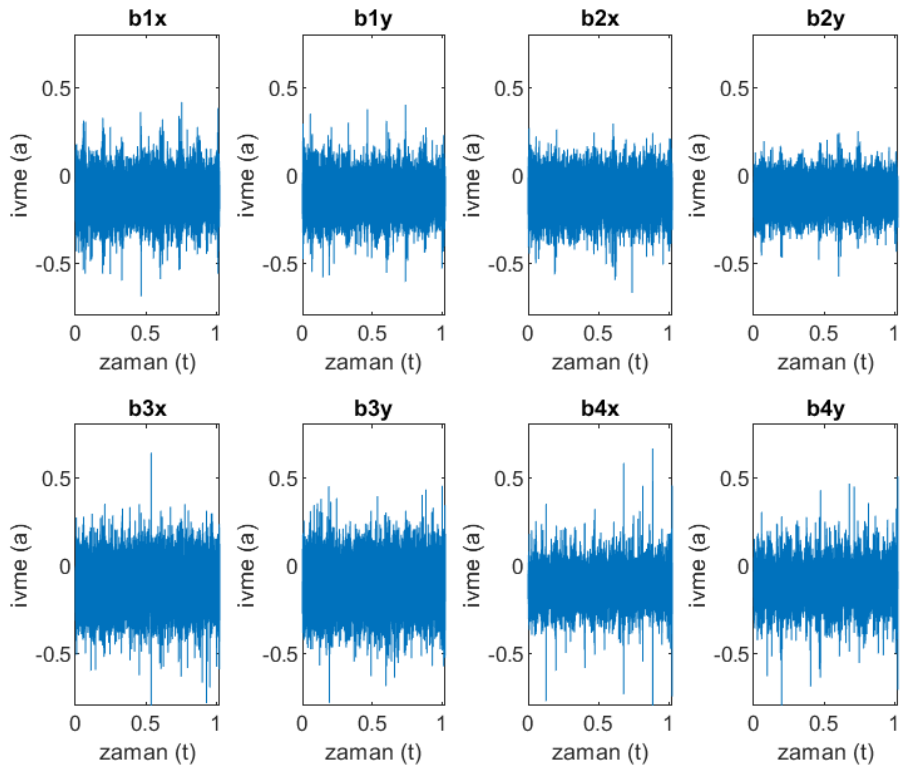
Tablo 5.5: Rulman Özellikleri

Pitch diameter	2.815 inch	71.5mm
Rolling element diameter	0.331 inch	8.4mm
Number of rolling elements per row	16	16
Contact angle	15.17°	15.17°
Static load	6000 lbs	26690 N

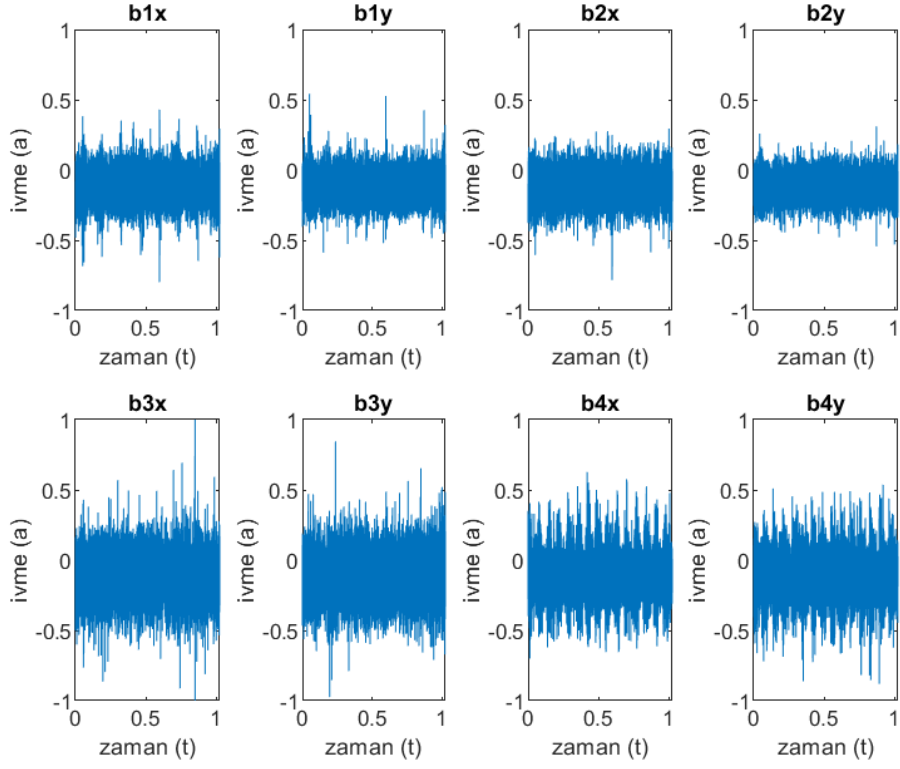
Rulmanların üzerindeki sensörlerden elde edilen, belirtilerin görüldüğü zamanlardaki ivme – zaman grafikleri aşağıda sıralanmıştır.



