

T.C.
FIRAT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

İlhan AYDIN

**ARIZA TEŞHİSİNDE VERİ MADENCİLİĞİ VE YUMUŞAK
HESAPLAMA TEKNİKLERİNİN KULLANIMI**

Tez Yöneticisi:
Doç. Dr. Erhan AKIN

YÜKSEK LİSANS TEZİ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

ELAZIĞ, 2006

T.C.
FIRAT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

**ARIZA TEŞHİSİNDE VERİ MADENCİLİĞİ VE YUMUŞAK
HESAPLAMA TEKNİKLERİNİN KULLANIMI**

İlhan AYDIN

YÜKSEK LİSANS TEZİ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

Bu tez, tarihinde aşağıda belirtilen jüri tarafından oybirliği /oyçokluğu ile başarılı / başarısız olarak değerlendirilmiştir.

Danışman : Doç.Dr. Erhan AKIN

Üye :

Üye :

Üye:

Üye:

Bu tezin kabulü, Fen Bilimleri Enstitüsü Yönetim Kurulu'nun/...../..... tarih ve sayılı kararıyla onaylanmıştır.

TEŞEKKÜR

Bu tez çalışmasını yönlendiren ve çalışma süresince yardımlarını esirgemeyen danışman hocam Sayın Doç.Dr. Erhan AKIN'a ve desteklerinden dolayı FÜBAP'a şükranlarımı arz ederim. Ayrıca her türlü yardımlarından dolayı Yrd. Doç. Dr. Mehmet KARAKÖSE'ye en içten teşekkürlerimi sunarım.

İÇİNDEKİLER

TEŞEKKÜR

İÇİNDEKİLER	I
ŞEKİLLER LİSTESİ.....	IV
TABLolar LİSTESİ.....	VII
SİMGELER	VIII
KISALTMALAR	X
ÖZET	XI
ABSTRACT	XII
1. GİRİŞ.....	1
1.1. Asenkron Motorlarda Arıza Teşhisinin Önemi.....	2
1.2. Asenkron Motorlarda Oluşan Temel Arızalar	3
1.3. Asenkron Motorlarda Oluşan Arızaların Teşhisi için Bazı Yaklaşımlar	3
1.3.1. Model Tabanlı Teknikler	4
1.3.2. Uzman Bilgisine Dayalı Teknikler	5
1.4. Bilgisayar Destekli Arıza Teşhis Algoritmalarının Gerekliliği	6
1.5. Asenkron Motor Arızalarının Teşhisi için Yumuşak Hesaplama Tekniklerinin Kullanılması.....	6
1.6. Arıza Teşhisinde Zaman Serileri Veri Madenciliği Yöntemi	12
1.7. Tezin Amacı.....	14
1.8. Tezin Yapısı.....	15
2. ASENKRON MOTORLARIN YAPISI VE BU MOTORLARDA OLUŞAN ARIZA TÜRLERİ	16
2.1. Asenkron Motorların Yapısı ve Çalışma Prensipleri.....	16
2.2. Asenkron Motorlarda Durum İzleme, Arıza Belirleme, Tahmin, Teşhis ve Kestirim Kavramları	20
2.3. Asenkron Motorlarda Oluşan Arıza Türlerinin Sebepleri ve Etkiledikleri Motor Parametreleri.....	21
2.3.1. Mil Yatağı Arızaları.....	22
2.3.2. Rotor Arızaları	22
2.3.3. Hava Boşluğu Eksantrikliği Arızaları.....	23
2.3.4. Stator Arızaları.....	25
3. YUMUŞAK HESAPLAMA TEKNİKLERİ.....	27
3.1. Giriş	27

3.2.	Yapay Sinir Ağları	28
3.2.1.	Sinir Hücresi Modeli.....	30
3.2.2.	İleri Beslemeli Yapay Sinir Ağları	31
3.2.3.	Geriye Yayılım Öğrenme Algoritması	33
3.3.	Bulanık Mantık	35
3.3.1.	Keskin Kümeler, Bulanık Kümeler ve Üyelik Fonksiyonları.....	36
3.3.2.	Temel Küme İşlemleri	38
3.3.3.	Bulanık Çıkarım Sistemi.....	41
3.4.	Bağışık ve Yapay Bağışık Sistemler.....	43
3.4.1.	Bağışık Sistemler	43
3.4.1.1.	Bağışık Sistemlerin Fizyolojisi.....	45
3.4.1.2.	Bağışık Sistemlerde Örüntü Tanıma.....	46
3.4.1.3.	Klonal Seçim Mekanizması ve Afinite Olgunlaşması	47
3.4.1.4.	Öz/Öz Olmayan Ayrımı.....	49
3.4.2.	Yapay Bağışık Sistemler.....	49
3.4.2.1.	Negatif Seçim Algoritması	52
3.4.2.2.	Klonal Seçim Mekanizması	53
4.	VERİ MADENCİLİĞİ.....	55
4.1.	Giriş	55
4.2.	Bilgi Elde Etme Süreci ve Veri Ön İşleme Adımları.....	56
4.3.	Veri Madenciliğinde Karşılaşılan Problemler	57
4.3.1.	Veritabanı Boyutu.....	58
4.3.2.	Gürültülü Veri.....	58
4.3.3.	Geçersiz Değerler	58
4.3.4.	Eksik Veri	59
4.3.5.	Artık Veri.....	59
4.3.6.	Dinamik Veri	59
4.4.	Veri Madenciliği Modelleri	60
4.4.1.	Birliktelik Kuralları	60
4.4.2.	Sınıflandırma ve Tahmin	62
4.4.3.	Kümeleme.....	63
4.5.	Zaman Serileri Veri Madenciliği	64
4.5.1.	Zaman Serilerinde Benzerlik Araması.....	65
4.5.1.1.	Benzerlik Araması için Öklid ve Manhattan Ölçümü.....	66
4.5.1.2.	Benzerlik Araması için Normalleştirme Dönüşümü	66

4.5.1.3.	Genel Dönüşümler	67
4.5.1.4.	Dinamik Zaman Eğriliği	68
4.5.2.	Zaman Serileri için İndeksleme Teknikleri.....	70
4.5.3.	Zaman Serilerinde Veri Azaltma Teknikleri.....	70
4.5.4.	Lineer Olmayan Zaman Serileri Analizi ve Veri Madenciliği Teknikleri	71
4.5.4.1.	Zaman Serileri Veri Madenciliğinde Bazı Kavramlar	72
4.5.4.2.	Zaman Serileri Veri Madenciliği Algoritmasının Genel Çatısı	77
5.	SİMÜLASYON VE DENEYSEL SONUÇLAR.....	82
5.1.	Uygulama Çalışması 1	82
5.1.1.	Arıza Teşhisi için Önerilen Yumuşak Hesaplama Tabanlı Yaklaşım.....	82
5.1.1.1.	İleri Beslemeli Yapay Sinir Ağı Modülü	83
5.1.1.2.	Bulanık Mantık Modülü.....	84
5.1.2.	Simülasyon Sonuçları	87
5.2.	Uygulama Çalışması 2	91
5.2.1.	Arıza Teşhisi için Önerilen Zaman Serileri Veri Madenciliği Yaklaşımı	91
5.2.2.	Uygulama Sonuçları	93
5.3.	Uygulama Çalışması 3	95
5.3.1.	Arıza Teşhisi için Önerilen Bağışıklık Tabanlı Algoritma	95
5.3.2.	Deneysel Verilerin Elde Edilmesi.....	98
5.3.3.	Deneysel Sonuçlar	102
5.4.	Uygulama Çalışması 4	105
5.4.1.	Önerilen Bağışık Sistem Tabanlı Bulanık Arıza Teşhis Algoritması	105
5.4.2.	Simülasyon ve Deneysel Sonuçlar.....	108
6.	SONUÇLAR.....	113
	KAYNAKLAR	116
	ÖZGEÇMİŞ	121
	EK-1	
	EK-2	
	EK-3	

ŞEKİLLERİN LİSTESİ

Şekil 1.1.	Model tabanlı arıza teşhis şeması	4
Şekil 1.2.	Uzman bilgisine dayalı arıza teşhis şeması	5
Şekil 1.3.	Arıza teşhisi için yumuşak hesaplama algoritmalarının kullanımı.....	8
Şekil 2.1.	Bir asenkron motorun yapısı	17
Şekil 2.2.	Bir asenkron motorun çalışma prensibinin gösterimi.....	18
Şekil 2.3.	Tipik bir sincap kafes rotorun yapısı.....	19
Şekil 2.4.	Asenkron motor arıza tiplerinin oluşum yüzdeleri.....	21
Şekil 2.5.	Mil yatağının yapısı.....	22
Şekil 2.6.	Hava boşluğu eksantrikliğinin şematik gösterimi	24
Şekil 3.1.	Yumuşak hesaplama bileşenleri	27
Şekil 3.2.	Biyolojik sinir hücresi	28
Şekil 3.3.	Yapay sinir hücresi modeli.....	30
Şekil 3.4.	Aktivasyon fonksiyonları	31
Şekil 3.5.	İleri beslemeli yapay sinir ağı yapısı.....	32
Şekil 3.6.	Keskin ve bulanık kümeler için üyelik fonksiyonu.....	37
Şekil 3.7.	Normal ve normal olmayan bulanık kümeler için üyelik fonksiyonları.....	37
Şekil 3.8.	Bulanık üyelik fonksiyonları	38
Şekil 3.9.	Örnek bir akım üyelik fonksiyonu	40
Şekil 3.10.	Bulanık mantıkta küme işlemleri	41
Şekil 3.11.	Bulanık çıkarım sisteminin blok diyagramı	42
Şekil 3.12.	Bağıışıklık sisteminin katmanlı yapısı.....	44
Şekil 3.13.	Lenfositler ve yüzey reseptörleri.....	45
Şekil 3.14.	Gen kütüphanesinden bir antikor molekülünün oluşumu.....	46
Şekil 3.15.	Bir antijenin antikor tarafından tanınması.....	47
Şekil 3.16.	Klonal seçim.....	48
Şekil 3.17.	Yapay bağıışık sistemlerin katmanlı yapısı.....	50
Şekil 3.18.	Tümleyen alanlar aracılığıyla tanıma	51
Şekil 3.19.	Şekil uzayının gösterimi.....	51
Şekil 3.20.	Negatif seçim algoritması.....	53
Şekil 3.21.	Klonal seçim algoritmasının temel adımları	54
Şekil 4.1.	Bilgi keşfi sürecinde bir adım olarak veri madenciliği	57
Şekil 4.2.	Veri madenciliği modelleri.....	60
Şekil 4.3.	Birliktelik kuralları için bir örnek	62

Şekil 4.4.	Temel veri sınıflandırma süreci.....	62
Şekil 4.5.	Kümeleme örneği	63
Şekil 4.6.	Zaman serisi örnekleri	65
Şekil 4.7.	Gürültülü ve gürültüsüz zaman serisi.....	67
Şekil 4.8.	Dönüşüm kuralı uygulama örneği	68
Şekil 4.9.	Dönüşüm kurallarının zaman serilerine uygulanması	68
Şekil 4.10.	İki seri arasındaki zaman eğriliği grafiği	69
Şekil 4.11.	Sismik bir zaman serisi	73
Şekil 4.12.	Faz alanı	75
Şekil 4.13.	Zaman serisi veri madenciliği algoritmasının blok şeması	78
Şekil 4.14.	Farklı gömülme boyutları için iki zaman serisi örneği.....	81
Şekil 5.1.	Yumuşak hesaplama tabanlı arıza teşhis algoritması	83
Şekil 5.2.	Arıza teşhisi için kullanılan yapay sinir ağı modeli	84
Şekil 5.3.	Bulanık mantık modülünün giriş ve çıkış üyelik fonksiyonları	86
Şekil 5.4.	Bulanık modül kural tabanı	86
Şekil 5.5.	Arıza teşhis şemasının deney düzeneği	87
Şekil 5.6.	Yapay sinir ağı için eğitim verisi örneği	88
Şekil 5.7.	Yapay sinir ağı için eğitim hatası grafikleri	88
Şekil 5.8.	Sağlam bir motorun akım spektrumu	89
Şekil 5.9.	Girişlere göre arızaların değişimi	90
Şekil 5.10.	Önerilen zaman serileri veri madenciliği arıza teşhis yönteminin blok diyagramı	92
Şekil 5.11.	Arıza teşhis yönteminin deney düzeneği.....	93
Şekil 5.12.	Sağlam motor için akımın eğitim ve test verisi.....	93
Şekil 5.13.	Arızalı motor için akımın eğitim ve test verisi.....	94
Şekil 5.14.	Motorun sağlam ve arızalı durumu için faz alanı grafikleri	94
Şekil 5.15.	Negatif seçim için kodlama.....	96
Şekil 5.16.	180 V ile beslenen bir kırk rotorlu motorun akım spektrumu	96
Şekil 5.17.	Yapay bağıklık tabanlı arıza teşhis şeması	97
Şekil 5.18.	Test ve repertuar arasındaki eşleşme denklemi	97
Şekil 5.19.	Örnek bir eşleştirme	98
Şekil 5.20.	Arıza teşhisi için kurulan deney düzeneği.....	99
Şekil 5.21.	Deney düzeneği şeması	99
Şekil 5.22.	Bir çubuğu kırk rotor.....	100
Şekil 5.23.	Veri toplama kartının analog girişine sinyalin bağlanması	100
Şekil 5.24.	Motordan akım verilerini almak için yazılan program.....	101

Şekil 5.25.	DAQAI Activex bileşeninin bazı özellikleri	102
Şekil 5.26.	180 V ile beslenen sağlam motorun akım ve spektrum grafikleri.....	103
Şekil 5.27.	180 V ile beslenen bir kırık rotorlu motorun akım ve spektrum grafikleri	103
Şekil 5.28.	Algoritmanın test sonuçları	104
Şekil 5.29.	220 V ile beslenen bir kırık rotorlu motorun akım spektrumu	105
Şekil 5.30.	Yapay bağışık modülünün eğitim aşaması	106
Şekil 5.31.	Bağışık sistem tabanlı bulanık arıza teşhis şeması	107
Şekil 5.32.	Bulanık modül üyelik fonksiyonları.....	107
Şekil 5.33.	220 V ile beslenen sağlam motorun akım ve spektrum grafikleri.....	108
Şekil 5.34.	220 V ile beslenen bir kırık rotorlu motorun akım ve spektrum grafikleri	109
Şekil 5.35.	Sağlam ve sürtünme arızası olan bir motora ait akım spektrum grafiği.....	109
Şekil 5.36.	Yapay bağışık sistemde izleme esnasında oluşan eşleşme sonuçları	110
Şekil 5.37.	İki girişe göre arıza türünün değişimi.....	110
Şekil 5.38.	Arıza şiddeti ve arıza türü grafiği.....	111

ÇİZELGELERİN LİSTESİ

Çizelge 2.1.	Arıza teşhisi için kullanılan yöntemler.....	25
Çizelge 5.1.	Tek fazlı sincap kafes bir asenkron motor için parametre değerleri	87
Çizelge 5.2.	Yapay sinir ağları test sonuçları	89
Çizelge 5.3.	Bulanık mantık simülasyon sonuçları	90
Çizelge 5.4.	Eğitim ve test verileri için yarıçap farkları.....	94
Çizelge 5.5.	Deneyde kullanılan asenkron motorun özellikleri.....	98
Çizelge 5.6.	Bulanık modül kural örnekleri	108
Çizelge 5.7.	Simülasyon sonuçları	110
Çizelge 5.8.	Simülasyon girişleri.....	111

SİMGELER

I	: Motor akımı
ω	: Motorun açısal hızı
N	: Sarım eşdeğer devir sayısı
B	: Sürtünme durumu
De	: Dinamik eksantriklik
N_{hb}	: Nominal hava boşluğu
Min_{hb}	: Minimum hava boşluğu
R_1	: Stator ile rotor arasındaki hava boşluğu mesafesi
R_2	: Gerçek minimum hava boşluğu mesafesi
η	: Öğrenme oranı
β	: Momentum katsayısı
μ	: Üyelik derecesi
T	: T bağışık hücreleri
B	: B bağışık hücreleri
TCR	: T hücre reseptörü
BCR	: B hücre reseptörü
A_n	: Antikor
A_t	: Antijen
V	: Şekil uzayı
V_ε	: Afinite eşiği
M	: Antikor ile antijen arasındaki öklid mesafesi
S	: Öz kümeler
B	: Korunmuş küme
A	: Detektör
σ	: Zaman serisinin standart sapması
l	: Bir zaman serisi parçasının uzunluğu
e	: Bir zaman serisi parçasının eğimi
p	: Zamana ait örüntü
P	: Zamana ait örüntü kümesi
Q	: Gömülme boyutu
τ	: Zaman gecikmesi
F	: Faz alanı matrisi
g	: Olay tanımlama fonksiyonu

f	: Hedef fonksiyonu
BP	: Bulanık zamana ait örüntü kümesi
C	: İki zaman serisi arasındaki otokorelasyon
K	: İki zaman serisi arasındaki karşılıklı bilgi
M_n	: Faz alanında bir noktanın en yakın komşusu
N_c	: Sarım arızası
F_c	: Sürtünme arızası
B_c	: Kırık rotor arızası
E_c	: Eksantriklik arızası
f_b	: Kırık rotor ile ilgili yan bant frekansı
f_h	: Hat frekansı
s	: Motor kayması
n_s	: Motorun senkron hızı
n_m	: Motorun mekaniksel hızı
RS	: Rotor çubuğu sayısı
CH	: Motorun çalışma hızı frekansı
B_U	: Kırık rotor üst yan bant frekansının genliği
BAE_F	: Kırık rotor alt yan bant ile Eksantriklik yan bant frekanslarının genlikleri farkı
Δi	: İlk fark filtrelemesi
m_l	: Faz alanında noktaların ağırlık merkezi
r	: Dönme yarıçapı

KISALTMALAR

ÖBTHO	: Öz Bağımlı Tümülek Hareketli Ortalama
HFD	: Hızlı Fourier DönüŖümü
DZE	: Dinamik Zaman Eğrilięi

ÖZET

Yüksek Lisans Tezi

ARIZA TEŞHİSİNDE VERİ MADENCİLİĞİ VE YUMUŞAK HESAPLAMA TEKNİKLERİNİN KULLANIMI

İlhan AYDIN

Fırat Üniversitesi

Fen Bilimleri Enstitüsü

Bilgisayar Mühendisliği Anabilim Dalı

2006, Sayfa: 121

Asenkron motorlar, endüstrideki elektromekaniksel enerji dönüşüm alanının büyük bir kısmında kullanılır. Bu tip motorların maliyetlerinin ucuzluğu, sağlamlık ve güvenilirlikleri birçok uygulamada tercih edilmelerini sağlamıştır. Fakat çalışma ortamları ve nem gibi etkiler bu motorların farklı bölümlerinde arızalar oluşmasına sebep olmaktadır. Arıza teşhisinin amacı oluşan bu arızaları erken bir aşamada belirlemektir.

Bu çalışmada asenkron motorların stator, rotor ve mil yatağı gibi bileşenlerinde oluşan arızalar yumuşak hesaplama ve veri madenciliği teknikleri ile teşhis edilmiştir. Bu teknikler arıza teşhisi için motor akımı ve rotor hızı gibi kolayca elde edilebilir sinyalleri kullanırlar. Dolayısıyla motorun içyapısını ve matematiksel dinamiklerini bilmeye gerek yoktur. Yapay sinir ağları, bulanık mantık ve yapay bağışık sistemler gibi yumuşak hesaplama teknikleri kullanılarak kırık rotor, sarım, mil yatağı sürtünmesi ve eksantriklik arızaları başarılı bir şekilde teşhis edilmiştir. Yapay bağışık sistemler ve bulanık mantık birleştirilerek iki farklı motor tipindeki arızalar belirlenmiştir. Zaman serileri veri madenciliği yöntemi ile sarım ve sürtünme arızalarını teşhis edecek özellikler motor hızı ve stator akımından çıkarılmıştır.

Arıza teşhisi için iki tip asenkron motor kullanılmıştır. Simülasyon verileri tek fazlı bir asenkron motordan elde edilmiştir. Deneysel veriler üç fazlı bir asenkron motordan veri toplama kartı aracılığıyla bilgisayara aktarılmıştır. Akım verilerini ölçmek için üç adet akım duyargası kullanılmıştır.

Anahtar Kelimeler: Arıza Teşhisi, Arıza Tespiti, Yumuşak Hesaplama Teknikleri, Veri Madenciliği, Zaman Serileri Veri Madenciliği, Asenkron Motorlar

ABSTRACT

Master Thesis

THE USING OF DATA MINING AND SOFT COMPUTING TECHNIQUES IN FAULT DIAGNOSIS

İlhan AYDIN

Firat University

Graduate School of Natural and Applied Sciences

Department of Computer Engineering

2006, Page: 121

Induction motors are used in a big part of the field of electromechanical energy conversion in industry. Reliability, robustness and cheap of the costs of these type motors has been provided the preferring of them in many applications. But effects such as operation environment and humidity cause faults in different parts of these motors. The goal of fault diagnosis is to detect these occurred faults at an early stage.