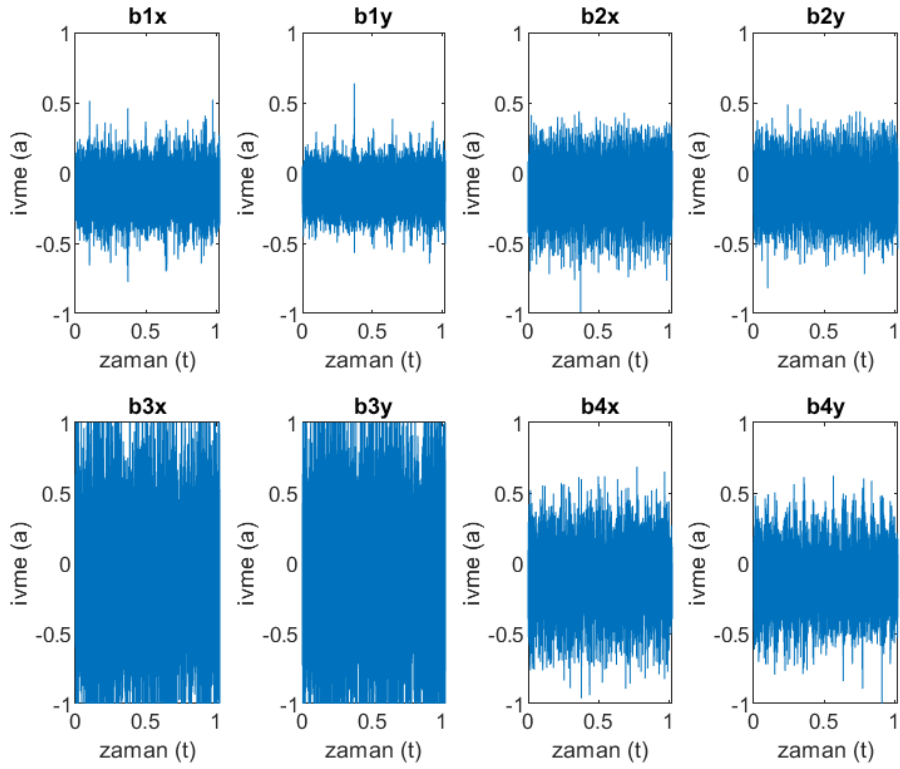
Şekil 5.2: Rulman1 Erken Evre



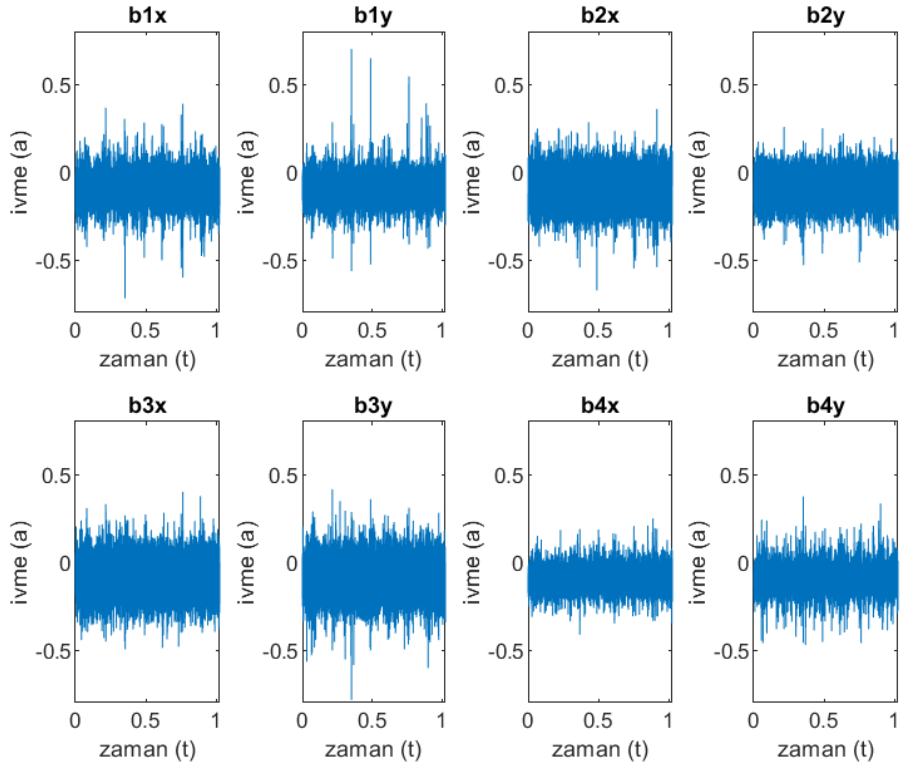
Şekil 5.3: Rulman1 Normal Evre



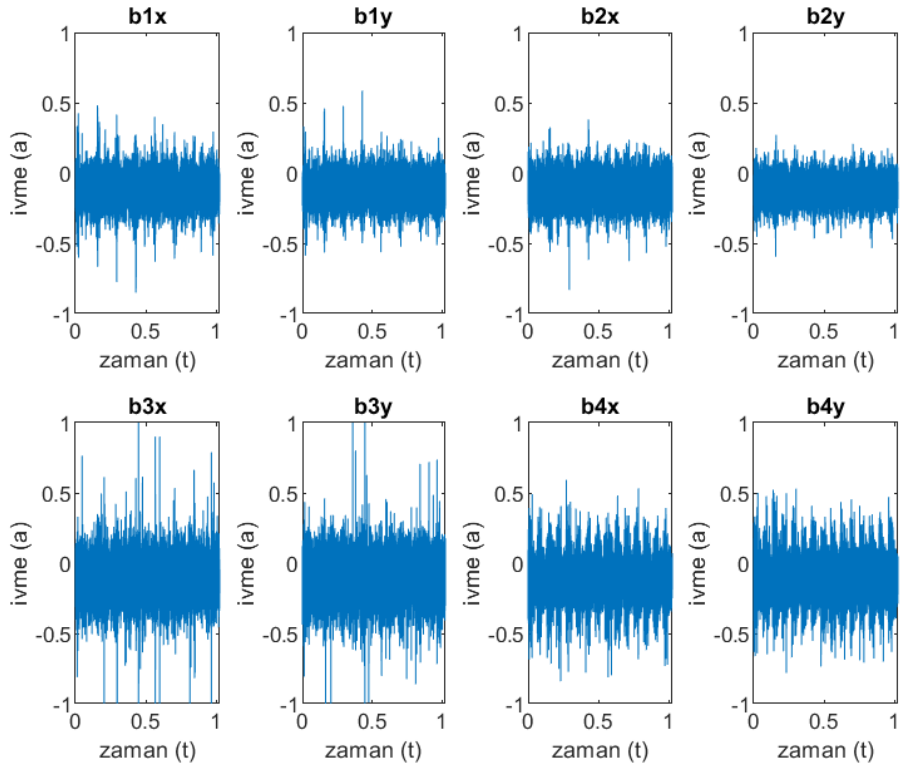
Şekil 5.4: Rulman1 Şüpheli Evre



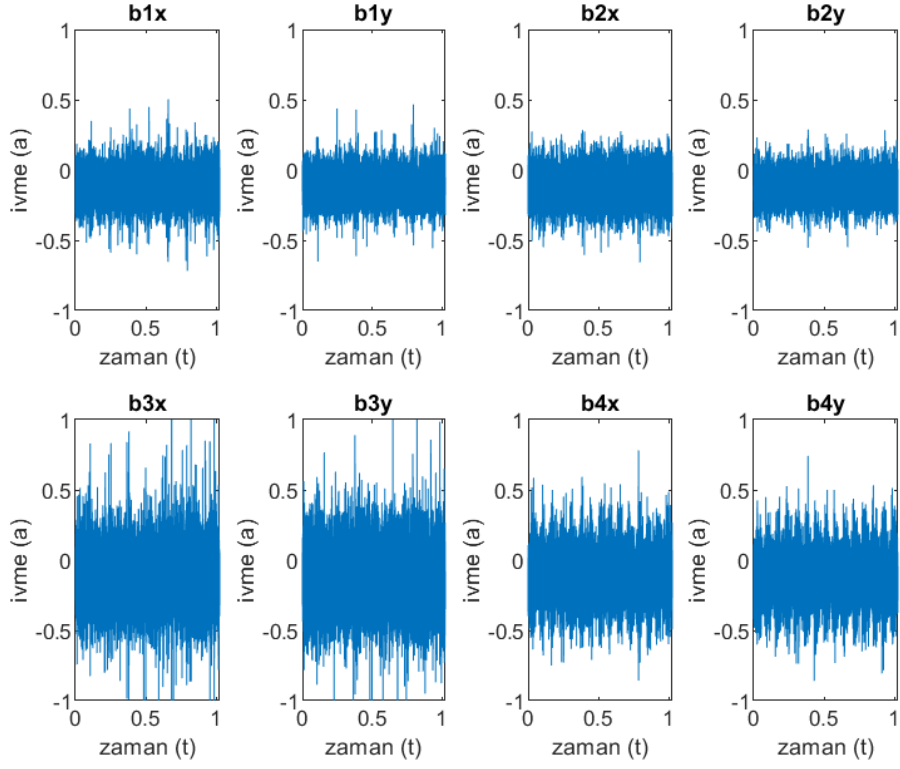
Şekil 5.5: Rulman1 Muhtemel Arıza Başlangıcı



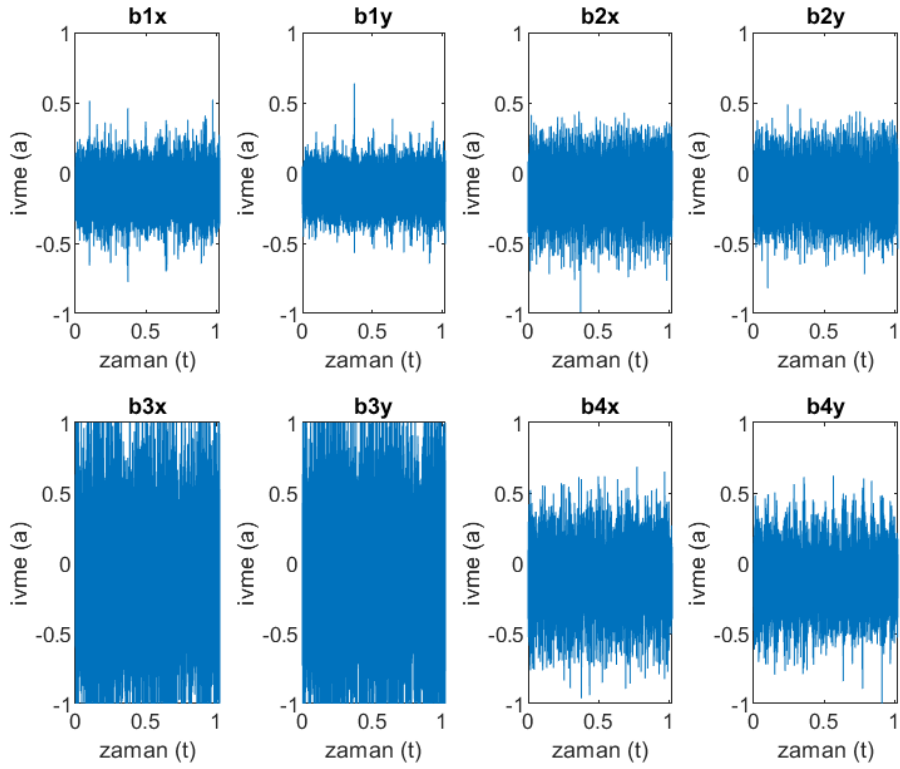
Şekil 5.6: Rulman2 Erken Evre



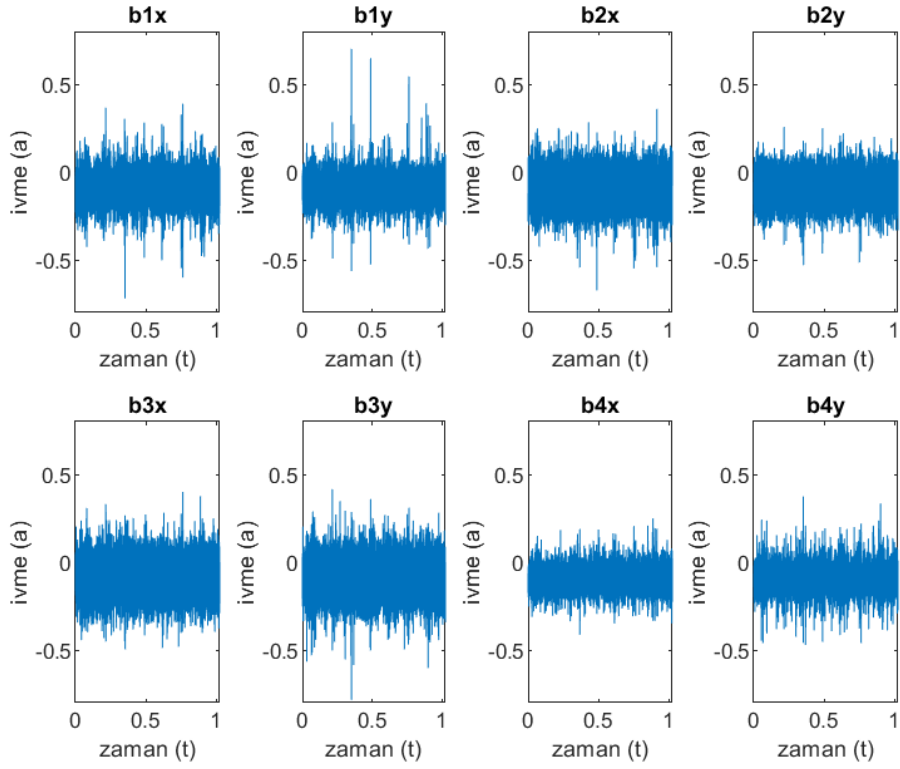
Şekil 5.7: Rulman2 Normal Evre



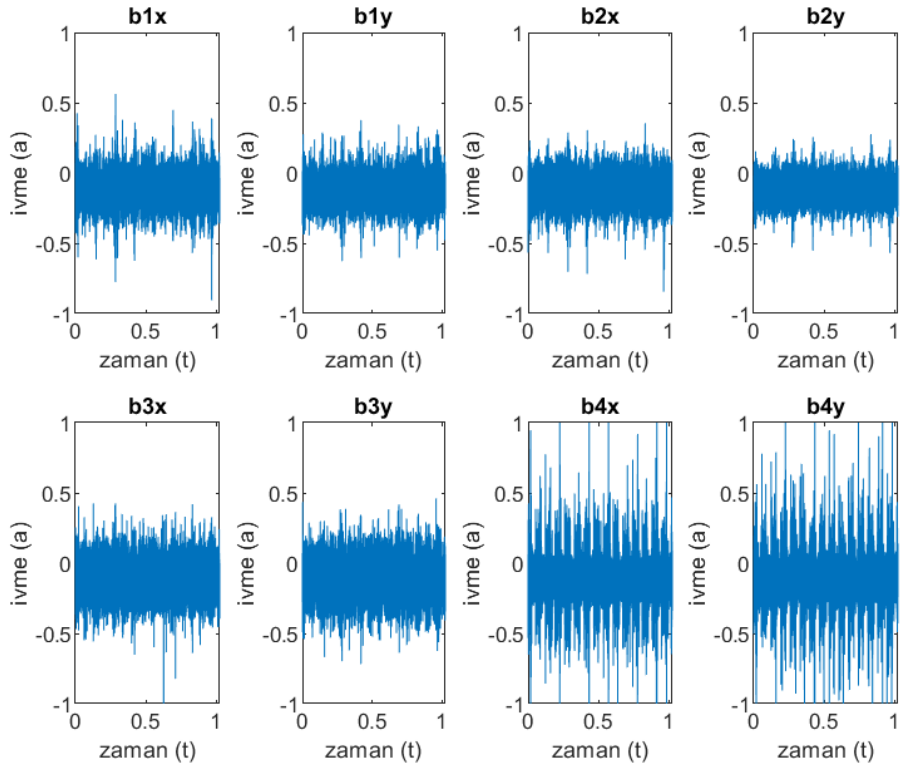
Şekil 5.8: Rulman2 Şüpheli Evre



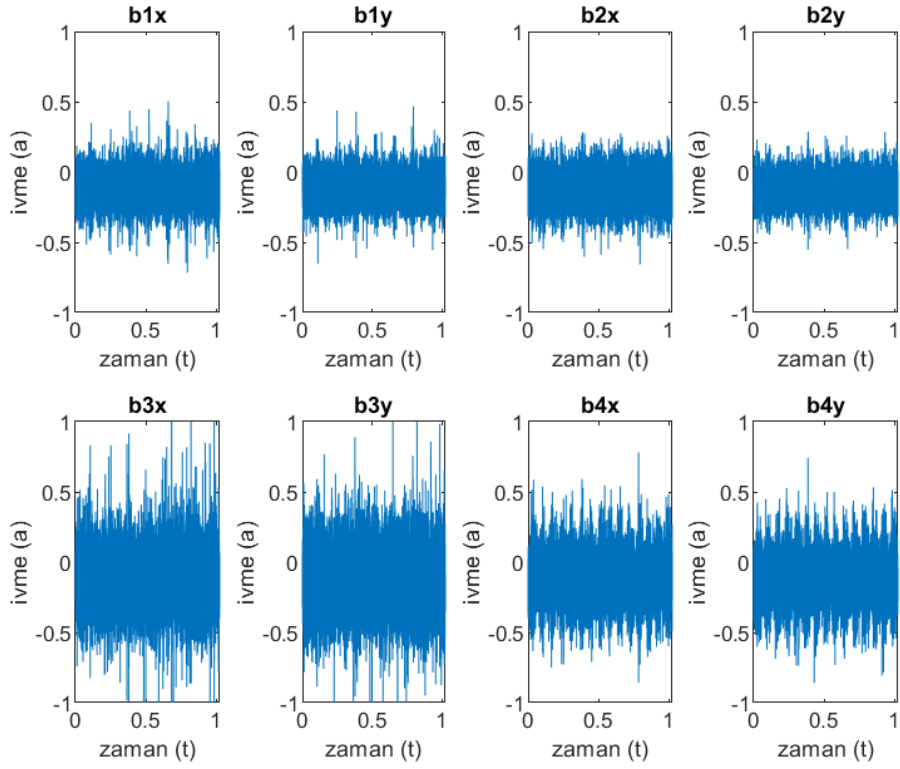
Şekil 5.9: Rulman2 Muhtemel Arıza Başlangıcı



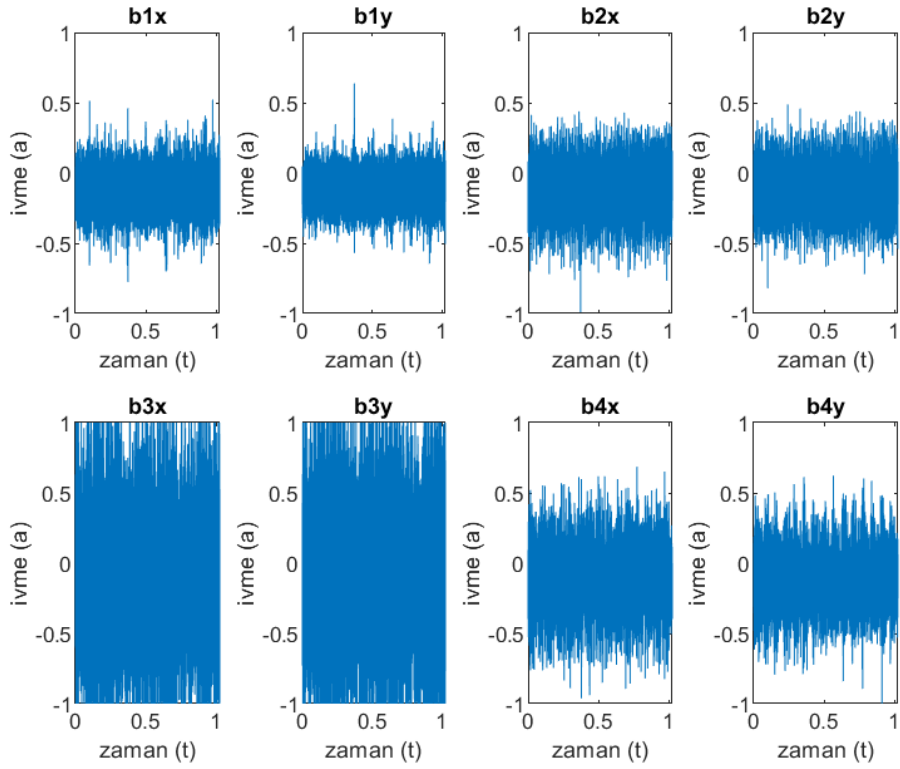
Şekil 5.10: Rulman3 Erken Evre



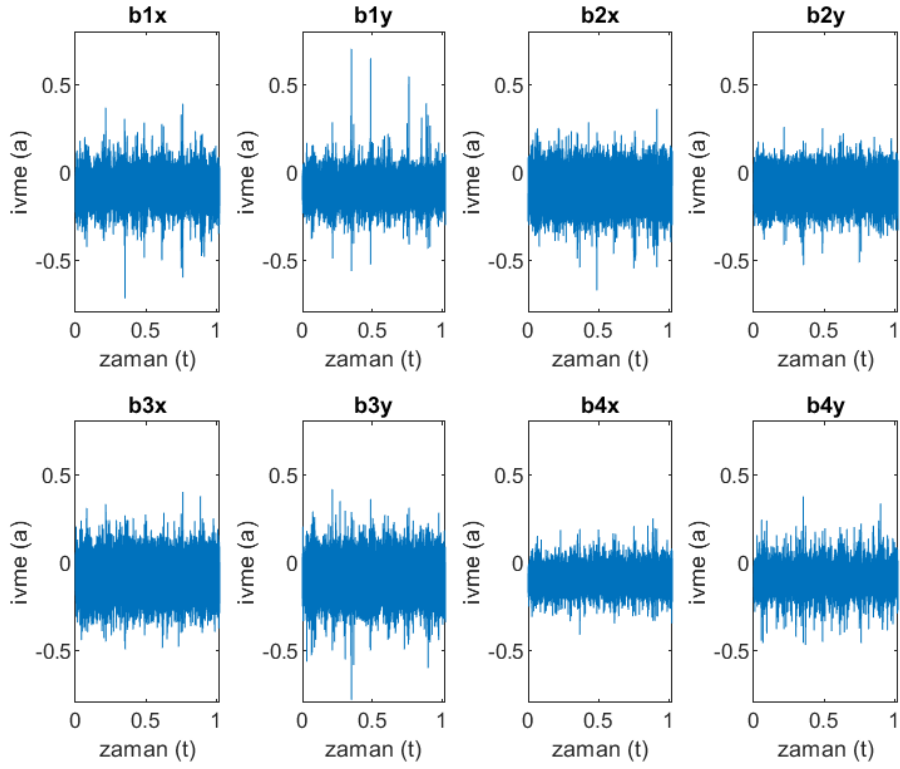
Şekil 5.11: Rulman3 Normal Evre



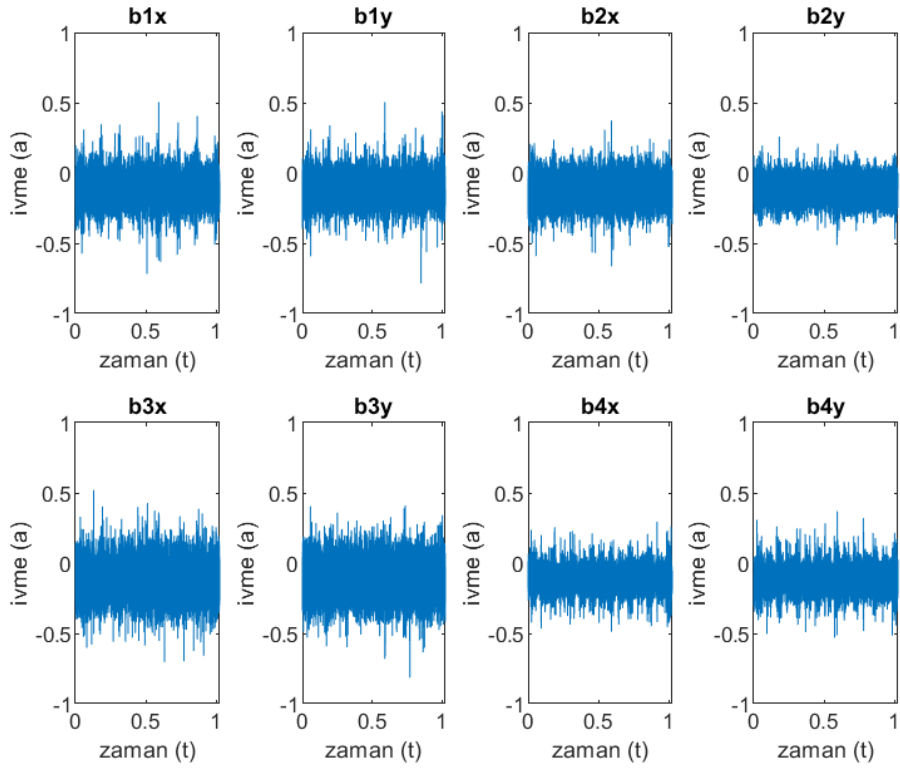
Şekil 5.12: Rulman3 Şüpheli Evre



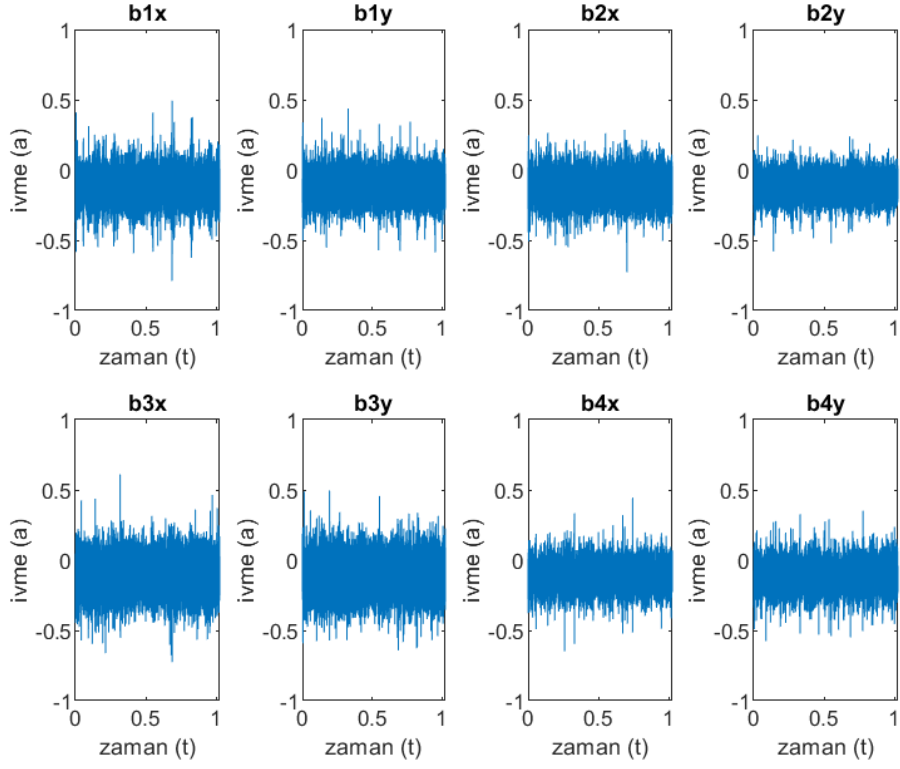
Şekil 5.13: Rulman3 Muhtemel Arıza Başlangıcı



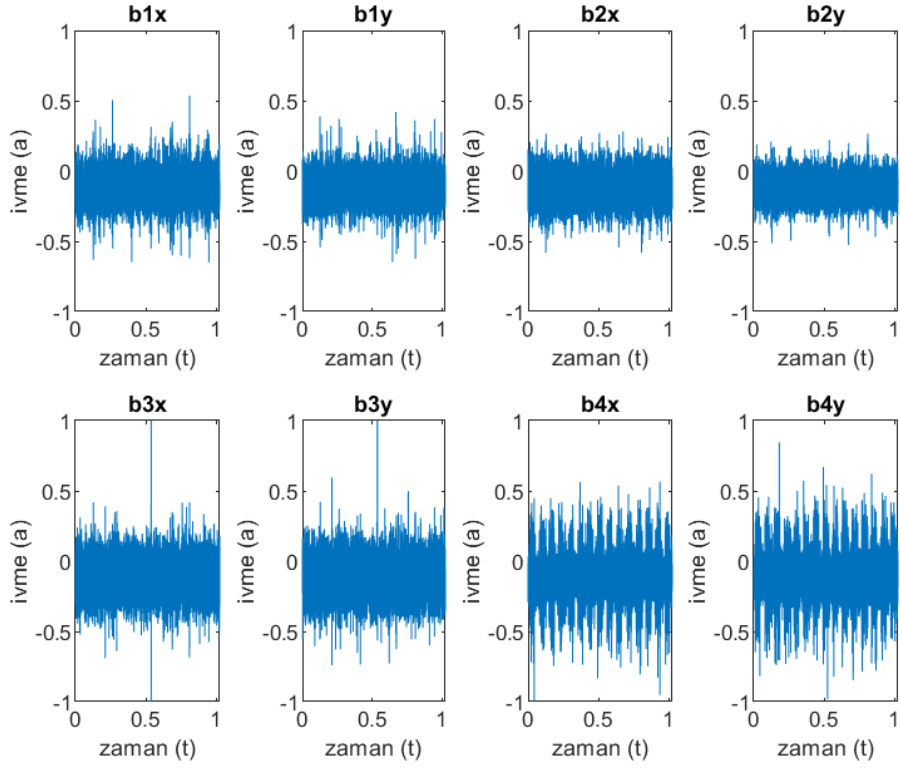
Şekil 5.14: Rulman4 Erken Evre



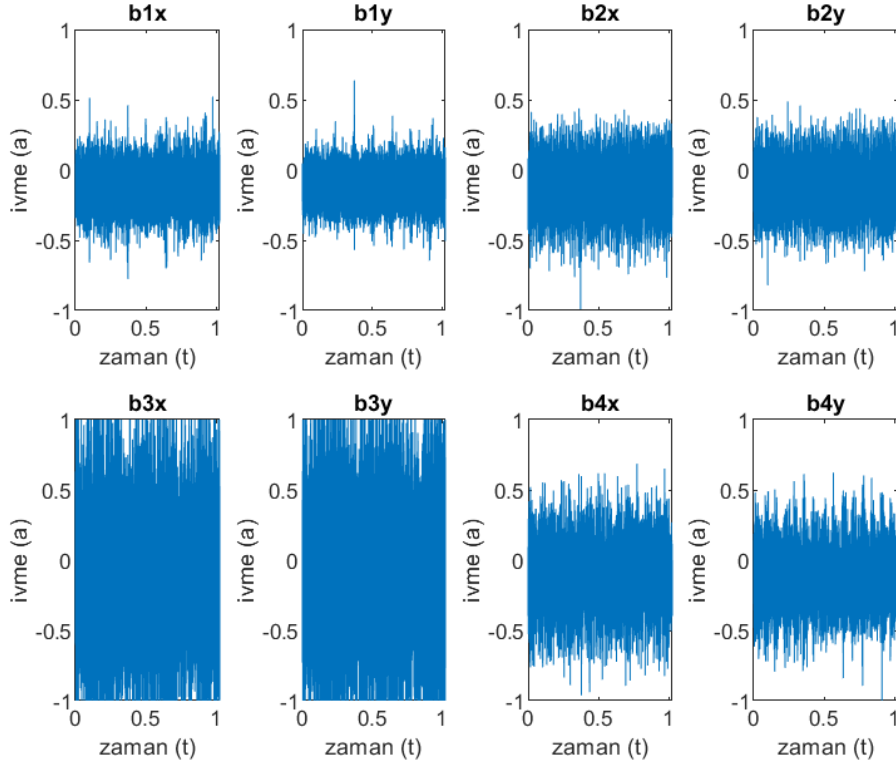
Şekil 5.15: Rulman4 Normal Evre



Şekil 5.16: Rulman4 Şüpheli Evre



Şekil 5.17: Rulman4 Bilye Arızası



Şekil 5.18: Rulman4 2. Aşama Arıza

5.4 Spektral Analiz Yöntemleri

Bir spektrumdan bir makine hakkında çeşitli bilgileri okuyabilmek için, spektrumda hangi frekansların oluşabileceğinin bilinmesi faydalı olacaktır. Spektrum genellikle bir dizi ayırık çizgi ve artan gürültünün olduğu alanlar içerir. Spektrumu, çeşitli semptomların meydana geldiği alanlara bölmek uygundur.