In this study, faults occurred in stator, rotor and bearing of induction motors have been diagnosed via soft computing and data mining techniques. These techniques use easily acquired signal such as rotor speed and motor's current for fault diagnosis. Consequently, mathematical dynamics and internal construction of motors are needn't known. Broken rotor bars, bearing friction, eccentricity and stator winding faults have been diagnosed via soft computing techniques such as artificial neural networks, fuzzy logic and artificial immune systems, successfully. Faults in two different type motors have been detected by combining fuzzy logic and artificial immune systems. Features to diagnose the stator winding and bearing friction faults have been extracted from stator current and rotor speed by time series data mining method.

Two type induction motors have been used for fault diagnosis. Simulation data have been obtained from single phase induction motor. Experimental data has been transferred the computer by using data acquisition card. Three current sensors have been used for measuring the current data.

Key Words: Fault Diagnosis, Fault Detection, Soft Computing Techniques, Data Mining, Time Series Data Mining, Induction Motors

1. GİRİŞ

Teknoloji alanındaki gelişmeler ile birlikte, geçmişte uzun zaman alan işlemler günümüzde artık çok kısa sürelerde yapılmaktadır. Bu gelişmelerin sonucu olarak büyük fabrikalardaki üretimler daha da hızlanmıştır. Bununla birlikte bu üretimi gerçekleştiren cihazların herhangi birinde oluşan bir problem, bütün sistemin performansını kötü yönde etkilemektedir. Bu üretimleri sürekli aynı performansta tutmak ve sistemin tamamen durmasını engellemek için fabrikada kullanılan araçların sürekli kontrol edilmesi gerekir. Uygun izleme araçları ile üretimde kullanılan araçların performansı sürekli ölçülmeli ve herhangi bir kural dışı çalışmanın sebebi veya sebepleri belirlenmelidir. Erken bir aşamada bu performans düşüklükleri belirlenip gerekli önlemler alınmazsa üretim tamamen durabilir. Endüstrideki büyük fabrikalarda bu tür problemler birçok sebebe bağlı olabilir. Fabrikalarda iş gücünün %80 gibi büyük bir kısmını asenkron motorlar yerine getirmektedir [1]. Dolayısıyla bu motorların sürekli izlenmesi ve arızalarının erken bir aşamada teşhis edilmesi önem kazanmaktadır.

Asenkron motorların kullanıldığı uygulamalar çok önemli olup motorların sürekli iyi bir performansta ve durumda tutulmaları gerekir. Küçük güçlü asenkron motorlar buzdolabı, çamaşır makinesi ve karıştırıcı gibi ev aletlerinde kullanılırken, büyük güçlü asenkron motorlar tekstil, kağıt ve petrokimyasal fabrikalarda kullanılırlar. Ayrıca askeri ve uzay alanlarındaki uygulamalar için bu motorlar kullanılmakta olup, güvenilirlik ve sağlamlık bu uygulamalar için önem taşır. Asenkron motorların korunması, bu iyi performansın sürekliliğinin sağlanması açısından çok önemlidir. Motor arızalarının belirlenmesi ve giderilmesi için bazı endüstriyel fabrikalar reaktif koruma ve arıza durumunda korumayı kullanmışlardır [2]. Reaktif koruma, arızanın ne kadar süre aynı performansta çalışacağını belirlemek zor olduğundan tercih edilen bir yöntem değildir. Bozulma durumunda koruma yönteminde ise arızalar oluşuktan sonra bu motorların incelenmesi ve arızaların giderilmesini sağlar. Koruyucu bakım yöntemi, arızalar oluşmadan önce onları teşhis etmek amacıyla donanım performansının sürekli izlenmesine dayanır. Diğer koruma şekli ise durum izlemesi olarak bilinir ve son yıllarda araştırmacılar tarafından ilgi duyulan bir alan olmuştur [3]. Günümüzde yüksek performanslı endüstriyel süreçler, motor performansının önemli derecede düşmesine sebep olan arızalara izin vermeyecek şekilde duyarlı bir yapıdadır. Bu duyarlılıktan dolayı elektrik motorlarında oluşacak arızaların tespit edilmesi ve düzeltilmesi önemli bir konudur.

Motorların başlangıç arızalarının tespiti, teşhisi ve izlenmesi mühendislik alanında zor ve önemli konulardan biridir. Uygun izleme ve arıza teşhis algoritmalarıyla birlikte, erken bir aşamada arıza teşhisi farklı mühendislik sistemlerinin çalışma güvenirliliğinin arttırılmasında

önemli rol oynamaktadır. Endüstride kullanılan donanımların güvenilirliğinin geliştirilmesi, kontrol amaçlı olarak çalışma esnasında performans izlenmesinden veya üretim esnasında kalite kontrol izlenmesi ile sağlanabilir. Çalışma veya üretim esnasında bu amaçlar için kullanılan teknikler veya metotlar arıza teşhisi veya tespiti olarak bilinir.

1.1. Asenkron Motorlarda Arıza Teşhisinin Önemi

Endüstride kullanılan motorlar, genellikle farklı çalışma ortamları ve durumlarına maruz kalmaktadırlar. Bu yüzden zamanla farklı harici ve dahili motor arızaları oluşmaktadır. Motor arıza teşhisi ve tespiti, bu arızalar oluşmadan önce performansı korumak için belirli bir toleransa izin vererek oluşabilecek arızaların önceden tespit edilmesi sürecidir [2]. Bu makineler genellikle sağlam ve güvenilir olmalarına rağmen, ani arızaların oluşması, mekanik enerjinin elektrik enerjisine dönüştürülmesi veya tersi durumlarda oluşan zorlanmalardan dolayıdır [3]. Ayrıca nem ve aşındırıcı ortamlar da motorun eskimesini ve bozulmasını hızlandırır. Bu çalışma ortamları düzeltilmezse performans düşüşleri sürekli devam eder ve motor sonunda tamamen bozulabilir. Bu yüzden motor performansını etkileyen bu belirtiler büyük arızaların habercisidir [4]. Büyük arızalar oluşmadan ve motorun performansı önemli bir derecede etkilenmeden önce oluşan bu performans düşmelerinin sebebi belirlenip düzeltilirse problemlerin çoğu giderilebilir. Uygun arıza teşhis teknikleri ve uygun durum izleme sistemleri geliştirilerek motorların koruma maliyetleri büyük oranda azaltılırken, bu makinelerin kullanılabilirliği önemli bir derecede artırılmış olur.

Günümüz endüstrisinde asenkron motorların kullanımı kapsamlıdır ve motorlar yanlış çalıştırma ve fabrika imalat hatalarına maruz kalabilirler. Dahili motor arızaları (örneğin, motor bağlama tellerinin kısa devre olması, sarımdan sarıma kısa devreler, topraklama arızaları, aşınmış veya kırık mil yatakları, kırık rotor çubukları) kadar harici motor arızaları da (örneğin faz hatası, ana destek asimetrisi, mekaniksel aşırı yüklenmeler ve tıkalı rotorlar) kaçınılmazdır. Eğer uygulama, ortam veya motor arızalarının etkisi ve sebebi anlaşılırsa çoğu arızalardan kaçınmak mümkün olur. Elektrik motorları için güvenlik gereksinimleri, teknolojiye ilerlemeler ve motor uygulamalarının öneminden dolayı sürekli artmaktadır.

Asenkron motorlar genellikle devre kesiciler, sigortalar veya röleler gibi elemanlar ile korunur. Fakat bu elemanlar büyük arızalardan önce haber veren erken uyarı sistemleri değildir. Sadece arıza durumunda elektriğin kesilmesi ve motorun yakınında çalışan kişilerin korunmasını sağlarlar. Ani arızalar, bozulmalar ve yanlış çalıştırmalar ile oluşan arıza durumlarında çalışan motor için koruma şemalarının geliştirilmesine ek olarak, kritik uygulamalarda asenkron motorların sürekli izlenmesi büyük arızaları minimize etmek ve

makine güvenilirliğini arttırmak için artan bir gereklilik olmuştur. Mikroişlemci tabanlı izleme sistemleri, olası arıza durumlarının tahmini ve makine değişkenlerinin düzenli analizi için kullanılabildiklerinden dolayı özel ilgi görmüşlerdir.

1.2. Asenkron Motorlarda Oluşan Temel Arızalar

Asenkron motorlarda oluşan arızalar motorun üç temel bileşeniyle ilişkilidir. Bunlar rotor, stator ve mil yatağıdır. Ayrıca bu arızaların bazıları elektriksel problemlerden dolayı oluşurken, bazıları mekaniksel ve manyetik devre problemlerine bağlıdır. Bir motorun elektriksel, manyetiksel ve mekaniksel devrelerinde oluşan arızaların teşhis edilmesi asenkron motorların en önemli problemlerinden biri olmuştur. Asenkron motorlarda oluşan arızalar 3 ana başlıkta toplanabilir [4]:

1. Sarım arızaları
 - Stator sargılarının kısa devre olması
 - Kırık rotor çubukları ve rotor sargılarının kısa devre olması
 - Rotor sonlandırıcı arızaları
2. Manyetik devre arızaları
 - Statik hava boşluğu eksantrikliği
 - Dinamik hava boşluğu eksantrikliği
3. Motorun mekaniksel sistem arızaları
 - Mil yatağı arızaları

1.3. Asenkron Motorlarda Oluşan Arızaların Teşhisi için Bazı Yaklaşımlar

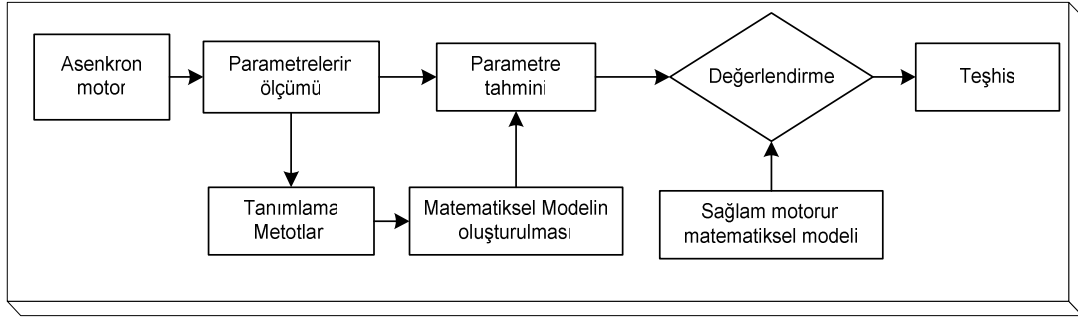
Asenkron motorların arıza teşhisi için kullanılan verilerin elde edilme maliyetleri, arıza teşhisinin uygunluğu açısından önemli rol oynar. Geliştirilen arıza teşhis şemalarının her biri belirli avantajlara ve dezavantajlara sahiptir. Bazı arıza teşhis teknikleri motorun durumunu belirlemek için çevrimdışı arıza analizi gerektirir. Bazı teknikler ise motor durumunu belirlemek için pahalı ölçüm tekniklerine ihtiyaç duyar. Örneğin; radyo frekans izleme tekniği, sarım izolasyonlarının arıza içerip içermediğini belirlemek için sinyal dalga biçimlerinin değişimini ölçer ve makinenin stator sarımına radyo frekans sinyallerini enjekte eder [5]. Bu teknik, pahalı donanımlar gerektirir ve sadece büyük ve pahalı makinelerin arızalarını belirlemek için kullanımı olanaklıdır. Bir diğer yöntem ise motorun iç yapısına zarar vermeden duyargalar aracılığıyla elde edilen verilerden arıza teşhisi yapmaktır. Bu amaçla motorun farklı özelliklerini ölçmede kullanılan farklı duyargalar geliştirilmiştir. Geliştirilen bu duyargalar titreşim, akım,

gerilim, akı, moment ve sıcaklık gibi parametreleri ölçmek için kullanılırlar [6]. Bu yöntemler motor yapısını bozmadan, motor durumunu tahmin etmek için kolayca erişilebilir ve pahalı olmayan ölçümlere dayanır.

Asenkron motorlarda arıza teşhisi için farklı yöntemler kullanılmıştır. Bu yöntemlerden bazıları herhangi bir ölçüm yapmadan motor dinamiklerinin matematiksel modeline göre arıza teşhisi yaparken, bazıları uzman bilgisine dayalı arıza teşhisi yapmıştır. Bunlar dışında maliyet açısından daha çok tercih edilen yöntemler, duyargalar aracılığıyla elde edilen bilgilerin arıza teşhisi için kullanılmasıdır. Aşağıda bu yöntemler tek tek incelenerek her birinin sunmuş olduğu avantajlar ve dezavantajlar detaylı olarak verilmiştir.

1.3.1. Model Tabanlı Teknikler

Arıza teşhisi için model tabanlı teknikler, ilgili sistemin matematiksel modellerine dayanırlar. Gerçek zamanlı fiziksel sistemlerde doğrusal olmayan davranışlarından dolayı bir makinenin matematiksel modeli elde edilmeden önce bazı varsayımlar yapılmalıdır [5]. Fakat matematiksel modeller eksik veya yetersiz olursa model yanlış sonuçlar verir [4]. Ayrıca model tabanlı teknikler gürültü ve titreşimlerden etkilenirler. Şekil 1.1 'de model tabanlı arıza teşhis şeması verilmiştir.



Şekil 1.1. Model tabanlı arıza teşhis şeması

Şekil 1.1'e göre motordan elde edilen parametreler kullanılarak sistemin matematiksel modeli oluşturulur. Elde edilen model, daha önce oluşturulan sağlam motorun matematiksel modeli ile karşılaştırılarak parametre değişimlerine göre arıza türü belirlenir. Arıza teşhisi bu farklar arasındaki eşik değerleri belirlenerek elde edilir.

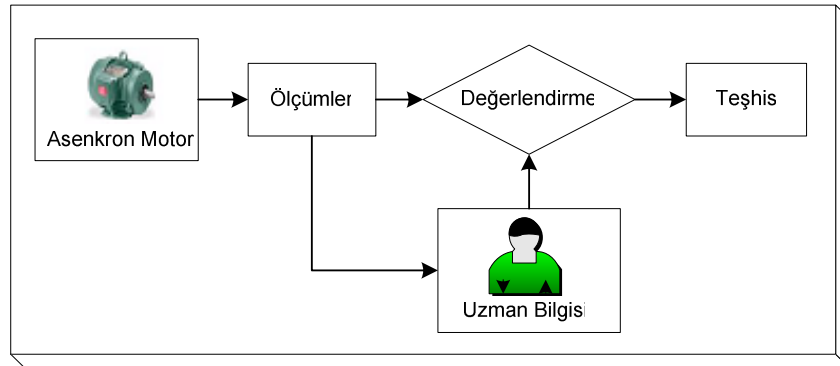
Parametre tahmin tekniği en popüler model tabanlı arıza teşhis yaklaşımıdır. Bu yöntemin doğru bir matematiksel modelin oluşturulmasını ve karmaşık sistem parametrelerinin detaylarının anlaşılmasını gerektirir. Bu parametreler motorun durumunu belirlemek için seçilir.

Örneğin, motorun mil yatağı durumu motorun mekaniksel denklemindeki sönüm katsayısını etkiler. Mil yatağı aşınırken, sönüm katsayısı artar. Bu yüzden parametre tahmin yaklaşımı sönüm katsayısının değerini tahmin etmek için motorun mekaniksel denklemini ve rotor hızı gibi ölçümleri kullanır. Seçilen parametrelerin nümerik değerleri tahmin edildikten sonra iyi veya kötü mil yatağı gibi niteliksel tanımlamalar yapabilmek için tahmin edilen motorun sönüm katsayısı değerlerini nümerik değerlere dönüştürmek gerekir.

Parametre tahmin tekniğinin bir dezavantajı, sistemin doğru bir matematiksel modelinin elde edilmesindeki zorluktur. Eğer sistemin matematiksel modeli yeterince doğru değil ise, farklar modelleme hataları içerir ve hataların etkisinden modelleme hatalarının etkisini ayırt etmek zor olur.

1.3.2. Uzman Bilgisine Dayalı Teknikler

Arıza teşhisi alanında uzmanlaşmış bir mühendis, motorun matematiksel modelini tam olarak bilmeden çalışma ortamına ve ölçümlere bakarak bir motordaki arızaların türü hakkında bilgi verebilir [2]. Bu yöntem model tabanlı yöntemlere göre daha avantajlıdır. Çünkü lineer olmayan modelleme teknikleri ortama ve gürültüye duyarlı olduklarından model için yapılan varsayımlarda bazen doğru sonuçlar vermeyebilir. Fakat arıza teşhisi için deneyimli bir mühendisin kullanılmasında da bazı zorluklar vardır. Bu zorluklardan biri deneyimli mühendisleri eğitmek maliyetli olup; gerçek zamanda böyle bir arıza teşhis uygulamasının yapılması zordur [4]. Ayrıca deneyimli mühendislerden bilginin tam olarak alınması da bazı durumlarda mümkün olmayabilir. Gerçekte motor arıza teşhisi için deneyimli mühendislerin kullandığı bilgi bu alanda daha önce motor arızaları ile ilgili karşılaştıkları problemlerden kazandıkları tecrübelerden edinilmiştir. Şekil 1.2’de uzman bilgisine dayalı arıza teşhis şeması verilmiştir.



Şekil 1.2. Uzman bilgisine dayalı arıza teşhis şeması

Şekil 1.2'den de görüldüğü gibi arıza teşhisi konusunda uzman olan kişi asenkron motordan gelen ölçümlere bakarak motorda ne tür arızaların oluştuğu hakkında bilgi vermektedir. Uzman kişinin aldığı ölçümlerden arıza teşhisi yapabilmesi için daha önce bu arızaların oluşturduğu etkiler üzerinde bir deneyime sahip olması gerekir.

1.4. Bilgisayar Destekli Arıza Teşhis Algoritmalarının Gerekliliği

Model tabanlı ve uzman bilgisine dayalı tekniklerin uygulanmasındaki zorluklardan dolayı bilgisayar destekli algoritmalara talep artmıştır [4]. Arıza teşhisi için daha etkili ve hızlı bilgisayar algoritmalarının geliştirilmesi ile insana ve modele bağımlılık gittikçe azaltılmış olunur. İşaret işleme teknikleri ve veri analizine dayalı erken arıza teşhisi ve tespiti endüstriyel fabrikalarda önemli olmaktadır. Bilgisayar destekli arıza teşhis algoritmaları kullanıldığında uzman bilgisi olmadan arızalar belirlenip teşhis edilebilir [7]. Bu tür algoritmalarda kullanılan veriler genellikle akım, gerilim, rotor hızı ve moment gibi sinyaller olduğundan motorun matematiksel modeli hakkında çok fazla bilgiye ihtiyaç yoktur. Bu algoritmaların kullanılması maliyet açısından da önemli avantajlar getirmektedir.

1.5. Asenkron Motor Arızalarının Teşhisi için Yumuşak Hesaplama Tekniklerinin Kullanılması

Asenkron motorlarda oluşan arızaların minimum maliyet ile çevrimiçi teşhis edilebilmesi önemlidir. Arıza teşhisinin en büyük amaçlarından biri, doğru ve güvenilir sonuçlar elde etmek için efektif olarak bilgiyi analiz etmek, ilgili bilgiyi toplamak ve uygun teknolojiyi kullanmak için yetenekler geliştirmektir. Akıllı arıza teşhis yöntemleri, bu görevleri başarılı bir şekilde yerine getirebilirler. Akıllı arıza teşhis yöntemlerinin çoğu insanın öğrenme kabiliyeti, bağışıklık sistemi ve muhakeme özelliklerine göre geliştirilmiştir. Bu yöntemlerin oluşturduğu genel çatı yumuşak hesaplama olarak bilinir.