Genellikle spektrumun değerlendirme süreci, shaftın dönme hızıyla ilgili frekansın tanımlanmasıyla başlar. “1X” gösteriminin kullanıldığı dönme frekansıdır.

Bu dönme frekanslarının (2X, 3X,...) integral katları tanımlanır, bunlar da harmonikler olarak tanımlanmaktadır.

Spektrum genellikle üç ana alana ayrılır:

- Eşzamansız olarak adlandırılan dönme frekansının altındaki alan.
- Dönme frekansından on katına kadar alan (10X).
- 10X'in üzerindeki alan, yüksek frekanslı olayların meydana geldiği alan.

5.5 Rulman Arıza Frekansları

Rulmanlar üzerindeki belirli kusurları tespit edebilmek için çeşitli formüller türetilmiştir. Bu formüller; rulmanın geometrisine, rulman içerisindeki bilyelerin sayısına ve rulmanın dönme frekansına dayanmaktadır.

Hatanın nerede meydana geldiğine bağlı olarak, rulman üzerinde dört tür hata ayırt edilebilmektedir. Rulman arızası frekansları, rulman parametreleri ve shaftın dönme hızına bağlı olarak hesaplanabilmektedir.

5.5.1 Ball Pass Frequency Inner (BPFI)

Ball Pass Frequency Inner (BPFI) veya iç yatak arızalanma frekansı, shaft her tam dönüş yaptığında iç yatağın belirli bir noktasından geçen bilye veya silindirlerin sayısına fiziksel olarak karşılık gelmektedir.

5.5.2 Ball Pass Frequency Outer (BPFO)

Ball Pass Frequency Outer (BPFO) veya dış yatak arızalanma frekansı, shaft her tam dönüş yaptığında dış yatağın belirli bir noktasından geçen bilye veya silindirlerin sayısına fiziksel olarak karşılık gelmektedir.

5.5.3 Ball Spin Frequency (BSF)

Ball Spin Frequency (BSF) veya bilye arızalanma frekansı, shaft her tam dönüş yaptığında bir rulman bilyesinin veya silindirinin yaptığı dönüş sayısına fiziksel olarak karşılık gelmektedir.

5.5.4 Fundamental Train Frequency (FTF)

Fundamental Train Frequency (FTF) veya kafes arıza frekansı, shaft her tam dönüş yaptığında bilye kafesini oluşturan dönüş sayısına fiziksel olarak karşılık gelmektedir.

$$BPFI = \frac{N}{2} \left(1 + \frac{B_d}{P_d} \cos(\phi) \right) n \quad (5.1)$$

$$BPFO = \frac{N}{2} \left(1 - \frac{B_d}{P_d} \cos(\phi) \right) n = N \times FTF \quad (5.2)$$

$$BSF = \frac{P_d}{2B_d} \left(1 - \left(\frac{B_d}{P_d} \cos(\phi) \right)^2 \right) n \quad (5.3)$$

$$FTF = \frac{1}{2} \left(1 - \frac{B_d}{P_d} \cos(\phi) \right) n \quad (5.4)$$

Tablo 5.6: Karakteristik Frekanslar

Şaft Frekansı	33.3 Hz
Ball Pass Frequency Outer (BPFO)	236 Hz
Ball Pass Frequency Inner (BPFI)	297 Hz
Ball Spin Frequency (BSF)	140 Hz
Fundamental Train Frequency (FTF)	15 Hz

5.6 Yapılan Çalışmalar

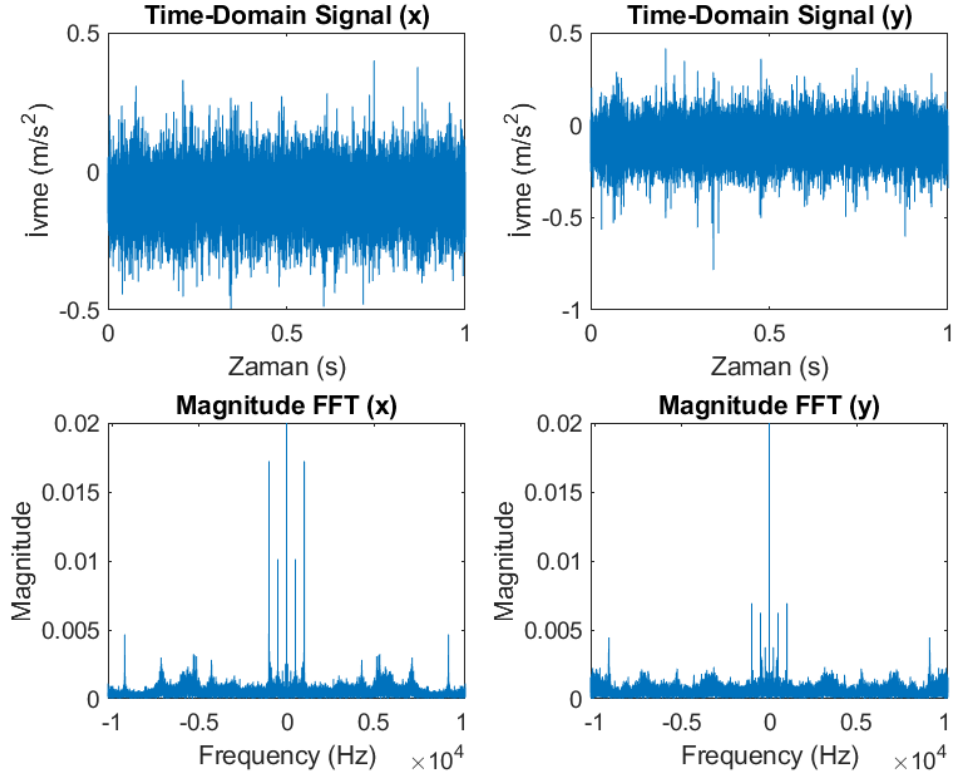
5.6.1 Frekans Alanında Çalışmalar

Üzerinde çalışılan veri kümesinde [41], toplamda 2156 adet dosya bulunmaktadır. Her bir dosyanın incelenmesi ve tez çalışmasına eklenmesi, ortaya 2156 adet grafik çıkaracağından dolayı sadece Tablo 5.3.'te belirtilen zaman aralıklarındaki grafikler çalışmaya eklenmiştir.

Şekil 5.2 ile Şekil 5.18 arasındaki grafiklere baktığımız zaman rulmanlar üzerinde yıpranma başladığı zaman, rulmanların X ve Y eksenlerinde oluşan ivme miktarlarının arttığını görmekteyiz. İvme zaman grafikleri üzerinden olası bir arızanın önceden tahmin edilmesi mümkündür.

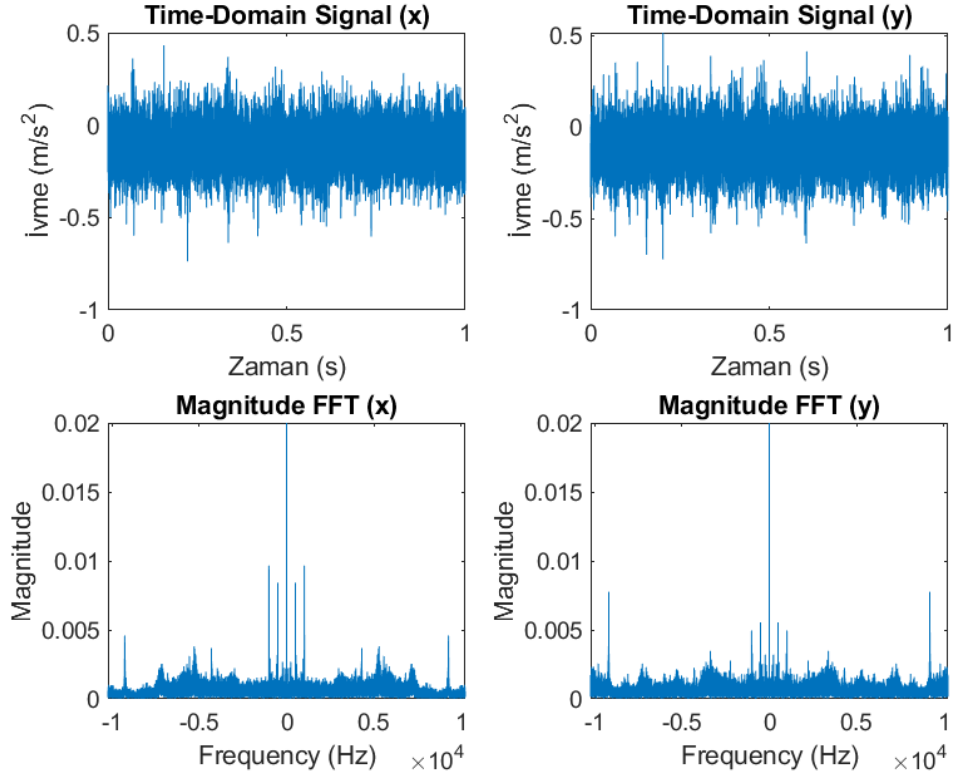
Olası arızanın önceden tahmini, ivme – zaman grafiklerine fourier dönüşümü uygulanarak, frekans alanına dönüştürülmesiyle de yapılabilmektedir. Frekans ve zaman alanlarında yapılacak kestirimlerin avantaj ve dezavantajları karşılaştırılıp anlatılacaktır.

Aşağıdaki grafik ve çalışmalar, 3. Rulman üzerindeki sensörlerden gelen veriler kullanılarak oluşturulmuştur. Tablo 5.3'te belirtilen evrelerin görüldüğü rulman üzerindeki sensörlerden elde edilen grafikler aşağıda incelenmiştir.