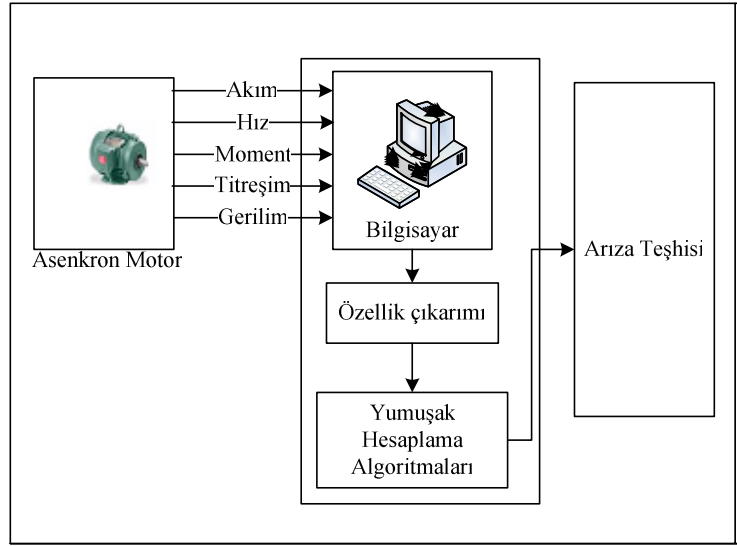
Yumuşak hesaplamanın alt dallarından biri olan yapay sinir ağlarının uygulanması ve geliştirilmesi oldukça kolaydır. Yapay sinir ağları, motor hakkındaki sinyaller ölçülerek elde edilebildiğinde kullanılabilir. Ölçümler elde edildikten sonra, yapay sinir ağı eğitim algoritmaları ile sistem öğrenilir. Böyle bir yöntemin temel avantajı, çok karmaşık matematiksel modeller üretilmeden bir arızanın boyutu ve türü hakkında çevrimiçi bilgi elde edilebilmesidir. Sinir ağı, örneklenen motor verilerinden toplanan giriş çıkış örneklerine dayalı olarak sistemi

öğrenir. Sinir ağı sinyalleri simülasyon veya deneysel çalışmalardan elde edilen veriyi kullanarak tasarlanabilir.

Bulanık mantık, insanın düşünme tarzını model alarak gerçek dünya problemleri için tam ve keskin olmayan çözümler sunar [8]. Bu yöntem insan tarafından tercih edilen heuristik, dilsel ve niteliksel bilgiyi, bilgisayar tarafından tercih edilen sayısal ve nicel bilgiye kolayca ve sistematik olarak dönüştürebilir. Dilsel değişkenler ve kural tabanı, bulanık mantığın temel bileşenlerini oluşturur. Bu bileşenler kullanılarak karar verme ve kontrol gibi endüstriyel uygulamalar kolayca gerçekleştirilebilir. Arıza teşhis prensiplerini sezgisel olarak uygulamak, sonuçları analiz etmek ve yorumlamak için bulanık mantık basit bir metot sağlar. Motorlarda arıza teşhisi için kullanılan bilgiler genellikle keskin değerler olmadığından, araştırmacılar bu alandaki birçok uygulamada bulanık mantık kullanmışlardır.

Bağışıklık sistemi, insan vücudunu hastalıklara karşı koruyan etkili bir savunma mekanizmasıdır. Bu sistem, vücudumuza giren bakteri, virüs ve kanser hücreleri gibi patojen olarak adlandırılan yabancı hücreleri tanır ve bunları yok etmek için belirli hücreler ile bir koruma sistemi oluşturur. Bağışık sistem, sahip olduğu hafıza aracılığıyla daha önce rastladığı yabancı hücreleri de tanır. Yapay bağışık sistemler, insanın bağışık sisteminin insan vücudunu koruma mekanizmasını ve yabancı hücreleri tanıma yeteneklerinden esinlenilerek geliştirilmiş bir yumuşak hesaplama yöntemidir [9]. Yapay bağışık sistemler bilgisayar güvenliği, optimizasyon ve öğrenme problemlerinin çözülmesinde yeni bir yaklaşım getirmiştir. Asenkron motorlarda oluşan arızalar da insan vücuduna giren patojenlere benzetilebilir. Sağlam bir motorun çalışma durumlarını öğrenen yapay bağışık sistem, herhangi bir arıza durumunda onu teşhis eden erken bir uyarı sistemi olarak düşünülebilir.

Arıza teşhisi için kullanılan yumuşak hesaplama teknikleri, farklı duyardardan gelen sinyalleri kullanmışlardır. Arıza teşhisi için direkt olarak alınan sinyallerin kullanılması pek uygun değildir. Şekil 1.3'te arıza teşhisi için yumuşak hesaplama tekniklerinin kullanımı şematik olarak verilmiştir.



Şekil 1.3. Arıza teşhisi için yumuşak hesaplama algoritmalarının kullanılması

Şekil 1.3'te görüldüğü gibi bir asenkron motordan alınan hız, moment, akım, titreşim ve gerilim sinyalleri bilgisayara aktarıldıktan sonra ilk olarak bu bilgilerden farklı arıza tipleri ile ilgili bilgiler çıkarılır. Daha sonra bu özellik sinyalleri üzerinde yumuşak hesaplama algoritmaları çalıştırılarak arıza tipi belirlenir. Son zamanlarda bu alanda yapılan çalışmalara ilgi oldukça artmıştır. Bu bölümün geri kalan kısmında yapılan çalışmalar ile ilgili literatür özeti verilmiştir.

Bu kategoriye giren çalışmalardan en önemlileri Chow tarafından yapılmıştır [2]. Chow, çalışmasında tek fazlı bir sincap kafes asenkron motorda oluşan sarım ve sürtünme arızalarının teşhisini yapmıştır. Yapılan çalışmada motor sarımlarının eşdeğer devir sayısına göre motor sarım durumu kötü, orta ve iyi gibi üç farklı kategoride sınıflandırılır. Aynı sınıflandırma işlemi, mil yatağı sürtünme arızalarını tespit etmek amacıyla da kullanılır. Motor akımı ve rotorun açısal hızının motorun sarım ve sürtünme durumuyla arasında bir bağıntı vardır. Motordan farklı durumlarda stator akımı ve rotor hızı değerlerini elde etmek için MOTORSIM olarak isimlendirilen bir simülasyon programı geliştirilmiştir. Elde edilen stator akımının efektif değeri ve ortalama rotor hızı (I, ω) yapay sinir ağının girişlerini oluşturur. Yapay sinir ağının çıkışları ise sarım ve sürtünme durumlarıdır (N, B). Yapay sinir ağının giriş ve çıkışları, $[0,1]$ aralığında normalleştirilir. Yapay sinir ağının çıkışlarından N stator sarımlarının eşdeğer devir sayısını, B ise farklı sürtünme durumlarını göstermektedir. Yapay sinir ağı başarılı bir şekilde eğitildikten sonra, I ve ω 'nın ölçülen değerlerinden sarım ve sürtünme durumları tanımlanabilir.

Filipetti ve diğ. [7] çalışmalarında, üç fazlı asenkron motorlardaki arızaları tespit etmek amacıyla yapay zeka tekniklerini kullanmışlardır. Burada stator faz akımları yapay zeka

araçlarına giriş olarak alınır. Arıza teşhis prosedürünün temel adımları; imza çıkarımı, arıza tanıma ve arıza şiddetinin değerlendirilmesinden oluşur. Asenkron motorun arıza imzaları, stator akımlarının harmonik bileşenleri ile bağlantılı olup her bir arıza belirli harmonik bileşenlerin varlığı ile ilişkilidir. Örneğin, kırık rotor çubuğu ile ilgili yan bant hesaplama formülü denklem 1.1’de verilmiştir.

$$f_{rbb} = f[k(1-s)/p \pm s] \quad k=1,5,7,..... \quad (1.1)$$

Burada s değeri kayma, f hat frekansı ve f_{rbb} kırık rotor ile ilgili frekansları temsil eder. Kırık rotor çubuğu arızaları incelenirken bu frekanslara bakılır. Veri elde etmek için bir veri toplama kartı ve dönüşüm için HFD (Hızlı Fourier Dönüşümü) kullanılır. Elde edilen spektrum bileşenlerinden arıza teşhisinde kullanılmayan bileşenler elenir. Yapay sinir ağları stator kısa devre arızalarının teşhisi için kullanılmıştır. Burada girişler kayma (s), negatif ve pozitif ardışık stator akımları giriş olarak kullanılıp çıkış olarak arıza yüzdesi hesaplanır. Kırık rotor çubuğu arızalarını bulmak için, bulanık mantık kullanılmış ve giriş olarak I_1 ve I_2 ’nin $(1 \pm 2s) f$ frekanslarındaki arıza bileşenleri alınmıştır. Mamdani tipi bir bulanık mantık sistemi tasarlanmıştır. Yine kırık rotor çubuğu arızalarını bulmak için, I_1 ve I_2 değerleri bir yapay sinir ağı tabanlı bulanık çıkarım sistemine giriş olarak verilip uygulamalar yapılmıştır.

Asenkron motorlarda arıza teşhisi için yapılan diğer bir çalışmada motordaki stator, rotor ve dönen mil yatağı arızaları teşhis edilmiştir [4]. Teşhis amacıyla ileri beslemeli yapay sinir ağları ile öz düzenlemeli kohonen ağları kullanılmıştır. Model tabanlı ve uzman bilgisine dayalı teşhis yöntemlerinin dezavantajları incelenmiştir. Akım spektrumundaki harmonik bileşenler, kırık rotor çubuğu arızalarını belirlemede kullanılabileceği için yapay sinir ağının girişlerini bu kayma frekans bileşenleri oluşturur. Bu amaçla iki yapay sinir ağı kullanılmıştır. Birinci yapay sinir ağında, sadece kırık rotor çubuklarının sayısının teşhis edilmesi amaçlanmıştır. Diğer sinir ağı yapısında ise hem kırık rotor çubuklarının sayısı hem de arıza tipi yapay sinir ağının çıkışlarını oluşturmaktadır. Dönen mil yatağı arızaları için hem akım spektrumu hem de titreşim spektrumu incelenmiştir. Her ikisi aynı anda kullanılarak da arıza teşhisi yapılmıştır. Stator arızaları için arama bobininde elektrik akımını oluşturan efektif gerilimin değeri ve bunun 50 Hz harmonik büyüklükleri alınır. Kohonen ağı ise çok katmanlı algılayıcı bir sinir ağı uygulanmadan önce ön aşamada sınıflandırma amacıyla kullanılır.

Awadallah ve diğ. [10] yaptıkları çalışmada yapay sinir ağları, bulanık mantık, uzman sistemler ve genetik algoritmalar gibi yapay zeka tekniklerinin elektrik makinelerin arıza teşhisinde nasıl kullanıldığı ile ilgili teknikler önermişlerdir. Motor hat akımının izlenmesi ile sağlam ve arızalı durumlarda frekans alanı karakteristiklerinin çıkarılması, algoritmalarının ilk

adımını oluşturmuştur. Arıza teşhisinde özellikle stator açık faz ve gerilim dengesizliğinde kullanılan park vektör yaklaşımı tanımlanmıştır. Stator sargı arızaları, kırık rotor, sonlandırıcı halka arızaları, mil yatağı arızaları ve eksantriklik ile ilgili arızaları içeren frekans spektral denklemleri verilmiştir. Son aşamada yapay sinir ağları, bulanık mantık ve uzman sistemlerin bu sinyallere nasıl uygulandığından bahsedilmiştir.

Bir asenkron motordaki kırık rotor arızalarını tespit etmek amacıyla Ayhan ve diğ. [11], üç fazlı bir asenkron motorun A fazından alınan akım değerlerini kullanmıştır. Yaptıkları çalışmada, farklı yük durumları göz önüne alınarak ilgili fazın güç spektral yoğunluğunu hesaplamışlardır. Kırık rotor çubuğu arızalarının tespiti için farklı spektrum analiz teknikleri karşılaştırılmıştır. Bu amaçla HFD, periodogram ve welch's periodogram gibi teknikler incelenmiştir. Bu tekniklerin performans açısından karşılaştırılması amacıyla t ve P hipotez testi kullanılmıştır. İstatistiksel sonuçlar, Welch's Perriodogram akım imza analiz metodunun arıza tespiti için daha iyi sonuçlar verdiğini göstermiştir.

Haji ve diğ. [12], kırık rotor arızalarının tespiti için çalışmalarında örüntü tanıma teknikleri önermişlerdir. Arızaları teşhis etmek için motor akım sinyalleri park vektörüne dönüştürülür. Park vektöründen elde edilen sinyaller akım spektrumuna dönüştürülür ve hız tahmini için motorun rotor sayısı ve kutup sayısı gibi bilgiler kullanılır. Bu özellik sinyalleri Bayesian sınıflandırıcıya verilerek kırık rotor arızaları tespit edilir.

Benbouzid ve diğ. [13], yaptıkları çalışmada üç fazlı bir asenkron motorda oluşan stator arızalarını teşhis etmek amacıyla bulanık mantık kullanmışlardır. Bu yöntem ile bir asenkron motorun statorunda oluşan açık faz ve gerilim dengesizliği gibi arızaların belirlenebileceğini göstermişlerdir. Bulanık mantığın girişi olarak üç faz akımının genliği alınır ve çıkış olarak sağlam bir motor için iyi, gerilim dengesizliği arızası olan bir motor için kötü ve açık faz arızası için çok kötü dilsel ifadeleri kullanılarak arızalar teşhis edilir.

Yapay sinir ağlarının lineer olmayan sistemleri öğrenebilme özelliği kırık rotor, mil yatağı, sarım, eksantriklik ve sürtünme arızalarının teşhisi için kullanılmıştır [14, 15]. Sarım ve mil yatağı sürtünme arızalarının teşhisi için, üç katmanlı ileri beslemeli bir yapay sinir ağı önerilmiştir. Sistemin girişleri için rotor hızı ve stator akımı kullanılır. Çıkış olarak mil yatağı ve sarımların durumu kötü, orta ve normal olarak derecelendirilir. Eksantriklik, kırık rotor ve mil yatağı arızalarını teşhis edebilen yapay sinir ağı, motordan alınan verileri lineer olmayan bir sistem olarak göz önüne alır.

Kolla ve diğ. [16], yaptıkları çalışmada tıkalı rotor, topraklama, dengesiz besleme gerilimi, aşırı yüklenme, yüksek ve düşük gerilim arızalarını teşhis eden ileri beslemeli bir yapay sinir ağı yapısı önermişlerdir. Bu arızaları teşhis etmek amacıyla, ileri beslemeli yapay sinir ağının girişlerine üç faz gerilimi ve akımı verilir. Yapay sinir ağı geriye yayılım öğrenme

algoritması ile eğitilir. Test için sağlam ve farklı arıza türü ile ilgili veriler yapay sinir ağına verilerek bahsedilen arızalar teşhis edilmiştir.

Açık faz ve gerilim dengesizliği arızalarını farklı yük durumları altında belirleyecek bir çalışma, Zidani ve diğ. [17] tarafından yapılmıştır. Üç fazlı bir asenkron motorun faz akımları alınarak, sağlam durum ve her bir arıza durumu için park vektörleri oluşturulur. Sağlam bir motorun park vektörüne göre arızalı bir durumun park vektöründeki fark ve bu farkın bir önceki adıma göre değişimi bulanık mantığın girişlerini oluşturur. Çıkış olarak, ilgili stator arızaları düşük, orta ve yüksek üyelik fonksiyonları olarak önem derecelerine göre sınıflandırılır.

Altug ve diğ. [18], tek fazlı bir asenkron motorda oluşan mil yatağı arızaları için iki farklı sinirsel bulanık modülü karşılaştırmıştır. Akım, hız ve moment bilgilerini sisteme giriş olarak alıp mil yatağı sürtünme arızaları teşhis edilmiştir.

Asenkron motor arızaları, akıllı hesaplama teknikleri ile teşhis edilmeden önce faz akımları üzerinde işaret işleme teknikleri kullanılarak özellik çıkarma algoritmaları [19, 20, 21, 22, 23, 24] çalışmalarında gerçekleştirilmiştir. Burada yapılan çalışmalar ile her bir arıza türüne özgü bazı özellikler çıkarılmıştır. Bu özelliklerin çoğu, motor akım imza analizi olarak bilinen ve herhangi bir faz akımı üzerinden HFD alınarak hesaplanan yöntemlere dayalıdır. Kırk rotor arızalarının teşhisi için Ayhan ve diğ. [24], akım bilgisi yanında hız bilgisini de kullanmışlardır. Yaptıkları çalışmada, dört farklı yük durumu için bir kırk rotor arızasını teşhis eden işaret işleme tabanlı bir yöntem geliştirilmişlerdir.

Yapay bağışık sistemlerin vücudu patojenlere karşı koruma ve yabancı hücreleri tanıma özelliklerinin benzetimi arıza teşhisine uyarlanabilir. Branco ve diğ. [25] kırk rotor, salınımlı yük ve dengesiz beslenme, salınımlı yük ve kırk rotor arızalarını teşhis etmişlerdir. Önerilen yaklaşımda stator akımı örneklenerek motor akım imza analizi yardımıyla akım spektrumuna dönüştürülür. Burada elde edilen özellik sinyalleri yapay bağışık sistemin girişini oluşturur. Sistemin sağlam ve arızalı durumları N değişken ile ifade edilir. Bunun için N -boyutlu bir durum vektörü bu değişkenleri tutmak için kullanılır. Yapay bağışık sistem üç modülden oluşmaktadır. Bu modüllerden, T-modülü arızalı motor verilerinden sağlam motor verilerini ayırt ederken, B modülü sıklıkla oluşan durumları T modülüne bildirir. D modülü ise durum uzayını kısaltmak ve algoritmanın daha hızlı çalışmasını sağlamak için kullanılır. Arızalı ve sağlam durumlar, şekil uzayında detektörler ile gösterilir. Önerilen algoritmaya göre arızalı veya sağlam bir durum, belirlenen bir sayıda tekrar ederse uyarım ve mutasyon süreçleri devreye girerek burada bir detektör oluşturur. Bu detektörler hangi arıza türünün oluştuğunu belirlemek için kullanılmıştır.

Gao ve diğ. [26], bir asenkron motorun mil yatağı arızalarını belirlemek amacıyla negatif seçim algoritması önermişlerdir. Negatif seçim algoritmasında detektör üretimi için ileri

beslemeli bir yapay sinir ağı kullanılmıştır. Titreşim duyargasından alınan veriler kullanılarak arızalar tespit edilir. Önerilen algoritmada arızanın türü belirlenmeyip sadece sağlam bir durumdan farklı çalışma durumları belirlenir.

Bir asenkron motorda oluşan rotor arızalarının belirlenmesi için Shulin ve diğ. [27], negatif seçim algoritmasına dayalı bir algoritma sunmuşlardır. Arıza teşhisi için motordan alınan titreşim verileri kullanılmıştır. Negatif seçim algoritması çevrimdışı eğitilerek arıza tiplerine göre detektörler üretilir. Bu detektörler daha sonra çevrimiçi kullanılarak arızalar tespit edilir.

Motor akımı üzerinden rotor arızalarının belirlenmesi için Martins ve diğ. [28], negatif seçim algoritması önermişlerdir. Önerilen algoritma detektör üretimi ve izleme olmak üzere iki aşamadan oluşmaktadır. Detektör üretim aşamasında elde edilen detektörler ile izleme aşamasında gelen örüntüler arasındaki eşleşme mesafesine göre arızalar teşhis edilmiştir.

1.6. Arıza Teşhisinde Zaman Serileri Veri Madenciliği Yöntemi

Bir zaman serisi, herhangi bir sistemden düzenli zaman aralıklarında kaydedilen ve her bir veriye kaydedildiği zaman değeri ile ulaşılabilen veri topluluklarıdır. Denklem 1.2’de bir zaman serisi örneği verilmiştir [29].

$$X=\{x_t, t=1,...,N\} \quad (1.2)$$

Burada t zaman indeksi ve N gözlemlerin sayısıdır. Zaman serileri analizi ekonomi, bilim ve mühendislikte çok sık karşımıza çıkmaktadır. En çok kullanıldığı uygulamalar arasında stok market fiyatlarının iniş-çıkışlarının tahmini, deprem tahmini, bir elektrik motorunda oluşan arızaların tespit edilmesi verilebilir. Bu yöntemin getirmiş olduğu dezavantajlardan biri, bir zaman serisinin analiz edilebilmesi için o zaman serisinin durağan hale getirilmesinin gerekliliğidir. Bir zaman serisindeki geçmiş verilere göre üretilen Öz Bağımlımlı Tümlleşik Hareketli Ortalama (ÖBTHO), zaman serileri modelleri zaman serisinin gelecekte alacağı değerleri tahmin edebilir. Fakat bu tür modellerin kullanılması, zaman serisinin zamanla değişmemesini yani durgun olmasını gerektirir. ÖBTHO modelleri, zaman serisini üreten sistemin lineer olduğunu ve diferansiyel denklemler veya fark denklemleri ile tanımlanabileceğini varsayar. Lineer olmayan gelişmelere paralel olarak veri analizi alanında birçok çalışma yapılmıştır. Bu çalışmalarda kullanılan veriler genelde bir deneysel çalışmadan elde edilir ve lineer denklemler ile ifade edilemezler. Bu tür zaman serilerine örnek olarak

fiziksel veya biyolojik bir deneyden zaman üzerinde alınan veriler, kimyasal bir reaksiyondaki yoğunluk ve internetteki ani trafik miktarı verilebilir [30].

Veri madenciliği, büyük miktardaki veri yığınlarından istatistik, makine öğrenmesi, yapay zeka ve örüntü tanıma yöntemlerini kullanarak faydalı ve kullanışlı bilginin elde edilmesi süreci olarak tanımlanmaktadır [31]. Veri madenciliği, bilimsel keşif sürecini otomatikleştiren metotların bir kümesini oluşturur. Veri madenciliğinin en büyük amacı, karmaşık ve büyük miktardaki verilerde gizlenmiş olan kullanışlı bilgiyi çıkarmaktır.

Zaman serileri veri madenciliği yöntemi, zaman serileri analizi ile veri madenciliği alanlarının birleştirilmesidir. Bu yöntem ile karmaşık, periyodik olmayan, düzensiz ve kaotik zaman serileri tahmin edebilir ve başarılı bir şekilde karakterize edebilir [29]. Zaman serilerini analiz etmek için veri madenciliği yöntemleri uyarlanarak geleneksel zaman serileri analizindeki durağanlık ve lineer gereksinimler gibi sınırlamalar ortadan kaldırılır. Zaman serileri veri madenciliği yöntemleri, durgun ve doğrusal olmayan zaman serilerinin analizini veri madenciliği yöntemlerini kullanarak yaptığından karmaşık matematiksel denklemlerin getirmiş olduğu yükü ve dezavantajları giderebilir.

Zaman serileri veri madenciliği yöntemi kullanılarak bir asenkron motorda oluşan arızalar tespit edilebilir. Zaman serisi olarak bir motorun stator akımı, rotor hızı, moment gibi sinyaller alınabilir. Bu sinyaller, doğrusal olmayan zaman serileri analiz yöntemleri kullanılarak belirlenen bir gömülme boyutu ve zaman gecikmesine göre faz alanına dönüştürülür. Faz alanında elde edilen grafikler veri madenciliği yöntemleri kullanılarak belirli sınıflar oluşturulur. Eğitim işleminden sonra test veri kümeleri oluşturulan eğitim modeline verilerek farklı arıza durumları ve sağlam durum ayırt edilir.

Bangura ve diğ. [32], bu alanda ilk çalışmalardan birini yapmışlardır. Geliştirmiş oldukları zaman serileri veri madenciliği yöntemi ile üç fazlı bir asenkron motorda oluşan kırık rotor, statik ve dinamik hava boşluğu eksantrikliği ve kırık sonlandırıcı arızalarını tespit etmişlerdir. Bu çalışmada sonlu elemanlar yöntemi kullanılarak her bir arızalı ve sağlam durum için asenkron motorun faz akımı ve moment sinyalleri elde edilmiştir. Elde edilen bu sinyaller, belirlenen zaman gecikmesi ve gömülme boyutuna göre bir faz alanına dönüştürülür. Faz alanında, akım sinyallerinden arızaları belirlemek zor olduğundan faz alanındaki moment grafiği incelenmiştir. Her bir moment sinyalinden elde edilen dönme yarıçapı bu arızaları teşhis etmek için önemli bir parametredir. Eğitim ve test verileri için elde edilen bu yarıçap değerleri karşılaştırılarak arızalı durumlar sınıflandırılmıştır.

Asenkron motorlarda zaman serileri veri madenciliğinin kullanılması ile ilgili diğer bir çalışma, Yeh ve diğ. [33] tarafından yapılmıştır. Yapılan çalışmada üç fazlı bir asenkron motorda oluşan stator kısa devre arızaları teşhis edilmiştir. Zaman serileri veri madenciliği

yöntemi ile statorun a, b ve c fazından alınan akım değerleri uzay vektörüne dönüştürülür. Elde edilen bu uzay vektörü için zaman gecikmesi ve gömülme boyutu belirlendikten sonra faz alanı oluşturulur. Faz alanında elde edilen grafiklerin Gaussian modeli oluşturularak, Bayesian sınıflandırma yöntemine göre arızalar sınıflandırılır.

Bu tez çalışmasında; asenkron motorlarda oluşan kırık rotor, eksantriklik, sarım ve sürtünme gibi arızaların tespit edilmesi amacıyla zaman serileri veri madenciliği ve yumuşak hesaplama tekniklerinin kullanılması amaçlanmıştır. Daha önce bu amaç için kullanılan parametre tahmini ve uzman kişiye dayalı arıza teşhis yöntemleri, karmaşık matematiksel modeller gerektirmesi ve pahalı olması gibi dezavantajlardan dolayı tercih edilmezler. Ayrıca matematiksel model motorun matematiksel dinamiklerinin bilinmesini gerektirmesine karşın, yumuşak hesaplama ve zaman serileri veri madenciliği yöntemleri sadece giriş ve çıkış durumlarının bilinmesi ile arıza teşhisi yapabilirler. Böylece arıza teşhisi yapabilmek için bir asenkron motor ile ilgili detaylı bilgi bilmeye ihtiyaç yoktur. Yumuşak hesaplama teknikleri kullanılarak gerçekleştirilen arıza teşhis algoritmalarının çoğu sadece tek tip motor için çalışmaktadır. Ayrıca akım verisi kullanarak arıza teşhisi yapan yöntemlerin de bazı eksiklikleri vardır. Önerilen tekniklerin birçoğu akım verisi yanında hız, gerilim veya daha maliyetli ölçümler gerektiren titreşim bilgisini kullanmaktadırlar. Bu çalışmada, daha önce zaman serileri veri madenciliği yöntemi ile moment sinyalleri kullanılarak yapılan arıza teşhisinin getirmiş olduğu maliyet'ten daha ucuz olan akım ve hız gibi veriler kullanılmıştır. Ayrıca farklı motor tipleri için çalışan arıza teşhis algoritmaları geliştirilmiştir. Geliştirilen yapay bağımsızlık algoritmaları, daha öncekilerden farklı olarak kural dışı çalışmaları tespit etmeyip arızanın türünü de belirlemektedir. Arıza teşhisi için yapay bağımsız sistemler ve bulanık mantık birleştirilerek daha iyi sonuçlar elde edilmiştir.

1.7. Tezin Amacı

Bu tezin amacı; asenkron motorların mil yatağı, stator ve rotor gibi temel bileşenlerinde oluşan arızaları teşhis etmek için yumuşak hesaplama ve veri madenciliğine dayalı algoritmalar geliştirmektir. Bu algoritmalar motorun matematiksel dinamiklerine bağlı kalınmadan rotor hızı ve stator akımı gibi sinyalleri kullanırlar. Motor sinyalleri, bir veri toplama kartı aracılığıyla bilgisayara aktarılır. Bu sinyaller üzerinde uygun işaret işleme teknikleri ile her bir arıza türü için ayırt edici olan özellikler çıkarılır. Bu özellikler, veri madenciliği ve yumuşak hesaplama algoritmalarına giriş olarak alınıp motor bileşenlerindeki arızalar teşhis edilir. Buradaki arızalar motorun tamamen bozulması ile oluşan arızalar olmayıp, performans düşüklüklerine yol açan

başlama aşamasındaki arızalardır. Dolayısıyla geliştirilen algoritmalar erken teşhis veya uyarım algoritmaları olarak da düşünülebilir.

1.8. Tezin Yapısı

Bu tez altı bölümden oluşmaktadır. Bölüm 2’de asenkron motorların yapısı, çalışma prensipleri, bu motorlarda oluşan arıza türleri ve motorda oluşturdukları etkiler detaylı olarak verilmektedir.

Bölüm 3’te yumuşak hesaplama tekniklerinden kısaca bahsedilmiş ve yapay sinir ağları, bulanık mantık ve yapay bağışıklık gibi yumuşak hesaplama teknikleri ayrıntılı olarak ele alınmıştır.

Bölüm 4’te veri madenciliği teknikleri ve veri madenciliği modelleri genel özellikleri ile verilir. Zaman serileri veri madenciliği kavramı, zaman serilerinde benzerlik araması, benzerlik aramasının performansını geliştirmek için önerilen algoritmalar, indeksleme ve boyut azaltma gibi teknikler anlatılmıştır. Lineer olmayan zaman serileri analizi yöntemi ve bu yöntemeye dayalı zaman serileri veri madenciliği teknikleri bölümün devamında verilmiştir.

Bölüm 5’te iki farklı motor tipinden simülasyon ve deneysel olarak alınan motor verilerini kullanan yumuşak hesaplama ve veri madenciliğine dayalı dört farklı uygulama çalışması sonuçları ile birlikte sunulmuştur. Simülasyon verileri, tek fazlı sincap kafes bir asenkron motordan alınarak sarım, sürtünme, eksantriklik ve kırık rotor arızaları teşhis edilmiştir. Deneysel verileri almak için bir bilgisayar, veri toplama kartı, üç adet akım duyargası ve dirençler kullanılır. Deneysel olarak alınan veriler için gerçekleştirilen algoritmalar ile kırık rotor arızalarını teşhis edebilmektedir. Ayrıca hem simülasyon hem de deneysel verilerin aynı anda kullanıldığı ve arızaların çevrimiçi olarak belirlendiği bir yöntem de sunulmuştur.

Bölüm 6’da çalışma ile ilgili genel sonuçlar ve çalışmanın literatürdeki yeri verilmiştir.

2. ASENKRON MOTORLARIN YAPISI VE BU MOTORLARDA OLUŞAN ARIZA TURLERİ

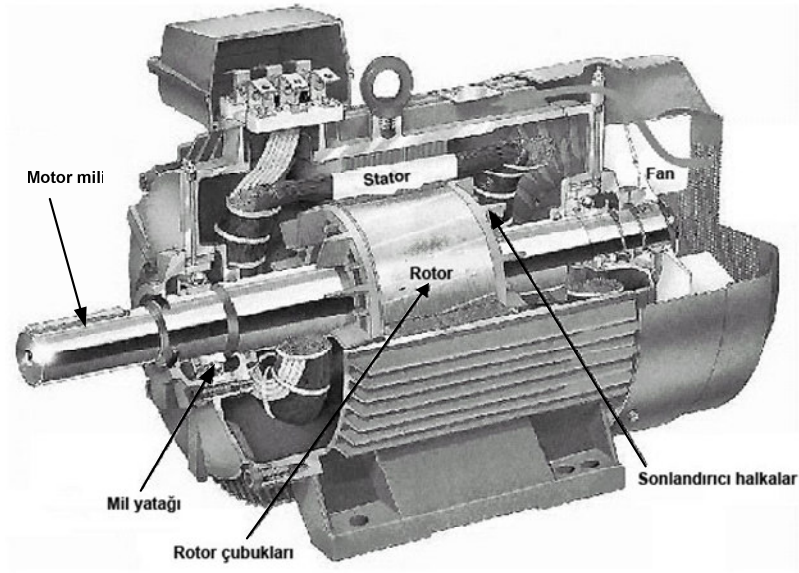
2.1. Asenkron Motorların Yapısı ve Çalışma Prensipleri

Endüstride, elektrik enerjisinin mekanik enerjiye dönüştürülmesinde en çok tercih edilen motor tipleri asenkron motorlardır. Asenkron motor, stator ve mil yatağına monte edilmiş bir rotora sahiptir. Rotor ve stator, bir hava boşluğu ile birbirinden ayrılır. DC motorlardan farklı olarak bu hava boşluğu asenkron motorlarda yoktur. Akım, direkt olarak stator sargılarına verilir ve rotor sargılarına ise indükleme ile verilir [14]. Bu özelliklerinden dolayı bu makineler indüksiyon motorlar da denilmektedir.

Asenkron motorların devir sayıları yükten çok fazla etkilenmediklerinden dolayı sabit devirli motor sınıfına girerler. Doğru akım şönt motorlarının devir sayısı büyük sınırlar içinde değiştirilebildiği halde, asenkron motorun devir sayısı sınırlı olarak bir veya iki kademe artırılabilir. Asenkron motorlar aşağıda verilen avantajlarından dolayı çok fazla tercih edilirler [15]:

- Asenkron motorlar daha ucuzdur.
- Sürekli bakıma ihtiyaç duymazlar.
- Elektronik devre kullanılarak devir sayılarının kolayca ayarlanması mümkündür.
- Asenkron motorların çalışması esnasında elektrik arkı oluşmaz.
- Yüklü çalışırken devir sayıları çok fazla değişmez.

Şekil 2.1’de üç fazlı bir asenkron motorun iç yapısı verilmiştir. Bu şekilden de görüldüğü gibi bir asenkron motorun en dış kısmında stator, statorun içinde oluşan manyetik alan ile dönen rotor ve rotorun üzerine monte edildiği bir mil yatağı görülmektedir. Motorun ısınma problemlerini gidermek için motora fan da monte edilir. Motor çalışmaya başladığı anda fan çalışarak motorun belirli bir ortam sıcaklığında tutulmasını sağlar.

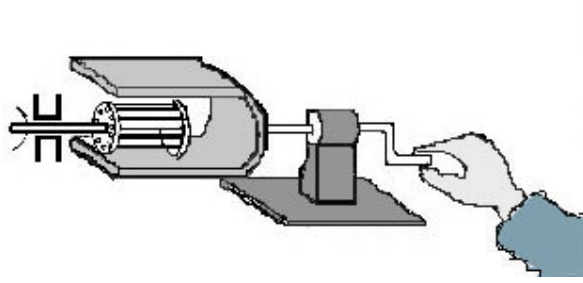


Şekil 2.1. Bir asenkron motorun yapısı [1]

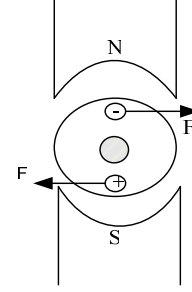
Stator, bir asenkron motorun sabit olan kısmıdır. Stator üzerine yerleştirilen sargılar, üçgen bağlı veya yıldız bağlı sargılardan oluşmaktadır. Stator sargılarından geçen akım, alternatif akım olduğundan manyetik devrede periyodik olarak değişen bir alternatif alan oluşturur. Alternatif alanın her bir harmoniğini iki döner alana ayırabiliriz. Bu döner alanlardan biri saat ibresi yönünde dönüyorsa, diğeri saat ibresinin tersi yönünde döner ve her ikisinin de açısal dönme hızı aynıdır. Üç fazlı asenkron motorlarda birbirinden 120 derecelik faz farkı ile oluşan akımlar, stator sargılarından geçerek üç adet alternatif alan meydana getirir [19].

Bir asenkron motorun mil yatağı üzerine monte edilmiş olan ve üzerinde bakır çubuklar bulunan kısmına rotor denir. Asenkron motorların tipini belirleyen bu kısım, üretim biçimine göre sincap kafes veya sargılı biçimde olabilir. Sincap kafes rotor aynı zamanda kısa devreli rotor olarak, sargılı rotor ise bilezikli rotor olarak isimlendirilir.

Asenkron motorların sağlam yapıda olmaları, ucuz olmaları, besleme gücü ile kolayca çalıştırılabilmeleri, düşük bakım maliyetleri gibi özelliklerine rağmen, genellikle elektrik enerjisinin mekanik enerjiye dönüşümü esnasında oluşan zorlanmalar motorun yapısını kötü yönde etkilemektedir [35]. Enerji dönüşümü bir taraftan mekaniksel zorlanma ve harekete, diğer taraftan manyetik ve elektrik alanlarıyla ilgili bir olayın varlığına bağlıdır. Bu enerji dönüşümü şekil 2.2 'de gösterildiği gibi gerçekleştirilir.



a) Gösterimsel çalışma biçimi



b) Rotor dönüşü

Şekil 2.2. Bir asenkron motorun çalışma prensibinin gösterimi [19]

Şekil 2.2 (a)'da gösterildiği gibi kalıcı bir mıknatıs el ile döndürülmektedir. Mıknatısın içine yerleştirilmiş sincap kafes rotor bir mil yatağı ile sabitlenmiştir. Rotor ile mıknatıs arasında herhangi bir sürtünme yoktur. Rotorun dönmesi tamamen arada oluşan manyetik alan ile olmaktadır. Sincap kafes rotor, sonlandırıcı halka veya plakalar ve bakır çubuklardan yapılır. Rotor çubukları, kapalı bir devre oluşturmak amacıyla sonlandırıcı halkalara kaynaklanmıştır. Mıknatısın dönme yönü ile rotorun dönme yönü aynıdır. Rotor mıknatısın dönme yönü ile aynı yönde döner. Mıknatıs saat yönünde döndüğünde rotor da saat yönünde döner. Mıknatıs saatin tersi yönünde döndürüldüğünde rotor da aynı şekilde saatin tersi yönünde döner. Mıknatısın dönme hızı yavaşlatıldığında veya hızlandırıldığında rotor da bu harekete doğru orantılı olarak davranır.

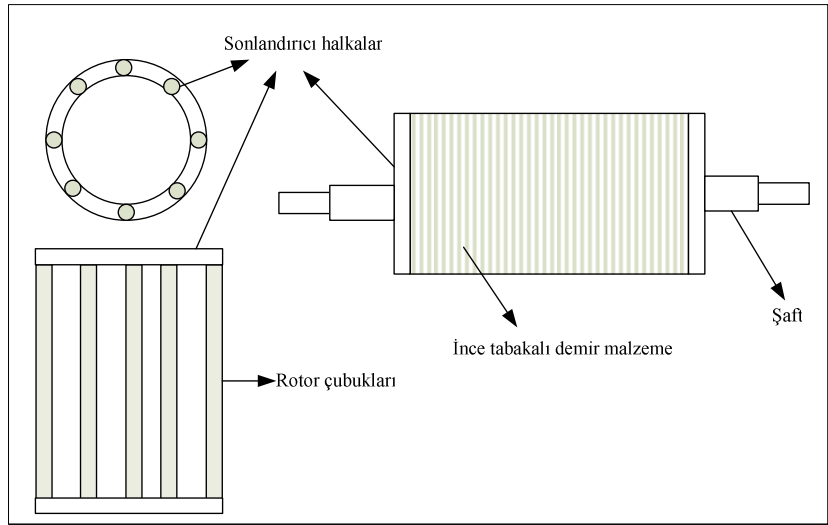
Bir asenkron motorun çalışma prensibi şekil 2.2 'de verilen gösterime benzerdir. Mıknatıs döndüğünde mıknatısın iki kutbu (N ve S) arasındaki akı rotor çubukları tarafından kesilir. Rotor çubuklarında bir gerilim indüklenir ve bu gerilim Faraday kanunları ile verilir. Gerilim kapalı devrede bir akım oluşturur. Rotordaki akım kendi manyetik alanını oluşturur. Akım taşıyan her bir rotor çubuğu etrafındaki manyetik alan kalıcı mıknatısın manyetik alanını etkiler. Sonuç olarak şekil 2.2 (b)'de gösterildiği gibi mekaniksel bir zorlanma kuvveti manyetik alandaki akım taşıyan her bir rotor çubuğuna uygulanır. Böylece bir elektromanyetik dönme momenti üretilir ve bu moment rotoru mıknatısın dönme yönünde dönmeye zorlar.

Rotorun dönme hızı, yük ve sürtünmeden dolayı mıknatısın dönme hızından daha yavaştır. Asenkron motorlarda bu hız farkı rotor kayması olarak bilinir. Rotor çubuklarında indüklenen akımlar kutuplar arasındaki manyetik akı ve rotor çubuklarının kestiği akı hızı ile orantılıdır. Mıknatıs hızlı döndüğünde manyetik akı ve çubuklar arasındaki ilgili hız artar. Dönme esnasında rotor çubuklarında indüklenen akım, rotorun manyetik alanının daha çok güçlenmesi ile sonuçlanarak artar. Böylece rotor daha hızlı döner.

Bir asenkron motorda yukarıdaki gösterimde, kullanılan kalıcı mıknatıs stator sargıları olarak isimlendirilen ve manyetik alan oluşturmak için kullanılan sargı bobinleri tarafından

yapılır. Değişken akım bir bobin grubundan geçtiğinde, sabit biçimde bir manyetik alan ve sinüzoidal olarak değişen genlikle sonuçlanacaktır. Bir manyetik kutup bu bobin grubunun merkezinde oluşur. Dahili stator sargı bağlantılarının dönme yönü, sargılara uygulanan gerilimi ve kutupların sayısını belirler. Akım sarımdan geçtiğinde iki zıt kutupta bir manyetik alan oluşur. Oluşan manyetik alan akım ile orantılıdır. Stator sargılarına üç faz akımı uygulandığında her bir faz akımı bir manyetik alan oluşturur. Aralarında 120 derece faz kayması olan üç faz akımı, ardışık sırada maksimum ve minimum değerlerine ulaşır. Manyetik alan, hat frekansı ve kutup sayıları ile belirlenen sabit bir hızda sürekli olarak döner.