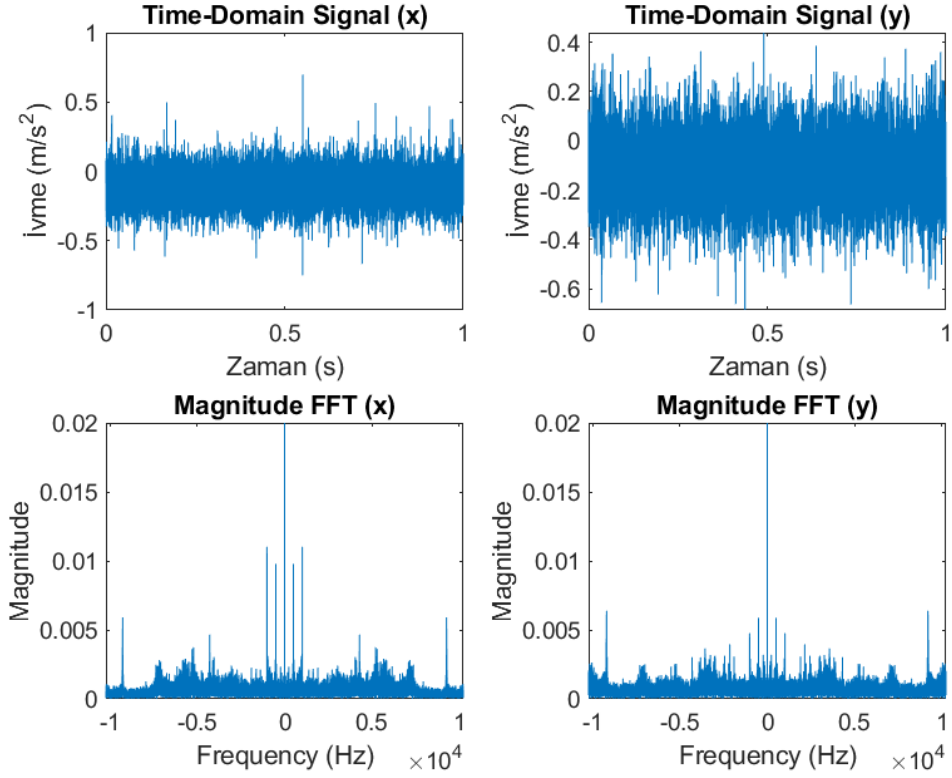
Şekil 5.19: Deneş Başlangıcında 3. Rulman

Şekil 5.19'da deneş başlangıcında 3. Rulman üzerinden alınan veriler incelenmiştir. Frekans alanındaki grafiklerde, frekans grafiklerinde hesaplanan rulman karakteristik frekansları açıkça görölmektedir. Örneğın, 9kHz ve -9kHz üzerinde görölen genlik, 297Hz'lik BPFİ genliğidir.



Şekil 5.20: 3. Rulman Normal Evre

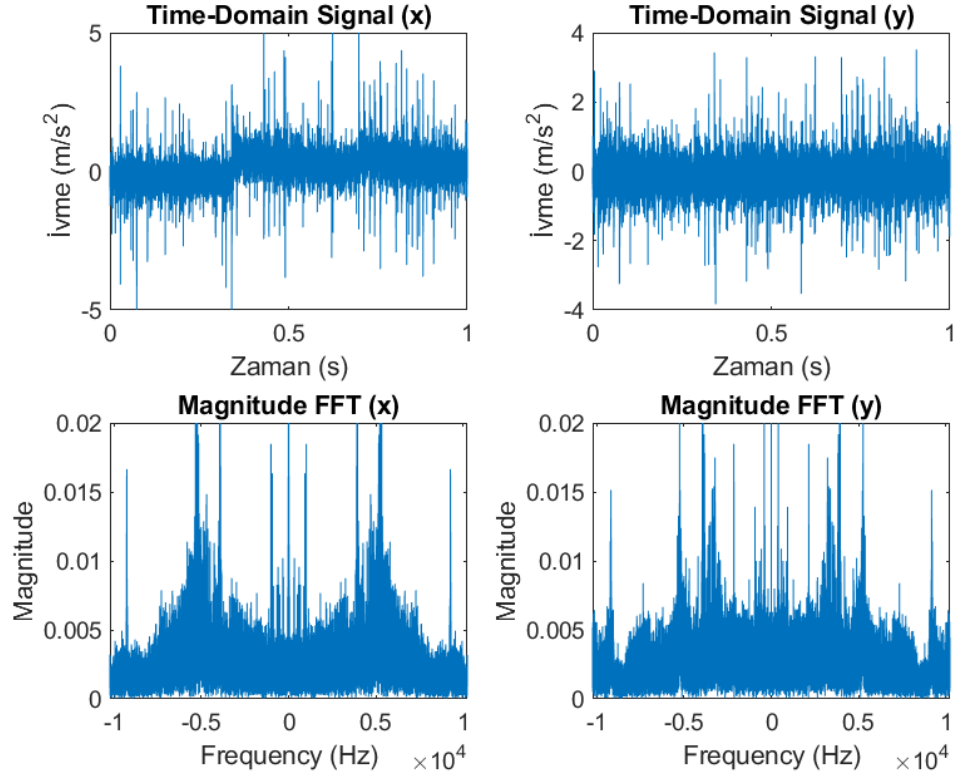
Şekil 5.20'de deney başlangıcından yaklaşık 8 – 9 gün geçmiş ve rulman üzerinde yıpranma belirtileri görülmeye başlanmıştır. İvme – zaman ve frekans genlik grafiklerinde oluşan değişimler gözükmemektedir. Rulman üzerindeki yıpranma sorun yaratacak seviyede olmadığından normal evre olarak adlandırılmıştır. Normal evrede elde edilen grafiği Şekil 5.19 ile karşılaştıracak olursak, rulmanın Y eksenindeki ivme sensöründen alınan verilerle 9kHz ve -9kHz üzerinde görülen BPFİ genliğinin 0.005 seviyesinin üzerine çıktığını, yani iç yatak üzerine yıpranma oluştuğunu görmekteyiz.



Şekil 5.21: 3. Rulman Arıza Şüphesi

Şekil 5.21’de rulmanın arıza yapma ihtimalinin başlangıcını görülmektedir. Deney başlangıcı ve normal evrede frekans grafiğine baktığımızda rulmanın karakteristik frekanslarındaki genliklerini görmekteyiz. Bu grafikte ise karakteristik frekansların genliğinin artmasının yanı sıra karakteristik frekansların yan bantlarının da genliğinin arttığını görmekteyiz.

Bu aşamada rulmanda arıza oluşacağını önceden tahmin edebilmekteyiz. Kestirimci bakım uygulamalarında üzerinde durulması gereken nokta ise arızalanacak rulmanın ne zaman değiştirilmesi gerektiğidir. Bu durumda iki seçenek karşımıza çıkmaktadır. Rulmanın bulunduğu makinenin asla durmaması gerekiyorsa hemen bir bakım planı yapıp arızalanması muhtemel rulman değiştirilebilir, bu daha kullanım ömrü olan rulmanın erken değiştirilmesi demektir. Diğer seçenek ise rulman arızalandığı zaman değiştirmektir. Bu durumda ise makinelerde plansız arıza duruşlar oluşabilmektedir. Plansız duruşlar özellikle imalat sektöründe istenilmeyen bir durumdur.



Şekil 5.22: 3. Rulman Arıza Durumu

Şekil 5.22’de rulmanın artık arızalandığını görmekteyiz. Zaman ve frekans alanlarında genliklerin ciddi oranda artmış olduğu görülmektedir. Bu durum kabul edilemez durumdur. Makine tam anlamıyla durmasa bile hassasiyet kaybı, gürültülü çalışma veya bu arızadan dolayı başka parçaların arızalanması kaçınılmazdır.

Bu tez çalışmasının asıl amacı, makinelerin bu aşamaya gelmeden arızanın fark edilip bakım planlaması yapılması ve minimum duruş ile makinenin tekrar çalışır hale getirilmesidir.

Tablo 5.6’da bulunan karakteristik frekanslar 5.1, 5.2, 5.3 ve 5.4 numaralı formüller ile elde edilmiştir. MATLAB üzerinde oluşturulan grafiklerde ise, Tablo 5.6’da bulunan frekanslarda bir miktar kayma olduğu gözlemlenmiştir (Tablo 5.7).

Tablo 5.7: Teori ve Ölçüm Farkları

	Hesaplanan	Ölçülen	Fark
Şaft Frekansı	33.3 Hz	33.7 Hz	1.2 %
BPFO	236 Hz	240 Hz	1.7 %
BPFI	297 Hz	309 Hz	4.04 %
BSF	140 Hz	143 Hz	2.1 %
FTF	15 Hz	16.9 Hz	12.6 %

Bu çalışmada kestirimci bakım çalışmaları için frekans alanı üzerinde çalışmalar yapılmıştır. Tablo 5.7’de bulunan frekanslar baz alınarak, frekanslar üzerine belirli bir tolerans eklenerek inceleme yapılmıştır. Örneğin, 236 Hz olarak hesaplanan BPFO frekansına $\pm 17\%$ frekans eklenerek 202 Hz – 270 Hz aralığında kestirim için incelenecek frekans bandı oluşturulmuştur. Bu bant içerisinde kalan tüm genliklerin kareleri toplanmış, toplamın da kare kökü alınmıştır. Bu durumda belirlenen bandın gücü hesaplanmıştır. Erken evre ile normal evre güçleri, şüpheli evre ile arıza evreleri ile karşılaştırılmıştır. Bu karşılaştırma sonrası bir güç eşiği belirlenmiş, gelen sinyaller bu ortalamayı geçtiği durumlarda muhtemel arıza için uyarı oluşturulmuştur.

Veri kümesindeki tüm dosyalar teker teker incelenmiş ve Tablo 5.7’de bahsedilmiş frekanslar incelenmiştir. Örneğin 297 Hz olarak belirlenen BPFI’nın frekans aralığında hesaplanan güç, belirlenmiş eşik değerini aşarsa, iç yatak arızası oluşabileceğini önceden anlamaktayız.

Bu kestirim methodu Tablo 5.7’de belirtilmiş tüm frekanslar üzerinde çalışılmıştır. Kısaca açıklamak gerekirse, hangi frekans üzerinde genlik artmaya başlıyor ise o frekansa karşılık gelen arıza başlangıcı tahmin edilmektedir.

Tablo 5.8: Algoritma Başarı Oranı

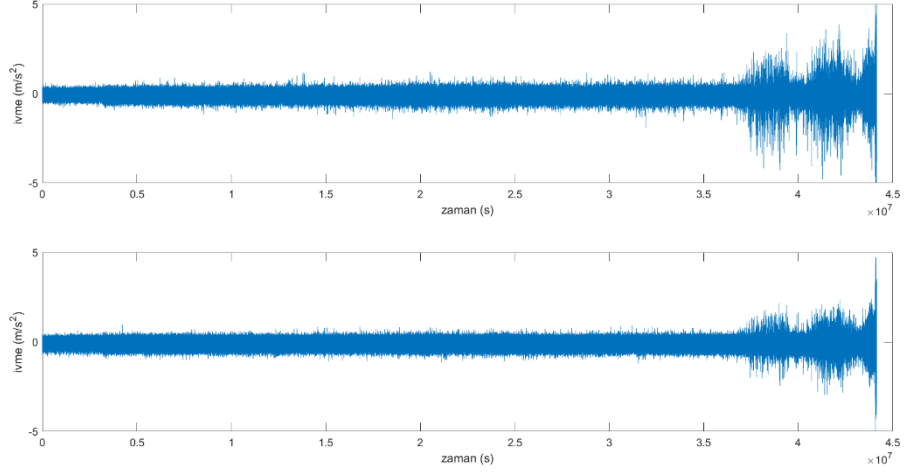
	Toplam Veri Dosyası	Başarı Oranı
Erken Evre	499 dosya	100 %
Normal Evre	1291 dosya	100 %
Şüpheli Evre	330 dosya	80 %
İç Yatak Arızası	36 dosya	100 %

MATLAB üzerine geliştirilen algoritma çalıştırılmış ve veri kümesi içerisindeki 2156 dosya tek tek incelenmiştir. Bu algoritma ile veri kümesi içinde bulunan erken, normal ve arıza evrelerinde 100 % lük kestirim başarıları yakalanmıştır. Şüpheli evre içerisinde bulunan 330 dosyanın 264'ü tespit edilmiştir. 66 dosyanın tespit edilememesinin sebebi, frekans aralıklarındaki gücün belirlenen eşik değerinin altında kalmasıdır. Eşik değeri düşürülerek bu verilerin yakalanması sağlanabilir. Fakat, bu durumda da olması gerekenden çok önce arıza bildirim yapılacaktır. Bu durum, daha kullanım ömrü olan rulmanın ekonomik ömrü dolmadan değiştirilmesine neden olacaktır.

Endüstride arızaların önceden tahmin edilmesinin önemli olduğu kadar, bakım maliyetleri de kritik öneme sahiptir. Bu durumda bahsedilen eşik değerinin iyi belirlenmesi gerekmektedir. Eşik değeri yükseltilerek arıza tahmini daha ileriye çekilebilir. Böyle durumda da rulmanların kullanım ömrünün azalacağını düşünürsek, bakım çalışması için kalan süre de azalmış olacaktır.

5.6.2 Zaman Alanında Çalışmalar

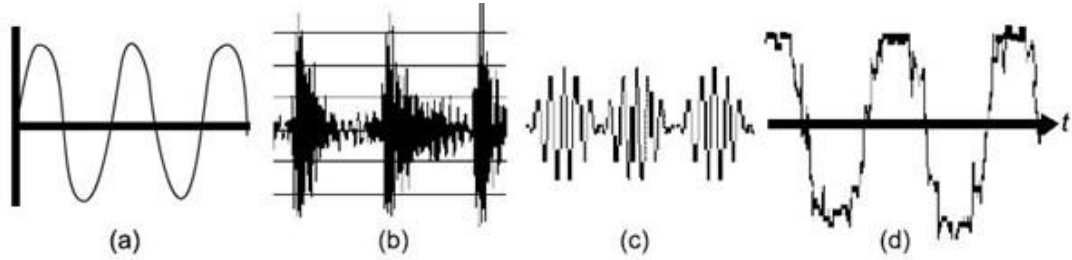
Bu bölümde ivme – zaman grafikleri üzerinden kestirimci bakım çalışmaları anlatılacaktır. Zaman alanındaki çalışmalar yine MATLAB üzerinde gerçekleştirilmiştir.



Şekil 5.23: 3. Rulman İvme-Zaman Grafiği

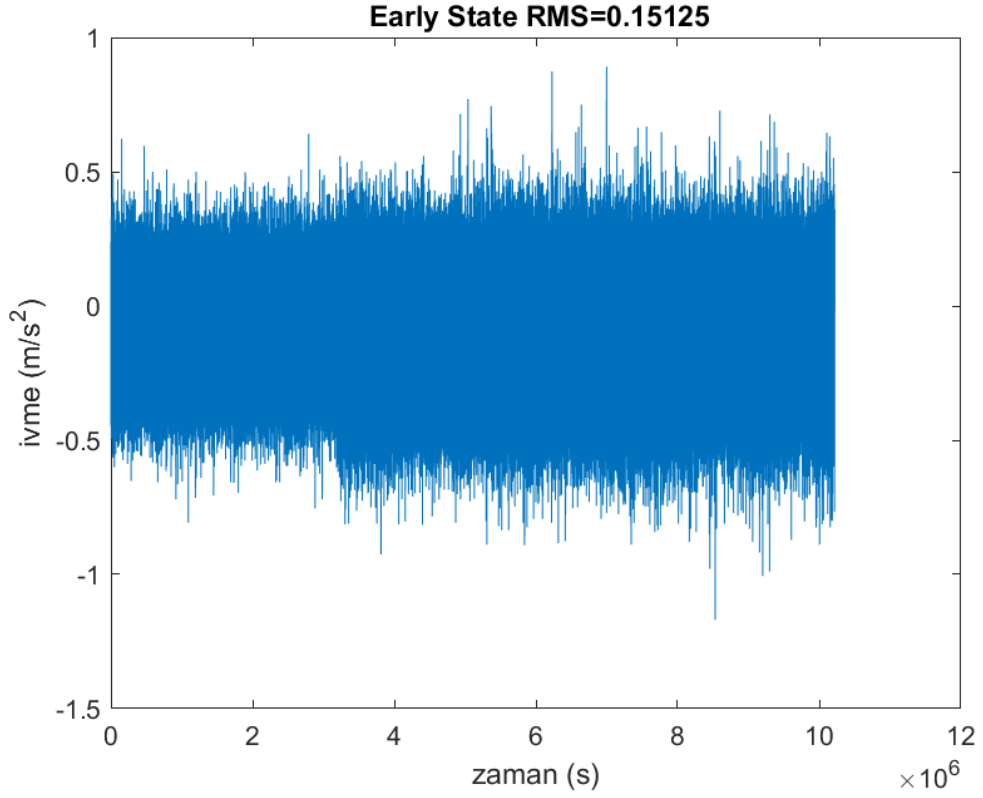
Şekil 5.23'te üstte 3. Rulmanın X eksenindeki sensörden elde edilen veriler, altta ise Y eksenindeki sensörden elde edilen veriler gösterilmiştir. Bu grafik deneyin başlangıcından sonuna kadar elde edilen verileri göstermektedir. Deney uzunluğu yaklaşık olarak bir ay sürmüştür. Grafiğin geneline baktığımızda arızanın başlangıcı net bir şekilde görülmektedir. Grafik, Tablo 5.3'te bulunan evrelere bakılarak detaylı olarak incelenmiştir.

İvme-Zaman grafiğinden görülebilecek arıza belirtileri aşağıdaki gibidir.



Şekil 5.24: a) Dengesizlik b) Yatak Hasarı c) Vuruş d) Gevşeklik

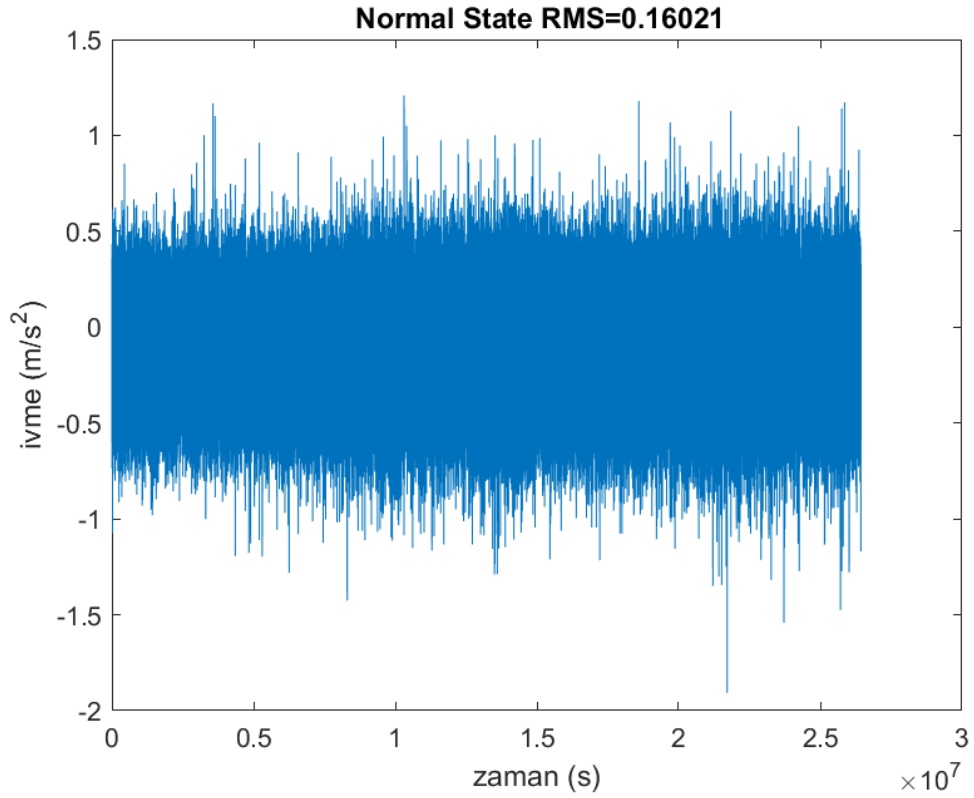
5.6.2.1 Erken Evre



Şekil 5.25: Erken Evre a-t Grafiği

Titreşim analizi yapılırken RMS (Root Mean Square) kullanılmaktadır. RMS, titreşimin enerjisiyle, daha doğrusu titreşimin yıkıcı etkisiyle ilgili bilgi vermektedir. Bu nedenle RMS, titreşim analizinde yağın olarak kullanılmaktadır. Şekil 5.25'te deney başlangıcında titreşim grafiği görülmektedir. Erken evrenin RMS değeri MATLAB üzerinde hesaplanmış ve grafiğe eklenmiştir.

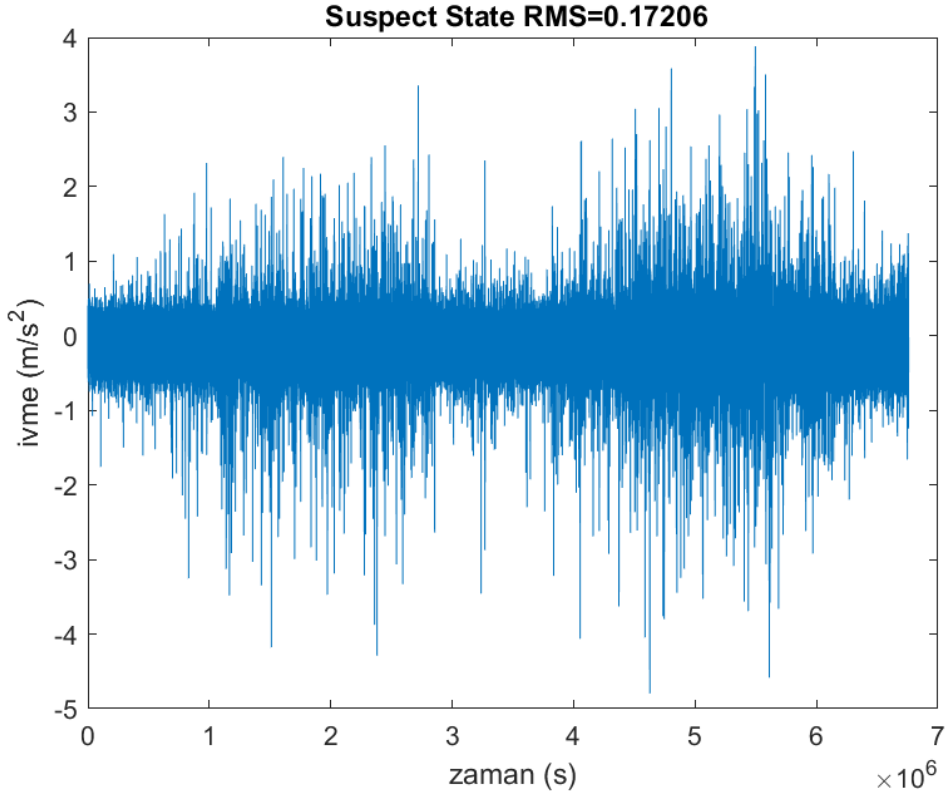
5.6.2.2 Normal Evre



Şekil 5.26: Normal Evre a-t Grafiği

Rulmanda yıpranma belirtileri görülmeye başladığında grafik üzerinde ivmenin arttığını görmekteyiz. RMS değerinde küçük bir artış olduğu da görülmektedir. Grafiğin geneline baktığımızda, erken evre ile benzer olduğu, herhangi bir bozulma olmadığı görülmektedir.