Sincap kafesli bir asenkron motorun yapısı şekil 2.3'te gösterilmiştir. Tipik bir sincap kafes rotorunda her bir rotor çubuğu bir sonlandırıcı halka ile sonlandırılarak kapalı devre bir sistem oluşturmuştur. Rotor çubuklarının içerisindeki malzeme magnezyum, alüminyum veya bakırdan oluşur. Standart sincap kafes bir motorun rotor çubukları, düşük gerilimlerde büyük akımlar taşıdığından izolasyona sahip değildir.



Şekil 2.3. Tipik bir sincap kafes rotorun yapısı

Sincap kafes rotorların son derece basit ve sağlam yapılarından dolayı bu tipteki motorların diğer asenkron motorlardan daha çok kullanılmalarını sağlamıştır. Bu tip motor kaymadan dolayı statorun manyetik alanının dönme hızına yakın bir hızda çalışır. Bir asenkron motorun genel özelliği rotor şaftına bağlanan yükten dolayı hızda az miktarda bir düşmenin olmasıdır.

2.2. Asenkron Motorlarda Durum İzleme, Arıza Belirleme, Tahmin, Teşhis ve Kestirim Kavramları

Asenkron motorun herhangi bir bölümündeki bir arıza, oluşmadan önce genellikle performans düşüklükleri ile kendisini gösterir. Bunun için de performansın sürekli izlenmesi gerekir. Bu yüzden, durum izleme kavramı ortaya çıkmıştır. Durum izleme; arıza oluşmadan önce motor problemlerinin düzeltilmesi, analizi ve belirlenmesi amacıyla motor parametrelerinin grafiksel gösteriminin sunulması olarak tanımlanır. Bu yöntem kullanılarak, sistem çalışmasını kontrol etmek için direkt olarak uzman bir kişi kullanmak yerine bu işlerin bir deney düzeneği ile yapılması sağlanır.

Erken aşamada arıza belirleme, teşhis ve tahmin yöntemleri aksaklıklar sırasında motor sistemi için ucuz maliyetli bir bakımın yapılmasına imkan sağlayarak sistem performansını ve kullanılabilirliğini geliştirir. Bu yöntemlerden bazıları, arıza tespiti için motorun içyapısındaki donanımsal kısımları incelemeyi gerektirir. Bunlar hem pahalıdır hem de gerçek zamanlı arıza tespiti yapmazlar. Genellikle motor akımı, hız, gerilim ve moment gibi gerçek zamanda düşük maliyet ile ölçülebilir sinyaller hem arızayı erken bir aşamada tespit etmek hem de motorun yapısına zarar vermemek açısından kullanışlı yöntemlerdir. Kritik uygulamalarda, motorların anlık izlenmesi büyük arızaları minimize etmek ve makina güvenilirliğini arttırmak için bir gereklilik olmuştur.

Arıza belirleme, sağlam bir motorun çalışma sınırları dışında bir durumun olup olmadığını belirler [36]. Arıza belirlemenin amacı, eğer sistemde arıza varsa anlık ölçümler kullanılarak bu arızaları belirlemektir. Arızaların yanlış tespit edilme ihtimallerinin ve hatalı arıza uyarılarının elenmesi bu aşamada yapılır.

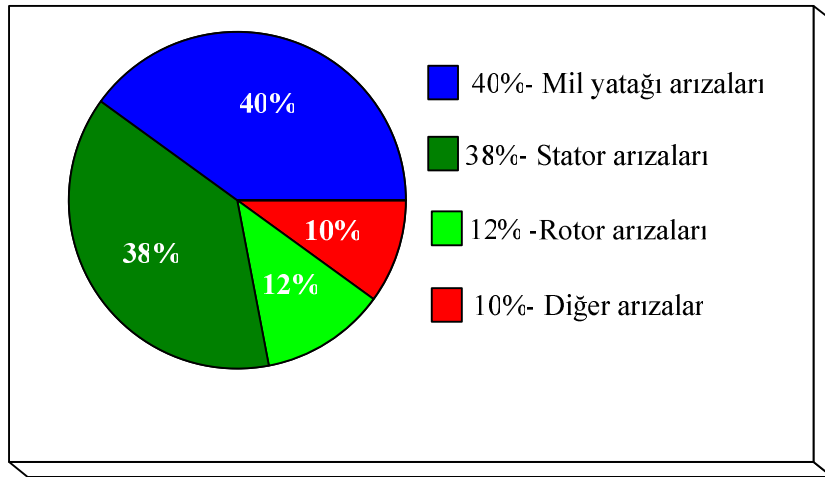
Arıza teşhisinde sadece sistemde arıza varsa belirlemek değil aynı zamanda onun sebeplerini bulmak ve arızayı izole etmek istenir. Bu aşamada türü, temel sebebi ve yeri bilinen bir arıza durumuna karşılık gelen bilginin çıkarılması sağlanır. Benzer karakteristikler gösteren arızaların ayırt edilmesi bu aşamada yapılır.

Arıza tahmininde ise arıza teşhisine ilave olarak donanım bileşenlerinin kalan ömrünün tahmin edilmesi de beklenir. Dönen motorlarda oluşan arızaların bazen bir sonraki aşaması daha ciddi arızalar olup, ciddi performans düşüşü veya motor durumunun tamamen bozulması kaçınılmazdır. Bu durumlarda, arıza tahmin yöntemlerinin bir sonraki adımda motorun durumu hakkında gerçek zamanlı olarak bilgi vermesi endüstriyel süreçlerde üretimin devamlılığı için önemli bir avantaj sağlar. Örneğin, motorda hava boşluğu eksantrikliği oluşması tespit edilmiş ise bir sonraki aşamada oluşacak arızanın stator ile rotor arasındaki sürtünmeler olacağı ve gerekli önlemlerin alınması gerektiği ikazı verilebilir.

2.3. Asenkron Motorlarda Oluşan Arıza Türlerinin Sebepleri ve Etkiledikleri Motor Parametreleri

Asenkron motorlarda bir arıza oluşması durumunda devre kesici ve sigorta gibi cihazlar motorun yanında çalışan kişiyi yaralanmalara karşı koruyabilmesine karşın, bu cihazların arıza belirleme özelliklerinin olmadığından daha önce bahsedilmişti. Bu cihazlar asenkron motorların imalatı esnasında motora monte edilirler. Asenkron motorlardaki arızaların büyük çoğunluğu, elektriksel ve mekaniksel zorlanmalar ile ortama ve sıcaklığa bağlı zorlanmalardan dolayı oluşur [35]. Son zamanlarda parçalarının üretimi ve tasarımının geliştirilmesinde belirgin ilerlemeler olmuştur. Bununla birlikte çeviriciler ile sürülen asenkron motorlar, hızlı açma ve kapama problemlerinden dolayı gerilim zorlanmalarına maruz kalmaktadırlar. Stator sargılarının korunması için geliştirilen yalıtım maddeleri, bu gibi durumlardan sargının minimum zarar ile çıkmasını sağlarlar. Fakat rotor tasarımı üzerine daha az gelişme sağlanmıştır. Böylece son yıllarda rotor arızalarında önemli artışlar olmuştur [1].

Asenkron motorlarda oluşan arızalar literatürde stator ile ilgili, rotor ile ilgili veya mil yatağı ile ilgili olmalarına göre kategorilere ayrılmıştır. Şekil 2.4'te asenkron motorlarda en sık oluşan arıza tipleri ve bunların yüzdelik dilimleri verilmiştir [4].

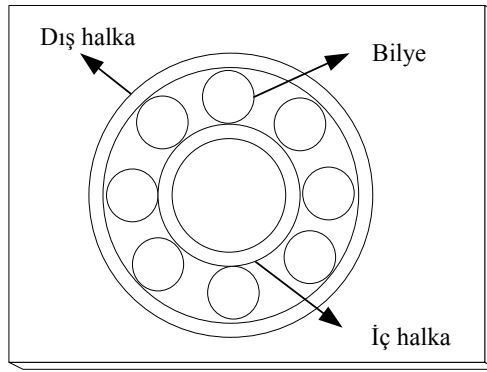


Şekil 2.4. Asenkron motor arıza tiplerinin oluşum yüzdeleri

Düşük ve yüksek güçlü asenkron motorların arıza istatistikleri geçmiş birkaç yıl içerisinde değişmiştir. Elektriksel ve manyetik arızalara karşılık, mekaniksel hatalarda önemli bir artış olmuştur.

2.3.1. Mil Yatağı Arızaları

Asenkron motorlarında en çok oluşan arıza türü mil yatağı problemleridir. Mil yatağının eğilmesi, milin sürtünmesi, mil yatağındaki bilyelerde oluşan arızalar en çok karşılaşılan mil yatağı arıza türleridir. Mil yatağının eskimesi, yağ sızıntısı ve yağın eskimesi ile mil yatağında sürtünme arızaları oluşur [2]. Oluşan bu sürtünmeler, erken bir aşamada giderilmezse mil yatağının kilitlenmesi ve motorun yanmasına sebep olur. Mil yatağında bilyeler iç ve dış halka olmak üzere iki halka içinde bulunmaktadır. Bu halkalar içinde yerleştirilmiş olan bilyeler motorun çalışması ile hareket ederler. Şekil 2.5'te mil yatağının yapısı verilmiştir.



Şekil 2.5. Mil yatağının yapısı

Dengeli yük ve sağlam bir motorun çalışmasında da gürültü ve titreşim oluşabilir. Motorun mekaniksel bölümündeki aşırı sıcaklık, mil yatağının aşınmasına sebep olur ve sürtünme arızalarına yol açar. Mil yatağı bilyeleri rotora destek oluşturduğundan, burada oluşacak herhangi bir arıza stator ve rotor arasında radyal bir zorlanma oluşturur [1]. Mil yatağında oluşan sürtünmelerden dolayı güç kayıpları oluşur. Rotor ile ilgili kayıplar kaymanın değişmesi ile oluşur. Stator kayıpları ise stator sarımlarında oluşan bakır kayıpları ile ilgilidir.

2.3.2. Rotor Arızaları

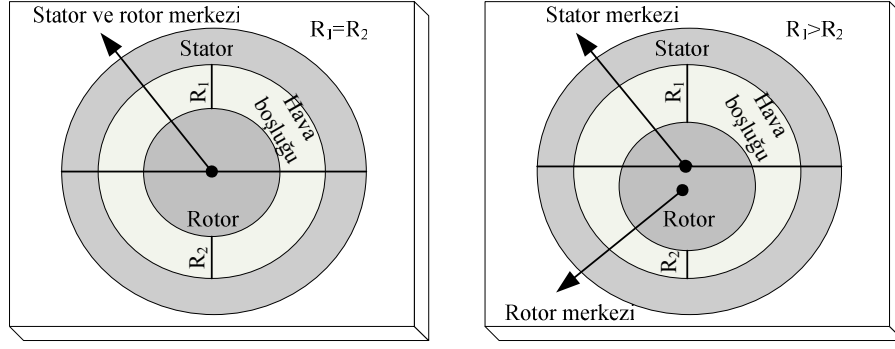
Asenkron motorlarda oluşan rotor arızaları, kırık rotor ve kırık sonlandırıcı halkalardır. Bu arızalar genelde elektromanyetik, termal, çevresel ve mekaniksel gibi sebeplerden oluşur. Rotorda termal aşırı yük durumu çalıştırma, durdurma veya arızalanma durumlarında oluşabilir. Motor kilitlendiğinde veya arızalandığında, rotorda problemler oluşabilir. Statoru korumak için kullanılan termal aşırı yükleme devreleri motorlara monte edilmiştir. Fakat bu devreler statordaki aşırı yükü algılayana kadar rotorda problemler oluşabilir. Yüksek rotor ısınmalarına

yol açan motor çalıştırmaları, rotor arızalanması, mil yatağı arızalarından dolayı oluşan rotor sürtünmeleri, rotorun sonlandırıcı halkalarının büyümesi ile oluşan kırık rotor arızaları ve rotorun ısınmasına yol açan dengesiz faz gerilimi termal aşırı yük problemlerine yol açar [37]. Ortamın nemli olması, havalandırma eksikliği, kırık rotor arızalarına yol açar. Mil yatağında oluşan sürtünme ve milin eğilmesi gibi mekanik arızalar da rotor arızalarına sebep olur. Kırık rotor arızalarının gelişimi aşağıda adım adım verilmiştir [15]:

- Rotor çubuğu çatlak olan noktada aşırı ısınır.
- Çubuğun etrafında rotor tabakasını kötüleştiren kırığa karşı bir ark oluşur.
- Kırık rotor üzerinden akım geçmediği için sağında ve solunda bulunan rotor çubukları daha fazla akım çeker.
- Rotor arızalarından dolayı oluşan merkezkaç kuvveti stator sargı arızalarını oluşturur.

2.3.3. Hava Boşluğu Eksantrikliği Arızaları

Asenkron motorlarda stator ile rotor arasında belirli bir hava boşluğu bulunmaktadır. Bu hava boşluğunun değişmesi ile hava boşluğu eksantrikliği arızaları oluşur. Hava boşluğu eksantrikliği arızaları, statik ve dinamik olarak iki şekildedir. Stator çekirdeğinin oval biçimde olması veya rotorun yanlış bir konumda olması statik eksantriklik arızasına sebep olur [20]. Dinamik eksantriklik, rotorun merkezi dönme merkezinde olmadığı ve minimum hava boşluğu rotor ile birlikte döndüğünde oluşur. Minimum hava boşluğu eksantrikliğinin oluşmaya başlama aşamasında dengesiz bir manyetik çekim meydana gelir ve bu çekim rotorun kötüleşmesine sebep olur böylece hava boşluğu eksantrikliği artar. Eğer hava boşluğu eksantrikliği belirli bir sınırı aşarsa, makinede bazı zıt etkiler motor performansını düşürür. Bu etkiler dengesiz manyetik çekimlere yol açar. Dinamik eksantriklik arızası rotor şaftının eğilmesi, mil yatağı aşınması veya dengesizliği ve kritik hızlarda oluşan manyetik rezonans gibi durumlarda ortaya çıkar. Şekil 2.6'da rotor ile stator arasındaki hava boşluğu normal olma durumu ve eksantriklik olması durumu verilmiştir.



a) Normal hava boşluğu

b) Eksantriklik arızası

Şekil 2.6. Hava boşluğu eksantrikliğinin şematik gösterimi

Eksantriklik arızaları % ile ifade edilir. Dinamik eksantriklik denklem (2.1) ile belirlenebilir.

$$\%De = \frac{Nhb - Minhb}{Nhb} * 100 \quad (2.1)$$

Denklem 2.1’de de dinamik eksantrikliği gösterir. Nhb nominal hava boşluğunu gösterir ve denklem 2.2 ile ifade edilir.

$$Nhb = \frac{R_1 + R_2}{2} \quad (2.2)$$

Denklem 2.2’de R_1 ve R_2 şekil 2.6’da gösterilen hava boşluğu kısımlarıdır. Örneğin şekil 2.6’da $R_1=75$ ve $R_2=25$ ise eksantriklik aşağıdaki gibi hesaplanabilir.

$$Nhb = \frac{(75 + 25)}{2} = \frac{100}{2} = 50 \quad (2.3)$$

$$De = \frac{50 - 25}{50} * 100 = \%50 \quad (2.4)$$

Burada R_2 gerçek minimum hava boşluğunu göstermektedir ve rotor ile statorun merkezi farklı noktalarda olduğu için dinamik eksantriklik arızası vardır.

2.3.4. Stator Arızaları

Motorun stator sargılarında oluşan arızalar, elektriksel sebepler ile oluşan önemli arızalardandır. Bir asenkron motorun stator sargısı sargı devirlerinin kısa devre olmasını önlemek amacıyla yalıtım yapılmıştır. Bu yalıtım, farklı maksimum çalışma sıcaklıklarına orantılı olarak farklı sınıflara ayrılmıştır [2]. Fakat bu maksimum çalışma sıcaklıklarına çalışma ortamları ve yük durumlarından dolayı uyulmayabilir. Bu ihmalden dolayı stator sargılarının belirli kısımlarında incelmeye, yalıtım kayıpları ve çatlaklar oluşur. Bu durum stator sargılarının kısa devre olmasına ve stator sarım sayılarının düşmesine yol açar.

Stator arızaları termal, elektriksel, mekaniksel ve ortama bağlı nedenlerden dolayı oluşmaktadır. Termal zorlanmalar ısınma ve aşınmadan dolayı oluşur. Sıcaklıktaki her 10^0 derece artış izolasyonun ömrünün ikiye bölünmesine neden olur. Sıcaklık veya termal aşınmanın etkisi çalışma sıcaklığı düşürülerek veya kullanılan yalıtkan malzemelerinin kalitesi artırılarak düşürülebilir.

Yukarıda anlatılan arızaların teşhisi için farklı yöntemler kullanılmıştır. Asenkron motorlarda arıza teşhisi için akım, gerilim, moment, titreşim, hız gibi parametreler ölçülerek yapılan analizlere ek olarak parametre tahmini gibi diğer yöntemlerde literatürde geniş bir şekilde uygulanmıştır. Çizelge 2.1 'de literatürde en çok karşılaşılan arızalar ve bu arızalara göre kullanılan arıza teşhis yöntemleri verilmiştir.

Çizelge 2.1. Arıza teşhisi için kullanılan yöntemler

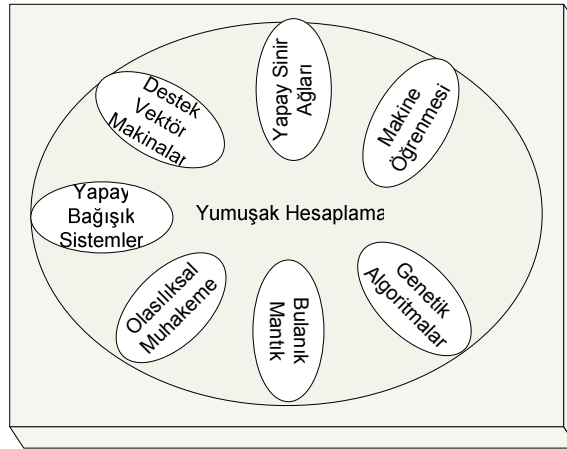
Yöntem	Teşhis Edilen Arızalar				
	Stator izolasyon hataları	Stator sargı hataları	Rotor hataları	Hava boşluğu eksantrikliği hataları	Mil yatağı hataları
Titreşim analizi	X	X	√	√	√
Motor akım imza analizi	X	√	√	√	√
Eksensel akı	X	√	√	√	X
Yağlanma yağının incelenmesi	X	X	√	X	√
Kısmi boşalma	√	√	√	X	X
Soğutma gazı	√	X	X	X	X

Literatürde her bir arıza türü için farklı motor parametreleri ölçülmüştür. Stator arızalarını teşhis etmek için yapılan çalışmalar genelde akım ve gerilim gibi sinyalleri kullanmıştır. Kırık rotor arızalarının teşhisi için motorun akım bilgisi yanında hız bilgisi de ölçülmüştür. Motor akımı üzerinde motor akım imza analizi, park vektör dönüşümü gibi işaret işleme teknikleri uygulanarak elde edilen özellikler incelenmiş ve arıza şiddeti değerlendirilmiştir. Mil yatağı arızalarının teşhisinde motor akımı, titreşim gibi sinyaller ölçülmüştür. Özellikle mil yatağındaki bilye arızaları için titreşim analizi daha çok kullanılmıştır. Hava boşluğu eksantrikliği arızaları motor akımı ve hızı ölçülerek belirlenmiştir.

3. YUMUŞAK HESAPLAMA TEKNİKLERİ

3.1. Giriş

Doğada karşımıza çıkan bütün problemler için tam çözüm bulmak her zaman mümkün olmayabilir. Bazı durumlarda problemlerin tam çözümünü bulmak maliyetli olduğundan, daha esnek ve doğruya yakın çözümler bulmak için yeni yöntemlere ihtiyaç duyulmuştur. İnsanın düşünme tarzı, öğrenme yetenekleri ve insanın genetik yapısından esinlenerek mühendislik alanındaki problemlere çözümler geliştirilmeye çalışılmıştır. Deneysel verilerden öğrenme ve bulanık mantık metotları, yumuşak hesaplamanın en önemli bileşenleridir. Ayrıca genetik veya evrimsel algoritmalar, olasılıksal muhakeme, kaos teorisi yumuşak hesaplamanın diğer bileşenleridir [39]. İnsanın bağışık sisteminin bir benzetimi olarak geliştirilen yapay bağışık sistemler; optimizasyon, öğrenme ve anomali tespiti gibi birçok mühendislik ve bilgisayar probleminde etkili çözümler getiren, yeni bir yumuşak hesaplama bileşenidir. Şekil 3.1 'de yumuşak hesaplamanın temel bileşenleri gösterilmiştir.