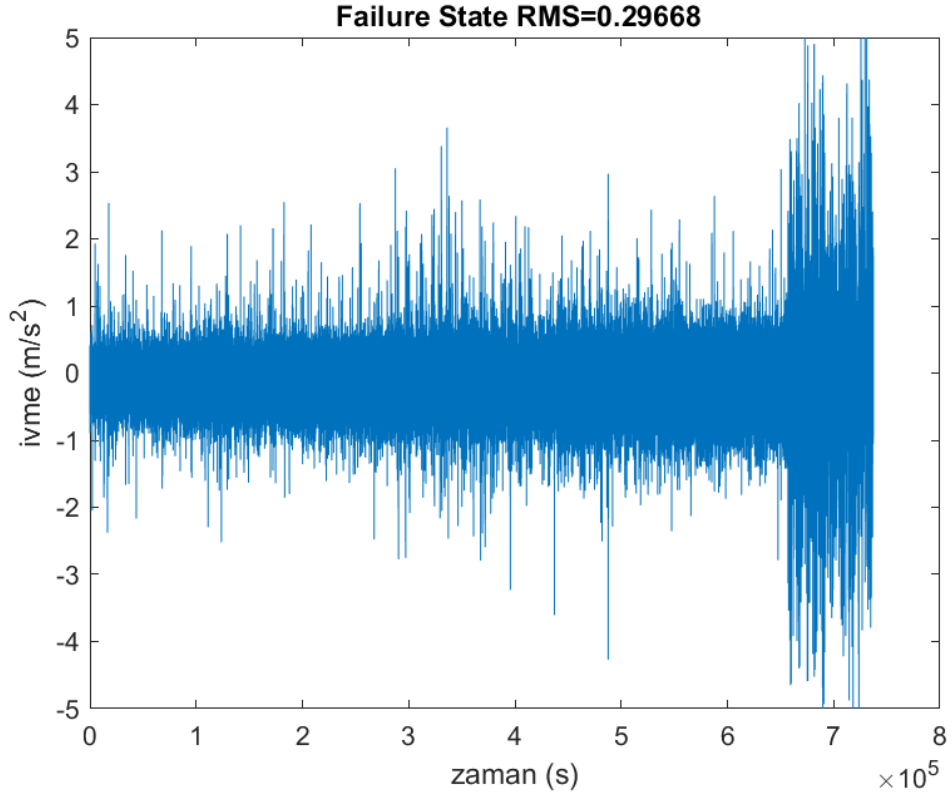
5.6.2.3 Şüpheli Evre



Şekil 5.27: Arıza Şüphesi a-t Grafiği

Şüpheli evrede ise grafik şeklinde bozulma olduğu görülmektedir. Veri aldığımız rulman, deney setindeki 3. Rulman ve deney sonunda bu rulmanda iç yatak arızası oluşacağını bilmekteyiz. Şekil 5.27'ye baktığımızda grafiğin Şekil 5.24 (b)'ye benzediğini görmekteyiz. Sadece grafiği yorumlayarak bu rulmanda yatak arızası oluşacağını tahmin etmemiz mümkün. Oluşan grafikte hasarlı yatağın frekansına veya BSF frekansına eşit olarak tekrarlayan tepe noktalar (peak) oluşmaktadır. RMS değerinde de artış devam etmektedir.

5.6.2.4 İç Yatak Arızası



Şekil 5.28: Rulman Arıza Durumunda a-t Grafiği

Bu durumda tepe noktalarının genliklerinin giderek arttığını, arızanın gerçekleşmesinin yakın olduğunu görmekteyiz. Rulmanın arızalanacağı zaman, RMS değerinin deney başlangıcındaki değer iki katına çıktığını görmekteyiz.

6. SONUÇLAR ve KARŞILAŞTIRMALAR

Bu çalışmada MQTT protokolü incelenmiştir. MQTT yapısı gereği uygulaması ve kullanması oldukça kolay bir protokoldür. Büyük veri toplanması için MQTT protokolüyle çalışan cihazlardan veriler toplanıp daha sonra kestirimci bakım çalışmaları için kullanılabilir. Kestirimci bakım sayesinde üretim tesislerinde bulunan makinelerin, arıza nedeniyle oluşacak plansız duruşlarının ve buna bağlı olarak üretim kayıplarının önüne geçilmesi hedeflenmektedir.

Makineleri oluşturan ekipmanlar, uygulanan kuvvete karşı karakteristik frekanslara sahiptir. Bu frekanslara bakılarak makine arızalarını teşhis etmek mümkündür. Çalışmada zaman ve frekans alanlarında yapılan arıza tahmin çalışmaları kıyaslanmıştır. Çalışma sonucunda, makineler üzerinde oluşabilecek çeşitli arızaların belirli spektrumlar oluşturduğu ve bu arızaların spektrumlar incelenerek arıza oluşmadan fark edilebileceği görülmüştür.

Zaman alanından elde edilen veriler kestirimci bakım çalışmaları için muhtemel arıza başlangıcını belirlemede yardımcı olmaktadır. Fakat zaman alanına bakarak kesin bir yorum yapmak mümkün değildir. Zaman alanında kestirimci bakım çalışması yapmak basit bir yöntemdir fakat sorunun temel nedenlerini bulmak istiyorsak zaman alanında çalışmak yerine frekans alanında çalışmak gereklidir. Zaman alanındaki veriler frekans alanına dönüştürülerek arıza spektrumları izlenebilir ve hangi arızanın neden meydana geldiği veya hangi arızanın ne sebepten dolayı meydana geleceği yorumlanabilmektedir.

Yapılan MATLAB uygulamasında, zaman alanında grafik üzerinde ve RMS değerler incelenerek meydana gelecek arızanın önceden tahmin edilebileceği görülmüştür. Frekans alanında ise kullanılan makine ekipmanlarının karakteristik frekansları gözlemlenerek oluşacak arızanın nereden kaynaklanacağını görmemiz mümkündür. Büyük veri analizi yapılacak ise, frekans alanında çalışılması daha kesin sonuçlar verecektir. Bir makinenin arıza yaptığı durumda büyük veri analizi yapılabilir, frekans alanında çalışmalar yapıp arızanın ne sebepten dolayı oluştuğu öğrenilebilir.

6.1 Literatür Karşılaştırması

[22]'de pompalar üzerinde titreşim analiziyle benzer bir çalışma yapılmıştır. Pompa rulmanlarından alınan veriler, bu çalışmada yaptığım gibi frekans analizi yapılarak incelenmiştir. Çalışmada pompalarda bulunan ayarsızlıklar fark edilmiş ve oluşabilecek çeşitli arızalar, ayarsızlıkların ortadan kaldırılmasıyla önlenmiştir. Yaptığım çalışmada da shaft frekansı incelenerek sistem üzerinde bulunan dengesizlikler fark edilebilir ve bu sebepten dolayı oluşabilecek arızalar önlenebilmektedir.

[38]'de de bu çalışmada kullanılan veri kümesi kullanılmıştır. Deep Neural Networks (DNN) kullanılarak derin öğrenme yöntemiyle kestirim yapılmıştır. Derin sinir ağları, giriş ve çıkış katmanları arasında birden çok katman bulunan yapay bir sinir ağıdır. İnsan beynine benzer şekilde çalışan bu bileşenler, diğer makine öğrenmesi uygulamaları gibi eğitilebilmektedir.

[38]'de kullanılan bu makine öğrenmesi methodu ile tüm evrelerde 94% lik bir başarı oranı yakalanmıştır. Aynı çalışmada bu method farklı bir deney düzeneği üzerinde kullanılmış ve yaklaşık 99% lik bir başarı oranı elde edilmiştir.

Bu çalışmada 80% başarı oranı olan şüpheli arıza evresi, bu yöntem kullanılarak daha verimli kestirim yapacak hale getirilebilir. İleride bu yöntem kullanılarak kestirimci bakım çalışmaları yapılacak ve çıkan sonuçları incelenerek başarı oranlarını karşılaştırılacaktır.

[39]'da da bu çalışmada kullanılan veri kümesi üzerinden çalışma yapılmıştır. Bu çalışmada ise Hjorth tanımlayıcıları kullanılarak rulman analizi yapılmıştır. Hjorth tanımlayıcıları Etkinlik, Hareketlilik ve Karmaşıklık olarak tanımlanmıştır.

Etkinliğin, bir rulman arızası ile bağlantılı olarak artan bir eğilim gösterdiği belirtilmiştir. Etkinlik, sinyalin ortalama gücü ile doğrudan ilişkili olduğu için, bir yataktaki bir hasar, önce titreşimde, ardından sinyalin gücünde bir artışa neden olmaktadır. Hareketliğin ise [39]'da rulman hasarları ile bağlantılı olarak çok net bir davranış göstermediği belirtilmiştir. Bazı durumlarda artan bir eğilim göstermiş, bazı durumlarda ise azalan bir eğilim göstermiştir. Karmaşıklığın ise frekans boyunca sinyal spektrumunun integrali ile ilgili olduğu ve sinyal spektrumu kesikli olduğunda

minimum değerine ulaştığı belirtilmiştir. Bir arıza meydana geldiğinde arızaya ait frekans artarken sinyalin karmaşıklığının azaldığı görülmüştür.

[39]'da bu çalışmada olduğu gibi rulman arızasının önceden fark edilebildiği görülmektedir. Çalışmanın başarı oranından bahsedilmediği için bir kıyaslama yapılamamaktadır. Bu yöntemin kullanıldığı yeni bir algoritma ile ileride bir çalışma yapılabilir ve başarı oranı hesaplanabilir.

[40]'da MQTT, CoAP ve AMQP protokolleri arasında karşılaştırma yapılmıştır. MQTT sahip olduğu QoS seviyeleri sayesinde mesaj güvenliğinin sağlandığı belirtilmiştir. İletişim için kolay yöntemler kullanıldığı, düğümler arasında asenkron iletişim olduğu ve mesajların her zaman yayınlanabildiği ve aboneliğin gerçekleştirilebilir olduğu görülmektedir. Bu protokolün en büyük dezavantajı ise; Broker'da bir arıza veya sorun olduğu zaman bütün bağlantılı istemciler arasındaki bağlantı kopmaktadır.

CoAP protokolünde ise Datagram Transport Layer Security (DTLS) güvenlik ve gizliliği sağlamaktadır. Kısıtlı kaynağa sahip cihazlar ile kullanılmaya uygun olduğu belirtilmektedir. Ancak, bu protokolde mesaj iletimi güvenilmezdir. Mesajların doğru geldiğini doğrulamak için bir onay paketi gönderilmektedir. Fakat, bu paket, mesajların şifrelemesinin doğru veya tam olarak çözüldüğünü göstermemektedir. MQTT protokolünde ise QoS seviyeleri sayesinde mesajın tam olarak iletilmesi garanti altına alınmaktadır.

AMQP protokolünün ise MQTT protokolüyle benzer özellikler taşıdığı görülmektedir. MQTT'den farklı olarak farklı standartlarda çalışacak şekilde geliştirilmesi mümkündür. Bu protokolün güvenliği ise SSL ile sağlanmaktadır. Bu protokolün en dikkat çeken dezavantajları ise diğer protokollere kıyasla daha yüksek bant genişliği gerektirmesi ve diğer protokoller kadar basit yapıda olamasıdır [40].

MQTT protokolünün mesaj iletilmesini garanti altına alan QoS seviyelerine sahip olması ve basit yapıda olmasıdır. Basit yapıda olması sayesinde düşük kaynaklı istemciler üzerinde sorunsuzca kullanılabilir. Sahip olduğu "Retained Messages" ve "Last Will and Testament" özellikleri sayesinde ağ bağlantısının düzensiz olduğu koşullarda bile istenilen mesajların abonelere iletilmesi ve beklenmedik kopmalarda bilgi mesajının istenilen abonelere iletilmesi sağlanmaktadır.

7. İLERİDE YAPILMASI PLANLANAN ÇALIŞMALAR

Bu çalışma, sabit hızda dönen bir mil üzerinde bulunan rulmanlardan elde edilen veriler kullanılarak yapılmıştır. Bu çalışmanın devamı olarak, devir yönü ve hızı değişen miller üzerinde bir çalışma yapılması planlanmaktadır. Bu durumda değişken hızlarda çalışan makineler üzerinde oluşacak rulman arızalarını önceden tahmin edilebilmesi öngörülmektedir.

İleride yapılması planlanan bir diğer çalışma ise lineer rulmanlar ve lineer arabalar üzerinde arıza tahmini çalışmasıdır. Lineer rulman ve arabalar, bu çalışmada incelenen bilyeli rulmanlardan farklı olarak doğrusal düzlemlerde ileri ve geri hareket etmektedir. Bu tür rulman ve arabalarda hareket sabit hızlı ve tek yönlü olmadığı için bu çalışmayı bir adım öteye taşıyacağı öngörülmektedir.

Çalışmada kullanılan MQTT haberleşme protokolü yerine diğer IoT protokolleri ile benzer çalışma yapılarak protokoller arasındaki avantaj ve dezavantajlar etkin biçimde gözlemlenebilir. Farklı protokoller kullanıldığı takdirde protokollerin; hız farkları, verilerin teslim edilme başarı oranları, kablolu veya kablosuz ağlarda nasıl performans sergiledikleri, veri güvenlikleri ve stabil ağ bağlantılarının olmadığı durumlarda hangi protokolün daha başarılı olacağını sonuçları ortaya çıkacaktır.

Bu çalışmada frekans analizi üzerinde durulmuştur. İleriki çalışmalarda ise SVM, K-ortalılar ve KNN algoritmaları gibi makine öğrenmesi algoritmaları kullanılarak arıza tahmini gerçekleştirilebilir. Farklı algoritmalar kullanılarak yapılacak kestirimci bakım çalışmaları ile algoritmaların başarı oranları hesaplanabilir ve hangi algoritmanın hangi makinede kullanılabileceği belirlenebilecektir.