Şekil 3.1. Yumuşak hesaplama bileşenleri

Şekil 3.1 'de verilen yumuşak hesaplama bileşenlerinden yapay sinir ağları öğrenme, eğri uydurma lineer olmayan denklemlerin çözümü gibi alanlarda uygulanmaktadır. Bulanık mantık ise matematiksel denklemi tam olarak çıkarılamayan sistemlerin çözümünde kullanılan kontrol amaçlı bir hesaplama yöntemidir. Genetik algoritmalar, kör bir arama metodu olup optimizasyon problemlerinin çözümüne uygulanır. Yapay bağışık sistemler, normal çalışan bir sistemdeki aksaklıkları tespit etmek, optimizasyon, örüntü tanıma ve öğrenme problemlerini çözmek için geliştirilmiştir [9]. Destek vektör makinalar, kullandıkları özel kernel polinomları

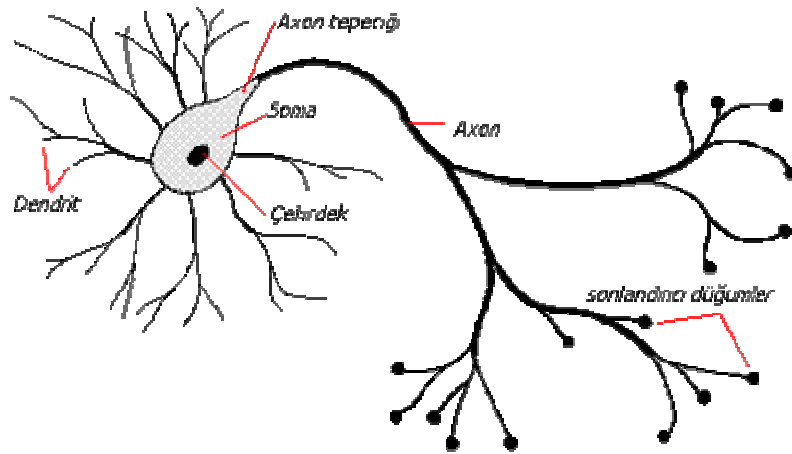
ile lineer olmayan sınıflandırma problemleri için kullanılırlar. Bu yöntemlerin hepsinde ortak olan nokta, tam ve keskin olan bir çözüm kümesinin bulunmamasıdır. Problemler çözülürken, belirli bir hata toleransına göre çözüm kümesi veya kümeleri oluşturulur. Bu yönü ile yumuşak hesaplama yöntemleri, keskin ve tam çözümler isteyen sert hesaplardan ayrılır. Yumuşak hesaplama, insan düşüncesinin önemli yeteneklerini kullanarak keskin ve tam doğru olmayan sistemleri öğrenebilme yeteneği ile akıllı hesaplama için geliştirilmiş bir yaklaşım olarak düşünülür [40]. Zadeh [8]'e göre yumuşak hesaplama kavramı iki özellik ile verilebilir:

- Gerçek dünya keskin ve tam olmayan bir biçimdedir.
- Keskinlik ve tam olma maliyetli olabilir.

Yumuşak hesaplama teknikleri, mühendislik alanında birçok alanda kullanılmasına rağmen, bu çalışmada arıza teşhisi için kullanacağımız yapay sinir ağları, bulanık mantık ve yapay bağışık sistemlerden detaylı olarak bahsedilecektir.

3.2. Yapay Sinir Ağları

İnsan beyni, nöron olarak adlandırılan yaklaşık 10^{11} sinir hücresinden oluşmaktadır. Sinir hücreleri farklı yapıda olmalarına rağmen, birbirlerine bağlanma açısından benzerlikler taşırlar. Bir biyolojik sinir hücresi hücre gövdesi, akson ve dendritlerden oluşur. Bu elemanların her biri bilgi girişi, işlenmesi ve iletiminde farklı görevleri üstlenmektedirler. Şekil 3.2 'de biyolojik bir sinir hücresinin genel yapısı verilmiştir.



Şekil 3.2. Biyolojik sinir hücresi

Şekil 3.2'ye göre dış ortamdan bilgi girişi, dendritler üzerinden alınır ve hücre gövdesine iletilir. Aksonlar ise elektriksel darbeler şeklindeki bilgiyi hücreden dışarı taşıyan

daha uzun bir yoldur. Akson ile dentrit bağlantısı, sinaps olarak adlandırılır. Dentriteler tarafından alınan ve sinapse gelen bilgiler genellikle elektriksel sinyallerdir. Belirli bir süre boyunca hücreye gelen girişler belirli bir eşik değerini aştığında hücre bir tepki oluşturur. İnsan beyni yaklaşık 10 milyar sinir hücresinden oluştuğundan, son derece karmaşık bir yapıya sahiptir. Bu nedenle öğrenme, genelleme gibi özellikleri ile paralel dağıtılmış bir işleme birimidir. İnsan beyninin bu çalışma şekli model alınarak yapay sinir ağları geliştirilmiştir.

Yapay sinir ağları, insanın sinir sistemi ve biyolojik nöronların davranışı ve yapısından esinlenilerek geliştirilmiş modellerdir [39]. Daha genel bir ifade ile yapay sinir ağları, insan beynindeki birçok nöronun veya yapay olarak basit işlemcilerin birbirlerine farklı etki seviyeleri ile bağlanması sonucu oluşan karmaşık bir sistem olarak düşünülebilir. Mühendislik alanında yapay sinir ağlarının geniş bir şekilde kullanılmasının en önemli nedeni, klasik teknikler kullanılarak çözümü karmaşık olan problemler için etkili bir alternatif oluşturmalarıdır [2]. Geleneksel bilgisayarlar, belirli bir görevi yerine getirmek için programlanır. Yapay sinir ağları ise bunlardan farklı olarak eğitilir, öğrenir ve eldeki problem için kullanılır. Kullanıcılar yapay sinir ağları kullanarak, görevlerini yapmak için komutların bir kümesini seçmez. Bunun yerine, kullanıcılar eldeki problemin ilişkilerini ve örüntülerini öğrenmek için uygun sinir ağı mimarisini ve uygun başlangıç ağırlıklarını seçer. Yapay sinir ağlarının temel özellikleri aşağıda verilmiştir:

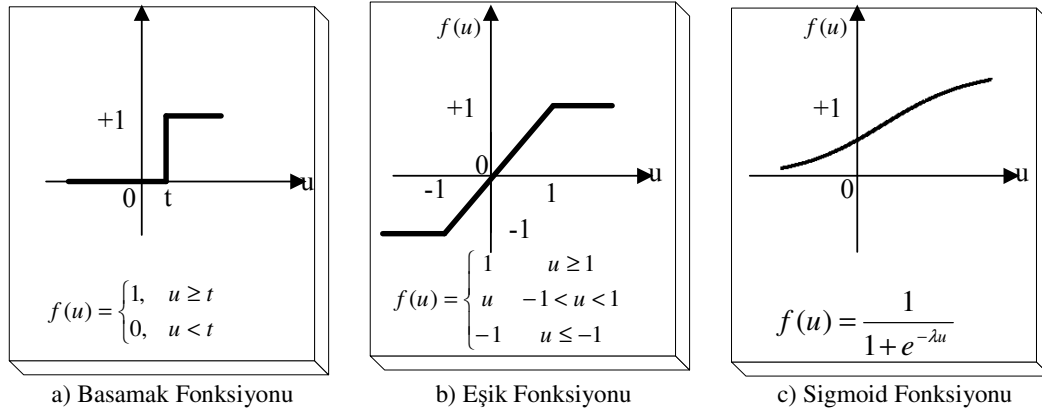
- **Lineer olmayan karmaşık problemlere uygulanma:** Yapay sinir ağları lineer olmayan problemleri kolayca öğrenebilir.
- **Genelleme:** Yapay sinir ağı herhangi bir problemi öğrendikten sonra farklı örnekler için genelleme yapabilir.
- **Sınıflandırma özelliği:** Yapay sinir ağları verilen kümeleri sınıflandırmak için kullanılabilir.
- **Hızlı çalışma:** Yapay sinir ağları paralel yapılarından dolayı oldukça hızlı çalışırlar.

Yapay sinir ağları ilk ortaya atıldığından beri, birçok yapay sinir ağı modeli geliştirilmiştir. Bunlardan bazıları ileri beslemeli yapay sinir ağları, Hopfield ağları, Kohonen ağları, Adaptif rezonans teorisi, ve radyal tabanlı fonksiyon ağları olarak verilebilir. Bu sinir ağı mimarilerinden, ileri beslemeli yapay sinir ağları bugünkü yapay sinir ağlarının %80'nini oluşturmakta olup en popüler sinir ağı mimarilerindendir [2].

$$u(t) = w_1(t)x_1(t) + w_2(t)x_2(t) + \dots + w_n(t)x_n(t) - w_0 \quad (3.1)$$

$$y(t) = f(u(t)) = f\left(\sum_{i=1}^n w_i x_i(t) - w_0\right) \quad (3.2)$$

Denklem (3.2)'deki f sembolü, nöronun çıkışına uygulanacak aktivasyon fonksiyonunu gösterir. Aktivasyon fonksiyonu, diferansiyeli alınabilir ve monoton olarak azalmayan bir fonksiyon olarak seçilir. Sigmoid, basamak, eşik ve doğrusal gibi aktivasyon fonksiyonları kullanılabilir. Sinir hücresinin davranışı, seçilen aktivasyon fonksiyonunu ile belirlenir [39]. Şekil 3.4'te en çok kullanılan aktivasyon fonksiyonları verilmiştir.



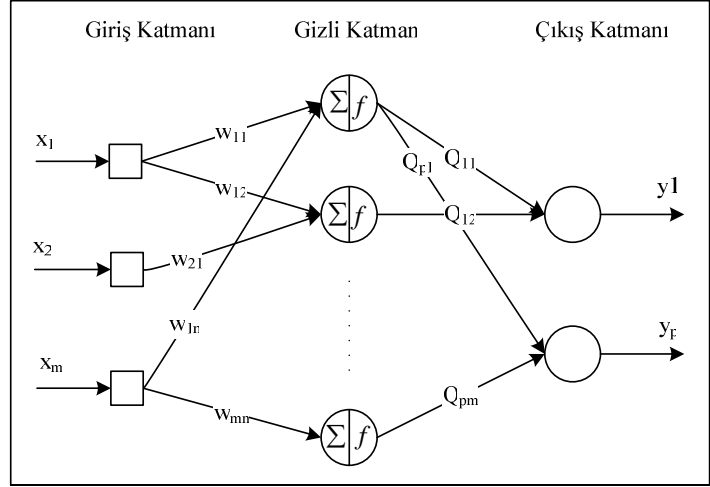
Şekil 3.4. Aktivasyon fonksiyonları

Yapay sinir ağına verilen bir örnek, ağın önceki öğretilmiş durumlarından çok farklı olursa, ağırlıklarını değiştirerek benzerliği arttırmak mümkündür. Aktivasyon fonksiyonu burada önemli bir rol oynamaktadır. Sigmoid fonksiyonlar biyolojik nöronların davranışlarını anımsattıklarından dolayı, en popüler aktivasyon fonksiyonlarından biridir [40].

3.2.2. İleri Beslemeli Yapay Sinir Ağları

Bir yapay sinir ağı tarafından öğrenilecek problemin giriş ile çıkışları arasında doğrusal olmayan ilişkiler varsa çok katmanlı yapay sinir ağı kullanılmalıdır [41]. Böyle bir yapay sinir ağı bir giriş katmanı, bir veya daha çok gizli katman ve bir çıkış katmanından oluşur. Bu tip ağda nöronlar birbirlerine belirli bir düzende bağlıdırlar. Giriş katmanındaki nöronlardan gizli katmandaki bir nörona bir veya birden fazla giriş olabilir. Yapay sinir ağları veri akışına göre iki gruba ayrılır: İleri beslemeli yapay sinir ağları ve geri beslemeli yapay sinir ağları. Ara nöronlardan işlenerek giriş katmanından çıkış katmanına doğru işaretlerin yalnız bir yönde

ilerlediği ağlara ileri beslemeli yapay sinir ağları denir. Herhangi bir nöronun çıkışından daha önceki giriş nöronlarına işaretin ulaştığı ağlara ise geri beslemeli sinir ağları denir. Şekil 3.5 'te 3 katmanlı ileri beslemeli bir yapay sinir ağı verilmiştir.



Şekil 3.5. İleri beslemeli yapay sinir ağı yapısı

İleri beslemeli yapay sinir ağında giriş katmanı, girişe verilen değerleri işlemeden diğer katmana iletir. Her bir düğüme $+1$ veya -1 bias değeri de eklenir. Gizli veya çıkış katmanlarındaki her bir düğüm daha alt katmanlardaki düğümlerin her birinden ağırlıklandırılmış girişler alır. Bu girişlerin toplamı nöronun net çıkışını belirler. Nöronun gerçek çıkışı bu net çıkışın aktivasyon fonksiyonundan geçirilmesi ile elde edilir.

İleri beslemeli sinir ağında her nöron, tek başına hesaplama veya gösterim için çok güçlü değildir. Aynı katmandaki nöronların bir diğerine bağlantısı yoktur, fakat giriş sinyalleri işletildikten sonra önceki katmandan sinyalleri alır ve sonraki katmana sinyalleri iletir. Bağlantıların her biri onunla ilgili bir ağırlığa sahiptir. Bu ağırlıkların değerleri, uygun eğitim algoritması ile değiştirilerek yapay sinir ağının problemi öğrenmesi sağlanır.

Üç katmanlı ileri beslemeli yapay sinir ağları, karmaşık sürekli lineer olmayan bir fonksiyonu öğrenebilir. Bununla birlikte, verilen problemi çözmek için gereken düğümlerin tam sayısını belirleyecek bir algoritma veya teori yoktur. Genellikle deneme yanılma yoluyla düğüm sayıları bulunur. Bazı uygulamalarda, genetik algoritmalar da en optimum yapay sinir ağı ağırlıklarını belirlemede kullanılabilir.

3.2.3. Geriye Yayılım Öğrenme Algoritması

Çok katmanlı ileri beslemeli yapay sinir ağlarında en çok kullanılan öğrenme yöntemlerinden biri geriye yayılım öğrenme algoritmasıdır. Geriye yayılım öğrenme algoritması, 0 ile 1 arasında reel bir sayı üreten sigmoid fonksiyonunu aktivasyon fonksiyonu olarak kullanır [40]. Geriye yayımlı öğrenme algoritmasında başlangıçta ağırlıklar rasgele olarak seçilir. Bazı uygulamalarda, yakınsamanın tek koşulu rasgele değerlerin uygun seçimine bağlıdır. En uygun rasgele değer üretecek fonksiyonun bulunması, ağıın birçok kez eğitime bağlıdır. Bu değerler ağıın öğrenme hızını da önemli ölçüde etkileyecektir. Bir öğrenme kuralına göre bu ağırlıklar değiştirilerek yapay sinir ağıının eğitim kümesi alınır ve verilen herhangi bir örüntüyü tanımak için eğitilebilir. Şekil 3.5'te de görüldüğü gibi çok katmanlı ileri beslemeli bir yapay sinir ağı bir giriş, bir veya birden çok gizli katman ve bir çıkış katmanından oluşur. Giriş katmanına uygulanacak vektörler karışık bir düzende verilir. Gizli katman bir anlamda ağıın zeka içeren kısmıdır. Gizli katmanda nöron sayısı fazla olursa ağı ezberlemeye yönelir. Bu da insan beynindeki nöronlarda bilgi tutulmasına benzer. Bir metni veya görüntüyü hatasız olarak beyninde tutan bir insan, beynindeki birçok nöronu bu iş için harcamıştır. Ancak, hatırlaması gerekenleri beyninde bilgi grupları arasında kurmuş olduğu ilişkilerden yola çıkarak hatırlayan insanlar, az sayıda nöronla çok fazla bilgiyi beyinlerinde depolarlar.

Ağıın ilk ağırlıkları rasgele belirlendikten sonra çıkış geriye yayılım algoritması ile ağırlıklar ayarlamak için tekrar kullanılır. Optimum ağırlıklar elde edilinceye kadar bu işleme devam edilir. Aşağıda geriye yayılım eğitim algoritması verilmiştir:

1. Orta katman ve çıkış katmanındaki her bir nörona I giriş uygulanır:

$$net_i = \sum_{j=0}^n w_{ij}x_j \quad (3.3)$$

Burada x önceki katmandaki nöronun çıkışı, h bias değeri ve w ise ağırlık değeridir.

2. Orta ve çıkış katmanındaki nöron çıkışı $y = f(I)$ ile hesaplanır ve $f(I)$ fonksiyonu aşağıdaki denklem ile hesaplanır.

$$f(I) = \frac{1}{1 + e^{-I}} \quad (3.4)$$

3. Eğitim esnasında ağırlıklar aşağıda verilen delta kuralına göre güncellenir.

$$\Delta w_{ij}^{q+1} = \eta E f(I) + \alpha \Delta w_{ij}^q \quad (3.5)$$

Burada η öğrenme oranı, α momentum katsayısı ve E hatayı temsil eder. α ve η 'nin değeri 0 ile 1.0 aralığında uygulamaya bağlı olarak seçilebilir.

4. Çıkış katmanındaki hata istenen çıkıştan ağ çıkışının çıkarılması ile elde edilir.

$$E_i^{cıkı} = y_i^{istenen} - y_i^{ysa} \quad (3.6)$$

5. Hata değeri çıkış katmanından orta katmana ve buradan giriş katmanına geriye doğru yayılır. Aşağıdaki denklemde orta katmandaki hatanın nasıl belirleneceği verilmiştir.

$$E_i^{orta} = \frac{df(I_i^{orta})}{dI} \sum_{j=1}^n w_{ij} E_i^{cıkıs} \quad (3.7)$$

6. Ağırlıklar hata değerine göre yeniden düzenlendikten sonra optimum ağırlıklar elde edilinceye kadar 2. adıma tekrar gidilir.

Yapay sinir ağı eğitildikten ve uygun ağırlıklar belirlendikten sonra, uygun bir giriş örüntüsü verilerek çıkışın doğruluğu test edilebilir. Eğitim aşaması, çevrimdışı olarak uygulanırken test aşaması çevrimiçi yapılır.

Geriye yayılım algoritmasının en önemli dezavantajı, yavaş yakınsama özelliğidir. Geriye yayılım hata yüzeyi genel olarak bir ağırlık yönü boyunca yassılaşır. Bundan dolayı, ani eğim bilgisi genlikte küçük olacaktır. Böylece algoritmanın istenen performansa ulaşması için birçok adım uygulanması gerekecektir [42]. Öğrenme oranının uyarlanması bu zorlukları giderebilecek kullanışlı bir metottur. Bu uyarlanma aşağıda verilmiştir.

$$\Delta \eta = \begin{cases} +\gamma & \Delta E < 0 \\ -\beta \eta & \Delta E > 0 \\ 0 & \text{diger} \end{cases} \quad (3.8)$$

Denklem 3.8'e göre, maliyet fonksiyonunda bir azalma olursa, ağırlık değişim vektöründeki büyüklükler yakınsamayı hızlandırmak için azalır ve öğrenme oranı γ gibi bir sabit eklenerek arttırılabilir. Eğer maliyet fonksiyonu geçmiş iterasyonlardaki birkaç ağırlık değişiminden arttırılırsa yörünge minimumu aşmaya yönelir. Bu yüzden öğrenme oranı

$(1 - \beta)$ ile azaltılabilir. Momentum katsayısının ağırlık güncelleme formülüne eklenmesi ile yerel minimumlardan kurtulabiliriz.

3.3. Bulanık Mantık

Bulanık küme kavramı, ilk kez 1965 yılında Prof. Lotfi Zadeh tarafından önerildi. Klasik küme teorisinde bir önerme doğru veya yanlış gibi keskin geçişler içerir. Fakat gerçek dünyadaki olayların ne derece yanlış veya ne derece doğru olduğunun belirlenmesi önemli bir gereksinimdir. Örneğin, klasik küme teorisinde 60 yaşında olan bir kişi ile 85 yaşında olan bir kişi yaşlı kategorisindedir. Fakat biz 85 yaşındaki birinin 65 yaşındakinden daha yaşlı olduğunu biliriz. Klasik küme teorisinde bu durum, yaşlı ve genç gibi iki küme ile ifade edilebilir. Fakat bulanık küme teorisinde yaşlı, daha yaşlı, genç, daha genç gibi ifadeler ile derecelendirmek mümkündür. Özelliği tam anlaşılamayan, belirsiz ve açık olmayan şeklinde tanımlanan bulanıklık, dereceli üyelik kavramı yardımı ile bilim ve mühendislik uygulamalarında geniş bir şekilde yer bulmuştur. Geleneksel küme teorisinin bir genelleştirmesi olarak, bulanık küme teorisi gerçek fiziksel dünyanın kesinsizliği ile ilgili bir metot olarak kullanılabilir. Bu yöntem, belirsizlikleri ve kesinsizlikleri göstermek için bir ölçüm sağlar. Bu kesinsizlik ve bulanıklık, bulanık mantığın çekirdeğidir [39]. Klasik küme, kümeye kesinlikle ait veya kesinlikle ait değil biçiminde iki grubun oluşturulması ile anlamlıdır [2]. Klasik kümede üye olanlarla olmayanlar arasında keskin bir fark vardır. Ama genelde günlük konuşmalarda kullandığımız “uzun boylu”, “orta boylu”, “kısa saçlı” gibi terimler bu karakteristiğe uymaz. Bunların sınırları belirsiz görünür ve üye olandan, üye olmayana geçiş keskin biçimde değildir. Bulanık küme, keskin geçişleri elimine ederek belirsizlik kavramının tanımını yeniden verir ve evrendeki bütün bireylere üyelik derecesi değerini atayarak matematiksel olarak tanımlar. Bulanık küme içerisinde üyelik dereceleri tarafından gösterilen daha büyük veya daha küçük değerlere ait olabilirler.