Bu çalışmada kullanılan veri kümeleri [41] yerine, aktif olarak çalışan makineler üzerine sensörler yerleştirilebilir ve gerçek zamanlı canlı verilerin takibi yapılabilir. Titreşim analizi için ivme sensörleri, yağlı çalışan parçalarda yağ kalitesini ölçmek için yağ viskozite sensörleri, çeşitli motorların veya hareketli parçaların sıcaklıklarını ölçmek için kızıl ötesi ısı sensörleri veya termokupllar kullanılabilir. Böylece bir

makinenin çeşitli kritik parçaları üzerinde oluşabilecek arızalar ve bu parçaların kalan kullanım ömürleri tahmin edilebilecektir.

REFERANSLAR

- [1] **Krajak, S., & Tuwanut, P.** (2015, October). A survey on internet of things architecture, protocols, possible applications, security, privacy, real-world implementation and future trends. In *Communication Technology (ICCT), 2015 IEEE 16th International Conference on* (pp. 26-31). IEEE.
- [2] **Park, J. S., & Shon, J. G.** (2016b). Design and Implementation of a Reliable Message Transmission System Based on MQTT Protocol in IoT. *Wireless Personal Communications*, 91(4), 1765–1777. <https://doi.org/10.1007/s11277-016-3398-2>
- [3] **S. Katsikeas et al.**, "Lightweight & secure industrial IoT communications via the MQ telemetry transport protocol," 2017 IEEE Symposium on Computers and Communications (ISCC), Heraklion, 2017, pp. 1193-1200, doi: 10.1109/ISCC.2017.8024687.
- [4] **J. McKendrick**, *IoT-Connected Devices Now Outnumber Non-Connected*, Real-Time Insights, Aralık 2020. Erişim: 16 Ocak 2021. [Online]. Bağlantı: <https://www.rtinsights.com/iot-connected-devices-now-outnumber-non-connected/#:~:text=Of%20the%202021.7%20billion%20active,within%20the%20next%20five%20years>.
- [5] **Madakam, S., Ramaswamy, R. and Tripathi, S.** (2015) Internet of Things (IoT): A Literature Review. *Journal of Computer and Communications*, 3, 164-173. <http://dx.doi.org/10.4236/jcc.2015.35021>
- [6] **Lianos, M. and Douglas, M.** (2000) Dangerization and the End of Deviance: The Institutional Environment. *British Journal of Criminology*, 40, 261-278. <http://dx.doi.org/10.1093/bjc/40.2.261>
- [7] **Ferguson, T.** (2002) Have Your Objects Call My Object. *Harvard Business Review*, June, 1-7.
- [8] **Yuan, M.** (2017, May 12). Getting to know MQTT. IBM. <https://developer.ibm.com/articles/iot-mqtt-why-good-for-iot/>
- [9] **Miorandi, D., Sicari, S., Pellegrini, F.D., Chlamtac, I.**, 2012. Internet of things: Vision, applications and research challenges. *Journal of Ad Hoc Networks*, 10, 1497-1516.
- [10] **Ercan, T., & Kutay, M.** (2016). Internet of Things (IoT) Applications in Industry. *Afyon Kocatepe University Journal of Sciences and Engineering*, 16(3), 599–607. <https://doi.org/10.5578/fmbd.43411>
- [11] **Li, D.X., Wu, H., Shangang, H.**, 2014. Internet of things in industries: A Survey. *IEEE Transactions on Industrial Informatics*, 10, 2233-2243.

- [12] **Xu, L.D.**, 2011. Enterprise systems: State-of-the art and future trends. *IEEE Transactions on Industrial Informatics*, 7, 630-641.
- [13] **Baxter, K.** (2018, August 28). Fourth Industrial Revolution [Illustration]. MEDIUM. <https://medium.com/salesforce-ux/human-rights-in-the-fourth-industrial-revolution-industrys-role-and-responsibilities-7aa07fbe255d>
- [14] **Hermann, M., Pentek, T., Otto, B.**, 2015. Design Principles for Industrie 4.0 Scenarios: A Literature Review. (Working Paper) Technische Universitat, Dortmund.
- [15] **Kleijn, S.** (2020, January 14). A schematic overview of an Industrial IoT platform [Illustration]. IXON. <https://www.ixon.cloud/knowledge-hub/10-criteria-to-select-the-best-industrial-iiot-solution>
- [16] **Khan, W. Z., Rehman, M. H., Zangoti, H. M., Afzal, M. K., Armi, N., & Salah, K.** (2020). Industrial internet of things: Recent advances, enabling technologies and open challenges. *Computers & Electrical Engineering*, 81, 106522. <https://doi.org/10.1016/j.compeleceng.2019.106522>
- [17] **Lampkin, V., Leong, W. T., Olivera, L., Rawat, S., Subrahmanyam, N., Xiang, R., ... & Locke, D.** (2012). Building smarter planet solutions with mqtt and ibm websphere mq telemetry. IBM Redbooks.
- [18] **Lee, S., Kim, H., Hong, D. K., & Ju, H.** (2013, January). Correlation analysis of MQTT loss and delay according to QoS level. In *Information Networking (ICOIN), 2013 International Conference on* (pp. 714-717). IEEE.
- [19] **Banks, A., & Gupta, R.** (2014). MQTT Version 3.1. 1. OASIS standard.
- [20] **D.J. Edwards, G.D. Holt, F.C. Harris**, Predictive maintenance techniques and their relevance to construction plant, *Journal of Quality in Maintenance Engineering* 4 (1) (1998) 25– 37.
- [21] **Ulusoy, A.** (2021, May 6). How is Fault Detection Performed? Sensemore. <https://sensemore.io/tr/ariza-tespiti-nasil-yapilir/>
- [22] **Yaman, G. ş., & Karadayı, H. M.** (2014). Titreşim Analizi ile Pompalarda Arıza Tesbiti ve Kestirimci Bakım İçin Örnek Bir Çalışma. Titreşim Analizi ile Pompalarda Arıza Tesbiti ve Kestirimci Bakım İçin Örnek Bir Çalışma.
- [23] **Orhan, S., Aktürk, N., & Çelik, V.** (2006). Vibration monitoring for defect diagnosis of rolling element bearings as a predictive maintenance tool: Comprehensive case studies. *NDT & E International*, 39(4), 293–298. <https://doi.org/10.1016/j.ndteint.2005.08.008>
- [24] **Zonta, T., da Costa, C. A., da Rosa Righi, R., de Lima, M. J., da Trindade, E. S., & Li, G. P.** (2020). Predictive maintenance in the Industry 4.0: A systematic literature review. *Computers & Industrial Engineering*, 150, 106889. <https://doi.org/10.1016/j.cie.2020.106889>
- [25] **Lee, C. K. M., Cao, Y., & Ng, K. H.** (2017). Big Data Analytics for Predictive Maintenance Strategies. *Supply Chain Management in the Big Data Era*, 50–74. <https://doi.org/10.4018/978-1-5225-0956-1.ch004>

- [26] **Van Horenbeek, A., & Pintelon, L.** (2013). A dynamic predictive maintenance policy for complex multi-component systems. *Reliability Engineering & System Safety*, 120, 39–50. <https://doi.org/10.1016/j.res.2013.02.029>
- [27] **Dai, H. N., Wang, H., Xu, G., Wan, J., & Imran, M.** (2019). Big data analytics for manufacturing internet of things: opportunities, challenges and enabling technologies. *Enterprise Information Systems*, 14(9–10), 1279–1303. <https://doi.org/10.1080/17517575.2019.1633689>
- [28] **Ghobakhloo, M.** (2020). Industry 4.0, digitization, and opportunities for sustainability. *Journal of Cleaner Production*, 252, 119869. <https://doi.org/10.1016/j.jclepro.2019.119869>
- [29] **Wang, J., Liang, Y., Zheng, Y., Gao, R. X., & Zhang, F.** (2020). An integrated fault diagnosis and prognosis approach for predictive maintenance of wind turbine bearing with limited samples. *Renewable Energy*, 145, 642–650. <https://doi.org/10.1016/j.renene.2019.06.103>
- [30] **Shamayleh, A., Awad, M., & Farhat, J.** (2020). IoT Based Predictive Maintenance Management of Medical Equipment. *Journal of Medical Systems*, 44(4). <https://doi.org/10.1007/s10916-020-1534-8>
- [31] **Wu, L., & Zhou, Q.** (2020). Adaptive sequential predictive maintenance policy with nonperiodic inspection for hard failures. *Quality and Reliability Engineering International*, 37(3), 1173–1185. <https://doi.org/10.1002/qre.2788>
- [32] **Thoben, K. D., Wiesner, S., & Wuest, T.** (2017). “Industrie 4.0” and Smart Manufacturing – A Review of Research Issues and Application Examples. *International Journal of Automation Technology*, 11(1), 4–16. <https://doi.org/10.20965/ijat.2017.p0004>
- [33] **Hashemian, H. M., & Bean, W. C.** (2011). State-of-the-Art Predictive Maintenance Techniques*. *IEEE Transactions on Instrumentation and Measurement*, 60(10), 3480–3492. <https://doi.org/10.1109/tim.2009.2036347>
- [34] **Jimenez-Cortadi, A., Irigoien, I., Boto, F., Sierra, B., & Rodriguez, G.** (2019). Predictive Maintenance on the Machining Process and Machine Tool. *Applied Sciences*, 10(1), 224. <https://doi.org/10.3390/app10010224>
- [35] **Sahal, R., Breslin, J. G., & Ali, M. I.** (2020). Big data and stream processing platforms for Industry 4.0 requirements mapping for a predictive maintenance use case. *Journal of Manufacturing Systems*, 54, 138–151. <https://doi.org/10.1016/j.jmsy.2019.11.004>
- [36] **Çınar, Z. M., Abdussalam Nuhu, A., Zeeshan, Q., Korhan, O., Asmael, M., & Safaei, B.** (2020). Machine Learning in Predictive Maintenance towards Sustainable Smart Manufacturing in Industry 4.0. *Sustainability*, 12(19), 8211. <https://doi.org/10.3390/su12198211>
- [37] **Moleda, M., Momot, A., & Mrozek, D.** (2020). Predictive Maintenance of Boiler Feed Water Pumps Using SCADA Data. *Sensors*, 20(2), 571. <https://doi.org/10.3390/s20020571>

- [38] **Zhang, R., Peng, Z., Wu, L., Yao, B., & Guan, Y.** (2017). Fault Diagnosis from Raw Sensor Data Using Deep Neural Networks Considering Temporal Coherence. *Sensors*, 17(3), 549. <https://doi.org/10.3390/s17030549>
- [39] **Analysis of NASA Bearing Dataset of the University of Cincinnati by Means of Hjorth's Parameters / CAVALAGLIO CAMARGO MOLANO, Jacopo; Strozzi, Matteo; Rubini, Riccardo; Cocconcelli, Marco.** - (2019). ((Speech presented at the International Conference on Structural Engineering Dynamics ICEDyn 2019 held in Viana do Castelo (Portugal) on June 24-26, 2019.
- [40] **Naik, G. P., & Bapat, A. U.** (2020). A Brief Comparative Analysis on Application Layer Protocols of Internet of Things: MQTT, CoAP, AMQP and HTTP. *International Journal of Computer Science and Mobile Computing*, 9(9), 135–141. <https://doi.org/10.47760/ijcsmc.2020.v09i09.014>
- [41] **J. Lee, H. Qiu, G. Yu, J. Lin, and Rexnord Technical Services** (2007). IMS, University of Cincinnati. "Bearing Data Set", NASA Ames Prognostics Data Repository (<http://ti.arc.nasa.gov/project/prognostic-data-repository>), NASA Ames Research Center, Moffett Field, CA

ÖZGEÇMİŞ

Adı Soyadı : Ulaş UZUNKAYA

EĞİTİM:

TC İstanbul Aydın Üniversitesi Elektrik – Elektronik Mühendisliği Yüksek Lisans
(Tez aşaması)

TC Yeditepe Üniversitesi Elektrik – Elektronik Mühendisliği Lisans

İŞ DURUMU:

ENTES Elektronik Cihazlar İmalat ve Ticaret A.Ş. SMD Takım Lideri