Bulanık mantık tabanlı bir sistem, üyelik fonksiyonları ve bulanık kurallar sayesinde heuristik bilgileri nümerik değerlere dönüştürür [8]. Bulanık mantık iki büyük bileşenden oluşur: üyelik fonksiyonları ve bulanık kurallar. Kurallar bulanık sistemin davranışını yansıttığından, bulanık kümeler kendi içerisinde öğrenmektedir. Bu bileşenleri kullanarak, mikroişlemci ve otomasyon uygulamaları için elverişli olan niteliksel tanımlamadan nicel tanımlamaya insan deneyimleri ve tercihlerini dönüştürmek olanaklıdır. Bulanık üyelik fonksiyonları, tasarımcının tercihleri ve deneyimlerine bağlı olarak farklı biçim ve şekillerde olabilirler. Diğer taraftan bulanık kurallar EĞER-O ZAMAN ifadelerinin şart-sonuç çiftleri olarak yazılırlar ve tablo biçiminde depolanırlar. Bulanık kuralları elde etmenin 4 yolu vardır:

- Uzman deneyimi ve mühendis bilgisinden
- İnsan operatörlerinin davranışlarından
- Bir sistem veya sürecin bulanık modeline dayanarak türetmelerden
- Öğrenmeye dayalı türetmelerden

Dilsel ifadeler ve bulanık kuralların kullanımı ile karar verme ve kontrol gibi çoğu uygulamalar uzman operatörler olmadan anlaşılması mümkün olabilir. Ek olarak, bulanık mantık karmaşık matematiksel analizler kullanmadan bir karar yüzeyini tam olarak tanımlayacak özelliklere de sahiptir.

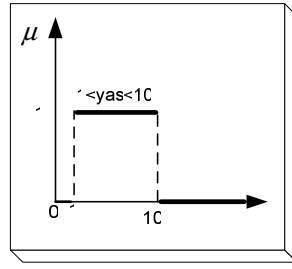
3.3.1. Keskin Kümeler, Bulanık Kümeler ve Üyelik Fonksiyonları

Klasik kümenin karakteristik fonksiyonu, evrensel kümede her bireye 1 veya 0 değerini atar. Bu ise üye olma veya olmama anlamındadır. Bu şekilde tanımlanan küme keskin kümedir çünkü sınırları iyi bir şekilde belirlenmiştir. Bir kümenin elemanlarının gösterilmesi birkaç farklı yolla olabilir:

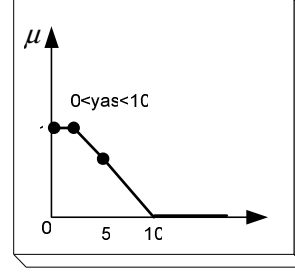
- Liste gösterimi: Örneğin $A_1 = \{\text{Ali, Ahmet, Ayşe}\}$, $A_2 = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$, $A_3 = \{\text{yarasa, kartal, serçe, atmaca}\}$
- Özellik ile gösterim: Örneğin $B_1 = \{x \in \mathbb{N} \mid x < 5\}$, $B_2 = \{y \in \mathbb{Z}^+ \mid x^2 < 100\}$.
- Üyelik fonksiyonu ile gösterim: Bir A kümesi için bu fonksiyon, her $x \in X$ için $\mu_A(X)$ değerini aşağıdaki şekilde belirler.

$$\mu_A(x) = \begin{cases} 1 & x \in A \\ 0 & x \notin A \end{cases} \quad (3.9)$$

Klasik kümede bu fonksiyon, 0 ve 1'lerden oluşan bir küme oluşturur. Bulanık kümeler için $\mu_A(x)$ bir A kümesine bir x elemanının ait olma derecesi $[0,1]$ aralığında $0 \leq \mu_A(x) \leq 1$ ifadesi ile verilir. Şekil 3.6 'da keskin ve bulanık kümeler üyelik fonksiyonu ile gösterilmiştir.



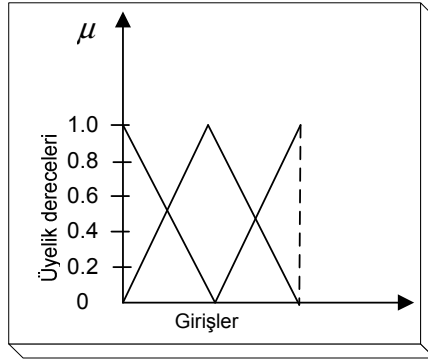
a) Keskin küme için üyelik fonksiyonu



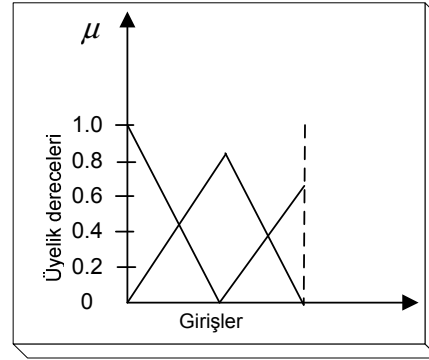
b) Bulanık küme için üyelik fonksiyonu

Şekil 3.6. Keskin ve bulanık kümeler için üyelik fonksiyonu

Şekil 3.6'da, aynı küme üzerinde klasik ve bulanık üyelik fonksiyonları verilmiştir. Klasik küme için (a) seçeneğinde verilen fonksiyonda yaşı bir ile on arasında olanlar için üyelik fonksiyonu 1 değerini almaktadır. Kümenin bütün elemanlarının derecesi bir olup aynıdır. Bulanık küme için (b) seçeneğinde verilen fonksiyonda ise küme elemanları farklı üyelik dereceleri ile kümeye ait olurlar. Örneğin beş yaşındaki bir çocuğun üyelik derecesi 0.5 iken bir yaşındaki birinin 1.0 gibi bir değer olur. Bulanık kümede bir elemanın kümeye ait olma derecesi maksimum 1.0 ve minimum 0'dır. Bu kurala uyan bulanık kümeler normal ve uymayanlar normal olmayan bulanık kümeler olarak isimlendirilir [39]. Bulanık kümenin giriş değişkenleri yani normal bulanık kümeleri ve çıkış değişkenleri ise normal olmayan bulanık kümeleri oluşturur. Şekil 3.7 de bu durumu gösteren bir grafik verilmiştir.



a) Normal bulanık küme



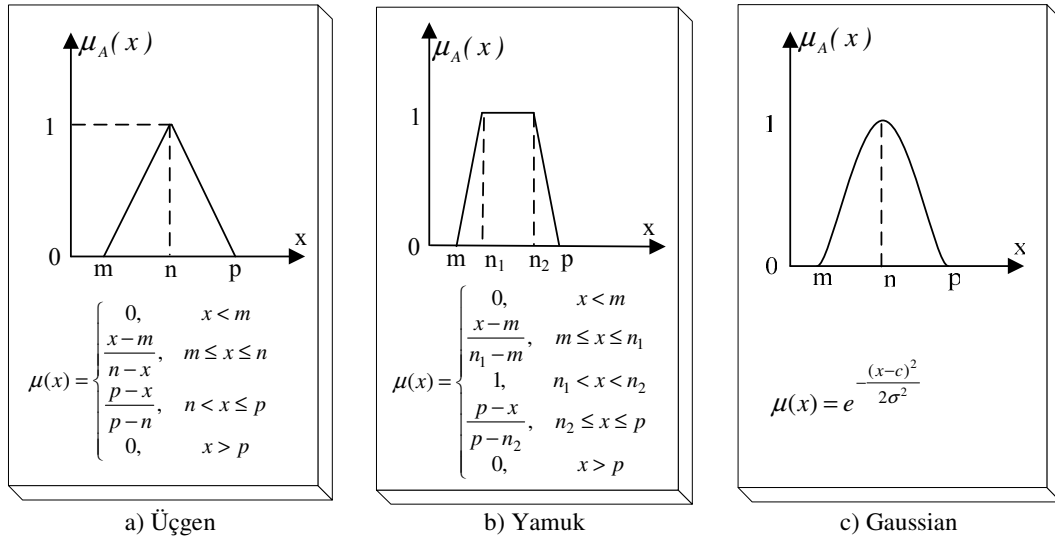
b) Normal olmayan bulanık küme

Şekil 3.7. Normal ve normal olmayan bulanık kümeler için üyelik fonksiyonları

Bulanık mantık, gerçek dünya hakkındaki insan bilgisi ve düşüncesini modelleme için kullanılır. Bulanık kümeler, kümedeki üyeliğin farklı derecelerine sahip elemanlar içerir. Onlar dilsel değişkenleri göstermede çok kullanışlıdır. Bulanık kümeler, doğal veya yapay dil ile tanımlanabilir ve bunların değerleri dilsel (niteliksel) olup nümerik (niceliksel) değildir. Bulanık mantıkta bir eleman aynı zamanda farklı ait olma dereceleri ile iki veya daha fazla kümenin bir

üyesi olabilir. Şekil 3.7 (a)'da her bir eleman aynı anda iki farklı kümeye üyelik dereceleri ile ait olabilir.

Üyelik fonksiyonları, bulanık mantık ile bir problemi çözmek için kullanılan temel bileşenlerden biridir. Bu fonksiyonlar farklı biçimlerde olup, hangi tür bulanık üyelik fonksiyonlarının kullanılacağı probleme göre değişebilir. En gelişmiş yöntemler, uzman tecrübelerinden faydalanarak küme değerlerini noktali olarak belirlemek ve analitik fonksiyon biçiminde tanımlamaktır. Şekil 3.8'de en çok kullanılan üyelik fonksiyonları verilmiştir.



Şekil 3.8. Bulanık üyelik fonksiyonları

3.3.2. Temel Küme İşlemleri

Klasik kümelerde kullanılan küme işlemlerinden çoğu, bulanık kümeler için de kullanılmaktadır. En çok kullanılan küme işlemleri tümleyen, kesişim ve birleşim olup bunlar mantıksal VE, VEYA ve DEĞİL operatörlerine karşılık gelir. A ve B evrensel kümenin iki alt kümesi olsun. A ile B'nin birleşimi $A \cup B$ ile gösterilir ve birleşim kümesinin elemanları A veya B'nin elemanlarından oluşur. Birleşim kümesine bir x elemanın ait olma derecesi $\mu_{A \cup B}(x)$ ile gösterilir matematiksel olarak denklem (3.10)'daki gibi ifade edilir. A ile B'nin kesişimi $A \cap B$ ile gösterilir ve kesişim kümesinin elemanları hem A'nın hem de B'nin elemanlarından oluşur. Birleşim kümesine bir x elemanın ait olma derecesi $\mu_{A \cap B}(x)$ ile gösterilir. Bu işlemin matematiksel gösterimi denklem (3.11)'de verilmiştir. A kümesinin tümleyeni ise $\bar{A}$ ile gösterilir ve A'nın elemanı olmayan bütün elemanları kapsar. Bir x

elemanın A kümesine ait olmama derecesi $\mu_{\bar{A}}(x)$ ile gösterilir ve matematiksel olarak denklem (3.12)'de verilmiştir.

$$\mu_{A \cup B}(x) = \begin{cases} 1, & x \in A \text{ veya } x \in B \\ 0, & \text{diğer} \end{cases} \quad (3.10)$$

$$\mu_{A \cap B}(x) = \begin{cases} 1, & x \in A \text{ ve } x \in B \\ 0, & \text{diğer} \end{cases} \quad (3.11)$$

$$\mu_{\bar{A}}(x) = \begin{cases} 1, & x \notin A \\ 0, & x \in A \end{cases} \quad (3.12)$$

Bulanık mantıkta birleşim, kesişim ve tümleyen operatörleri için maksimum, minimum ve tümleme operatörleri kullanılır. Bu operatörler sırasıyla denklem (3.13), (3.14) ve (3.15)'te verilmiştir.

$$\mu_{A \cup B}(x) = \max[\mu_A(x), \mu_B(x)] \quad (3.13)$$

$$\mu_{A \cap B}(x) = \min[\mu_A(x), \mu_B(x)] \quad (3.14)$$

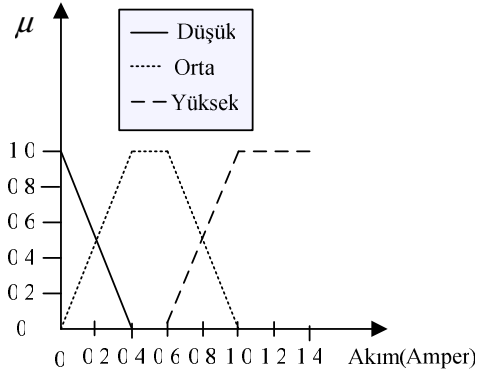
$$\mu_{\bar{A}}(x) = 1 - \mu_A(x) \quad (3.15)$$

Bulanık kesişim ve birleşimi ifade etmek için Zadeh [43] tarafından sunulan bir diğer yaklaşım denklem (3.16) ve (3.17) 'de verilmiştir. Burada birleşim işlemi klasik kümelerdeki birleşim işlemi ile benzerlik gösterirken, kesişim işlemi iki kümenin üyelik derecelerinin cebirsel çarpımına karşılık gelir.

$$\mu_{A \cup B}(x) = \mu_A(x) + \mu_B(x) - \mu_A(x)\mu_B(x) \quad (3.16)$$

$$\mu_{A \cap B}(x) = \mu_A(x)\mu_B(x) \quad (3.17)$$

Şekil 3.8'de, bir motor akımına ait üç dilsel değişkenin üyelik fonksiyonlarını göstermektedir: Düşük, Orta ve Yüksek. Bunlar stator akımını tanımlamak için kullanılabilir. Örneğin, 1.4 Amper akım seviyesi düşük veya orta olarak düşünülemez. Çünkü her iki üyelik fonksiyonunda da sıfır değere sahiptir. Buna karşın, 0.2 Amper 0.5 üyelik derecesi ile Düşük üyelik fonksiyonuna ait iken, 0.5 üyelik derecesi ile de orta üyelik fonksiyonuna aittir.



Şekil 3.9. Örnek bir akım üyelik fonksiyonu

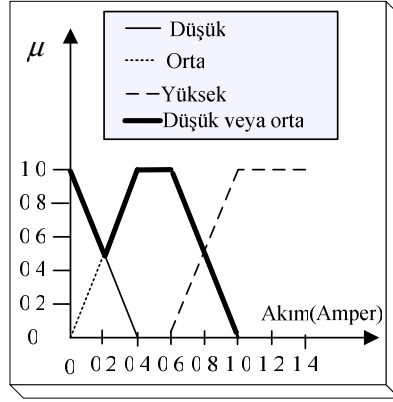
Şekil 3.9'a göre, bütün $x \in X$ A ve B değerleri için μ_A ve μ_B üyelik değerleri ile iki bulanık küme olsun. $[0, 1.4]$ Amper aralığını kapsayan akımı belirleyen Düşük, Orta ve Yüksek için üç üyelik fonksiyonu tanımlayalım.

$$\mu_{Dusuk}(Akım) = 0/1 + 0.2/0.5 + 0.4/0 + 0.6/0 + 0.8/0 + 1.0/0 + 1.2/0 + 1.4/0 \quad (3.18)$$

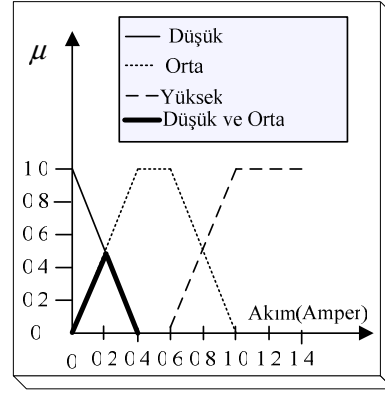
$$\mu_{Orta}(Akım) = 0/0 + 0.2/0 + 0.4/1 + 0.6/1 + 0.8/0.5 + 1.0/0 + 1.2/0 + 1.4/0 \quad (3.19)$$

$$\mu_{Yuksekk}(Akım) = 0/0 + 0.2/0 + 0.4/0 + 0.6/0 + 0.8/0.4 + 1.0/1 + 1.2/1 + 1.4/1 \quad (3.20)$$

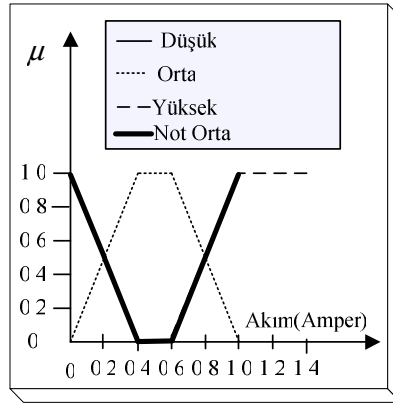
Şekil 3.10'da akım bilgisi için kullanılan üyelik fonksiyonları kullanılarak, bulanık mantıkta küme birleşimi, kesişimi ve tümleyen işlemleri üyelik fonksiyonları üzerinden gösterilmiştir.



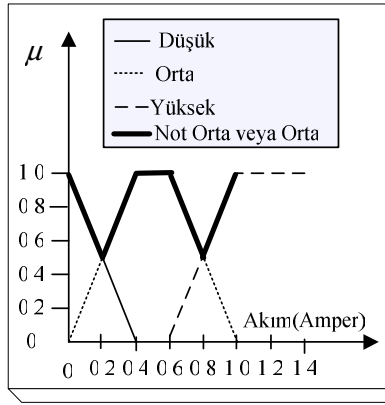
a) Birleşim ($\mu_{Düşük \cup Orta}$)



b) Kesişim ($\mu_{Düşük \cap Orta}$)



c) Tümleneyen ($\mu_{\neg Orta}$)



d) Tümleneyenin kendisi ile birleşimi ($\mu_{Orta \cup \neg Orta}$)

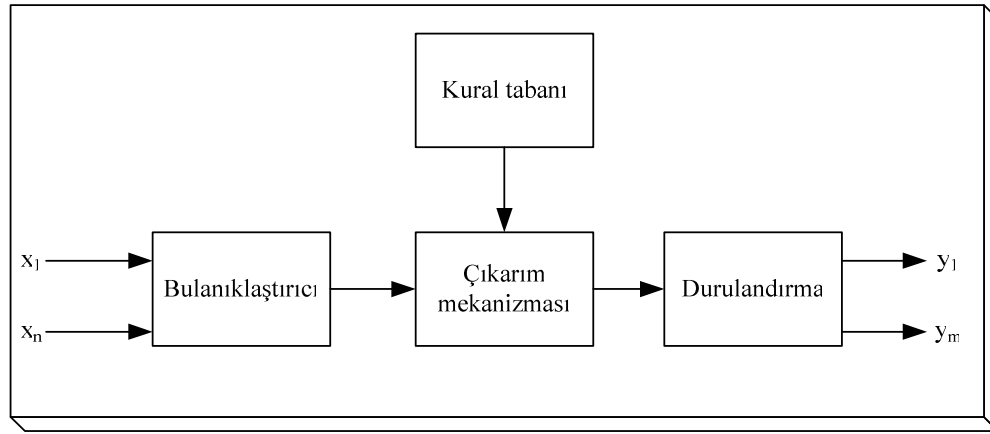
Şekil 3.10. Bulanık mantıkta küme işlemleri

Şekil 3.10 (d)'de, bir kümenin kendi tümleneyeni ile birleşimi verilmiştir. Klasik kümelerde bir kümenin kendi tümleneyeni ile bileşimi birdir. Fakat bulanık kümelerde şekilden de görüldüğü gibi birleşim bire eşit değildir. Klasik kümelerde bir kümenin tümleneyen ile kesişimi sıfır iken bulanık kümelerde sıfırdan farklıdır.

3.3.3. Bulanık Çıkarım Sistemi

Gerçek dünyada, bir sistemin durumu dilsel terimlerden çok genellikle sayısal değerler ile verilir. Bulanık çıkarım sistemi, bulanık kurallar kullanılarak sayısal bir çıkış verisi oluşturmak için giriş verilerinin lineer olmayan eşleşmesi olarak tanımlanır [44]. Bu işlem giriş ve çıkış üyelik fonksiyonları, bulanık mantık operatörleri, bulanık EĞER-O ZAMAN kuralları, çıkış kümelerinin toplamı ve durulandırma işleminden oluşur. Bulanık çıkarım sisteminin genel modeli şekil 3.11'de verilmiştir. Burada $X=\{x_1, x_2, \dots, x_n\}$ bulanık sistemin girişlerini, $Y=\{y_1,$

$y_2, \dots, y_n$ ise bulanık sistemin çıkışlarını gösterir. Kural tabanı, probleme bağlı olarak uzman kişiden elde edilen dilsel ifadelerdir. Bulanıklaştırma, girişleri alarak üyelik fonksiyonları yardımıyla her bir girişin bulanık kümelerine ait olma derecesini belirler [40]. Bazı problemlerde sayısal verilere göre sistemin davranışı gözlemlenerek de kurallar oluşturulabilir. Çıkarım mekanizması, giriş bulanık kümesinden çıkış bulanık kümeyi belirlemek için bulanık kuralları kullanır. Bir bulanık küme, aynı anda birden fazla bulanık kümeye ait olabilir. Bu durumda bulanık mantıksal operatörler kullanılır. Bir veya birden fazla kural aynı anda gerçekleşebilir. Bütün kuralların çıkışları toplanır. Toplama esnasında her bir kuralın çıkışını gösteren bulanık kümeler bir tek bulanık kümeye birleştirilir. Durulandırma işlemi çıkış bulanık kümelerden sayısal bir değer üretir.



Şekil 3.11. Bulanık çıkarım sisteminin blok diyagramı

Uzman bilgisi, EĞER-O ZAMAN kuralları biçiminde dilsel olarak ifade edilebilir. Bir sistemin davranışı hakkında karar verilirken uzman bilgisi genellikle dilsel ifadeler ile verildiğinden, bulanık mantık bilgiyi uygun hareket ve kararlara dönüştürerek problemleri çözer. Bulanık kurallar, şart ve sonuç kısmı olarak iki kısımdan oluşur. Bir bulanık kuralın genel biçimi aşağıdaki gibi tanımlanabilir.

EĞER(Şartların bir kümesi sağlanırsa) O ZAMAN (uygun işlemleri gerçekleştir)

Bir EĞER-O ZAMAN kuralının işletilmesinde, ilk olarak gelen kuralın 0 ile 1 aralığında bir üyelik derecesi belirlenir. Bulanık sistemin birden fazla girişi varsa, bulanık operatörler uygulanarak tek bir çıkış elde edilir. Üç fazlı bir asenkron motorun bir fazı devre dışı kaldığında, diğer iki faz çok fazla akım çekecektir. Arıza teşhisinde böyle bir kural aşağıdaki gibi yazılabilir.

If A is S and B is CB and C is CB then Stator is CK

Bu kuralda A, B ve C bulanık girişleri üç fazlı bir asenkron motorun girişlerini gösterir. Çıkış olarak statorun durumu belirlenir. Burada S sıfır, CB çok büyük ve CK ise Çok kötü üyelik fonksiyonlarını ifade eder. Bulanık sistemin girişinde birden fazla giriş olduğu için, kuralın çıkışını belirlemede minimum veya çarpım operatörleri kullanılır.

Bulanık çıkarım sisteminde, kurallardan sayısal çıkışı belirlemek için durulandırma işlemi uygulanır. Durulandırma sürecinin girişi, bulanık bir küme olup çıkış ise sayısalıdır. Birçok durulandırma yöntemi mevcuttur. Bunlardan en çok kullanılanı ağırlık merkezi yöntem olup denklem (3.21)'de gösterilmiştir.

$$y' = \frac{\sum_{i=1}^n y_i \mu_B(y_i)}{\sum_{i=1}^n \mu_B(y_i)} \quad (3.21)$$

Denklem (3.21)'de y' B'nin ağırlık merkezini belirler ve S $\mu_B(y)$ 'nin destek değeridir. Durulandırma için ağırlık merkezi yönteminin kullanımı ile çıkış bulanık alan ağırlıklı ortalaması hesaplanarak sistem çıkışı belirlenir.

3.4. Bağışık ve Yapay Bağışık Sistemler

3.4.1. Bağışık Sistemler

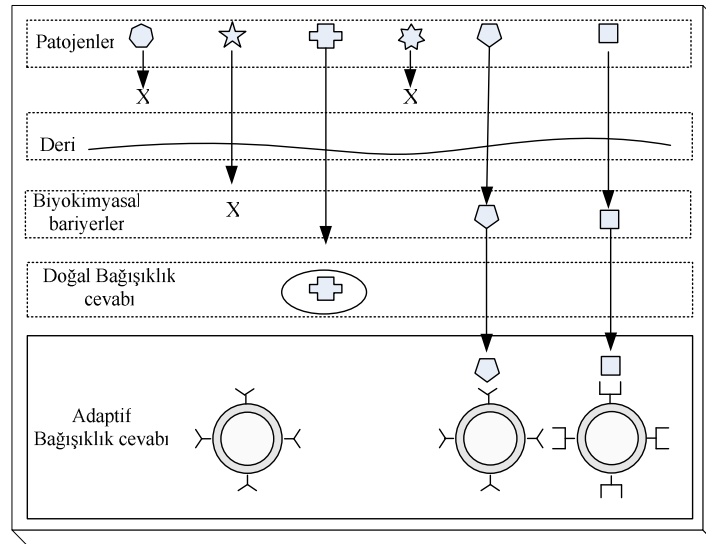
Yaşayan her canlı, kendine has özellikleri ile bir bağışıklık sistemine sahiptir. Bazı bitkiler, yırtıcı hayvanlardan ve herhangi bir tehlikeden korunmak için kendi bağışıklık sistemi olan koruyucu dikenler barındırırlar [9]. Bağışıklık sistemi üzerine yapılan çalışmalar genelde insan bağışıklık sistemine dayanır. Çünkü insan bağışıklık sistemi, yapay bağışıklığın tasarımıındaki geniş uygulanabilirliği, işlevselliği ve ilginç özellikleri ile incelemeye değerdir.

Bağışıklık sistemi en az beyin kadar karmaşık bir yapıya sahiptir. Bağışık sistem organları, diğer vücut hücrelerimiz ile birlikte çalışarak hayati önem taşıyan fonksiyonlarımızın kararlılığını sağlar. Bağışık sistemlerin en önemli görevlerinden biri patojen olarak adlandırılan ve hastalık oluşturan virüslere karşı vücudun korunmasını sağlamasıdır [45]. Virüs, mantar, parazit gibi mikroorganizmalar vücuda zarar veren patojenlerdir. Bağışık sistemlerin patojenleri tanıması için ilk önce kendi vücut hücrelerini tanıması gerekir. Bu yüzden bağışık sistem kendi vücut hücrelerini tanıdıktan sonra herhangi bir yabancı hücreye daha kolay cevap verir. Bir patojen, vücuda girer girmez, belirli hücreler bundan kurtulmak için savaşımaya başlar [46].

Daha önce karşılaştığı bir mikroorganizma ile tekrar karşılaşan bağışık sistem hücreleri, bu mikroorganizmayı tanır ve daha çabuk tepki verir. Örneğin zararlı bir mikroorganizma vücudumuza girdiğinde, buna karşı antikorlar üretilir ve yok edilmeye çalışılır. Aynı mikroorganizma ile tekrar karşılaşıldığında antikorlar onu tanır ve hastalanmamızı önler. Antikor, yabancı maddelere karşı savunma hücrelerinin verdiği yanıttır. Daha önce karşılaştığı vücut hücrelerini tanıma özellikleri ile bağışık sistemler bir hafızaya sahiptir. Aşağıda bağışık sistemlerin özellikleri verilmiştir.

- **Tek olma:** Her bireyin bağışıklık sistemi kendine has özellik ve yeteneklere sahiptir.
- **Kural dışı çalışma tespiti:** Bağışıklık sistemi, vücut hücrelerinin daha önce hiç karşılaşmadığı patojenleri belirleme ve buna tepki gösterme özelliğine sahiptir.
- **Dağıtılmış tespit:** Sistem hücreleri tüm vücuda dağıtılmış olup beyin gibi bir merkezi kontrol sistemi yoktur.
- **Gürültü toleransı:** Bir antikorun patojeni tanıması ve ona tepki göstermesi için tam bir tanıma gerek yoktur.
- **Takviyeli öğrenme ve hafızaya sahip olma:** Bağışıklık sistemi daha önce karşılaştığı patojenler ile tekrar karşılaştığında daha hızlı cevap vermektedir.

Bağışıklık sistemlerin yapısı çok katmanlı olup çeşitli seviyelerde dağılmış bir koruma sistemi vardır. Bu katmanlı yapı şekil 3.12’te verilmiştir.

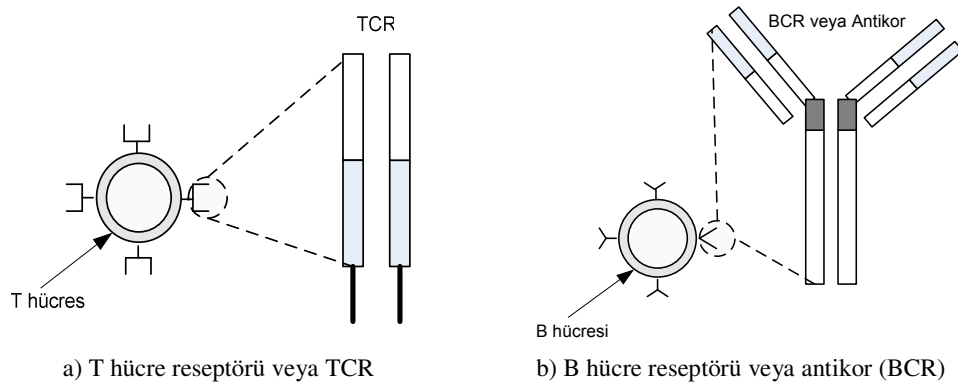


Şekil 3.12. Bağışık sisteminin katmanlı yapısı

Şekil 3.12’de görüldüğü gibi, derimiz patojenlere karşı bağışıklık sistemimizin ilk koruma mekanizmasını oluşturur. Solunum sistemi de yabancı hücrelerin vücuda girmesini engeller. Bademcik ve burun kılları vücuda giren mikroorganizmaları tutarak hapşırma ve öksürme yardımıyla onları dışarı atmamızı sağlar. Ayrıca gözyaşı ve ter gibi sıvıların içeriğindeki tahrip edici enzimler içerdiğinden biyokimyasal bariyerleri oluştururlar. Bütün insanlar belirli mikroplara karşı doğal bir bağışıklık ile donatılmış olarak doğarlar. Bu yüzden doğal bağışıklık, çok sayıdaki patojeni ilk karşılaştığı anda yok edebilmektedir. Adaptif bağışıklık, bir patojen ile daha önceden hiç karşılaşılmamışsa bile, vücuda onu tanımasını sağlayacak bağışıklık kabiliyetinin kazandırılmasıdır [46]. Örneğin Hepatit-B hastalığından korunmak isteyen bir kişi aşı olduğunda bu hastalığa karşı bir adaptif bağışıklık kazanmış olur.

3.4.1.1. Bağışık Sisteminin Fizyolojisi

Bağışık sistem hücrelerinin üretilmesi ve geliştirilmesinden sorumlu iki temel organ; kemik iliği ve timüstür. Kemik iliği, bütün kan hücrelerinin üretilmesi ve geliştirilmesinden sorumludur. Timüs ise T- hücreleri olarak isimlendirilen bağışık sistem hücrelerinin taşınması ve olgunlaşmasını sağlar. Lenfositler, patojenlerin tanınmasını sağlayan beyaz kan hücreleridir. Bağışık sistemlerde, patojenlerin tanınmasını sağlayan temel lenfositler B ve T hücreleridir. B hücreleri kemik iliğinde üretilir. T hücreleri ise timüste olgunlaşır ve burada geliştirilir. Hem B hem de T hücrelerinin yüzeylerinde bulunan ve reseptör olarak algılanan yapılar, vücut hücreleri ve yabancı antijenlerin algılanmasını sağlarlar [45]. Şekil 3.13’te B ve T hücrelerinin yapısı ve yüzey reseptörleri gösterilmiştir.

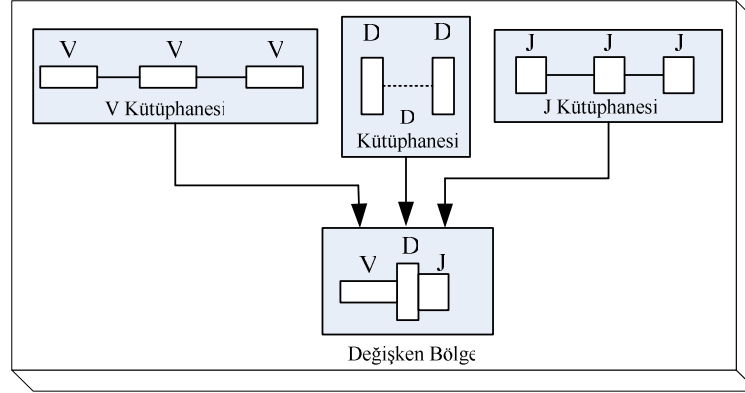


Şekil 3.13. Lenfositler ve yüzey reseptörleri

T hücreleri, yardımcı T hücreleri ve öldürücü T hücreleri olarak iki ana gruba ayrılır. B hücre reseptörleri, Y biçiminde iki temel bölgeden oluşan antikorlara sahiptir. Değişken bölge

antijenlerin tanınmasından sorumluyken, sabit bölge etki fonksiyonlarının uygulanması ve yüzey hücrelerinin antikora bağlanmasından sorumludur.

Antikor molekülleri, çok fazla gene ihtiyaç duyulmadan bir DNA'nın yeniden düzenlenme sürecinden geçerek kemik iliğinde oluşur. Her antikor zinciri, gen kütüphanelerinin yeniden düzenlenip bir araya gelmesi ile oluşur. Şekil 3.14'te kemik iliğinde bir antikorun üretiminin basitleştirilmiş bir biçimi verilmiştir.

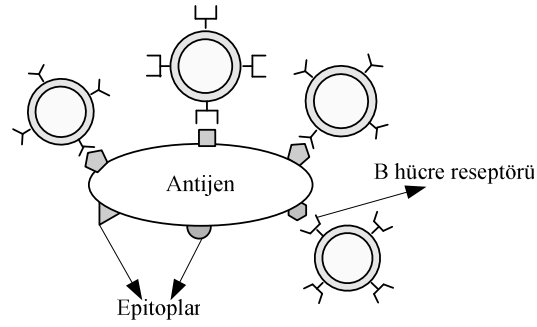


Şekil 3.14. Gen kütüphanesinden bir antikor molekülünün oluşumu

Vücuda giren yabancı hücrelerin tanınması ve öz vücut hücrelerinden ayırt edilmesinde antikorlar önemli rol oynarlar. Antijenlerin çok farklı yapıda olabilmeleri bunları tanıyacak antikorlarında farklı yapıda olmalarını gerektirir. Şekil 3.14'te de görüldüğü gibi bu antikorların farklı yapıda olmalarını sağlayan, V, D ve J genlerinin farklı şekillerde bağlanma özellikleridir. Antikor çeşitliliği ayrıca çok sayıda segmentin varlığı, mutasyon ve antikor modelleri oluşturulurken farklı zincirlerin bir araya gelmesine de bağlıdır.

3.4.1.2. Bağışık Sistemlerde Örüntü Tanıma

Bağışık sistemlerde örüntü tanıma işlemi B ve T hücrelerinin yüzey reseptörleri aracılığıyla moleküler bir seviyede yapılır. T ve B hücrelerinin yüzey reseptörleri, bir antijenin yüzeyinde bulunan ve epitop olarak isimlendirilen hücreler ile tümleyici şekilde bağlanarak antijenlerin tanınmasını sağlarlar [47]. Bir antijen ile B hücresinin bağlanmasına örnek şekil 3.15'te verilmiştir.

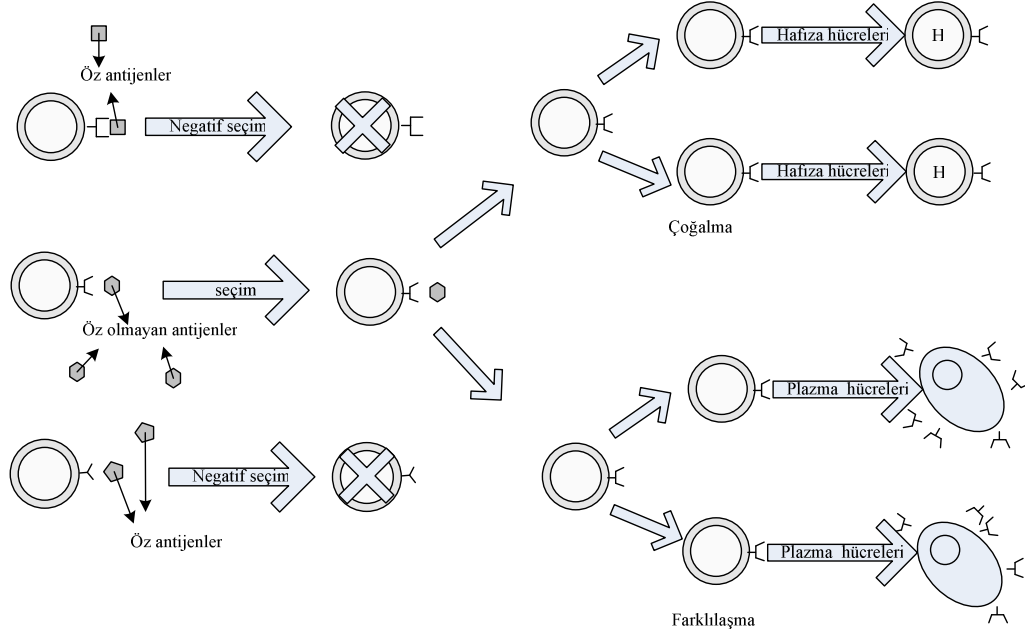


Şekil 3.15. Bir antijenin antikor tarafından tanınması

Şekil 3.15'ten de görüldüğü gibi bir antikor sadece bir tip yüzey reseptörüne sahip olmasına karşın, bir antijen farklı tipte epitoplara sahiptir. Bundan dolayı bir antijen birden fazla antikor tarafından tanınabilir.

3.4.1.3. Klonal Seçim Mekanizması ve Afinite Olgunlaşması

Bağışık hücreler tarafından bir antijen belirlendikten sonra, bir diğer aşama bu antijenleri tanıyabilen hücrelerin çoğaltılması ve diğerlerinin elenmesinin sağlanmasıdır. Bu işlem, klonal seçim olarak adlandırılır. Klonal seçim, hem T hem de B hücreleri için geçerlidir. Bir antijen tespit edildiği zaman, kemik iliği antikor üreterek bağışıklık tepkisi veren hücreleri salgılar. Bu antikorlar antijenlerle birleştiği zaman yardımcı T hücrelerinden gelen sinyallerin de etkisi ile B-hücresini uyarır. B-hücresi uyarma ile birlikte çoğalır ve plazma hücreleri denilen terminal antikor salgılayan hücelere dönüşür. Hücre bölünmesi süreci bir klon oluşturur. Bu klon bir tek yavru hücre veya bir küme yavrudur. Beyaz plazma hücreleri en aktif antikor salgılayıcılarıdır. Ayrıca, hızlı şekilde bölünen büyük B lenfositleri de daha düşük bir oranda antikor salgırlar [9]. Lenfositlerin çoğalma ve farklılaşma dışında diğer önemli bir özelliği ise hafıza hücrelerine dönüşebilmeleridir. Hafıza hücreleri, daha sonra bir antijenik durum ile karşılaşıldığında daha hızlı cevap verilmesini sağlar. Bir antijenin tam tanınması için hücre çoğalması ve farklılaşma şekil 3. 16'da gösterilmiştir.



Şekil 3.16. Klonal seçim

Şekil 3.16'dan da görüldüğü gibi öz olmayan hücreleri tanıyan bağışık sistem hücreleri, çoğalma ve farklılaşma evrelerinden geçerek hafıza hücrelerini oluştururlar. Böylece, öz olmayan antijenleri tanıyan hücreler daha sonra karşılaştıkları bu antijenleri tanıma kabiliyetine sahip olacaklardır.

Afinite olgunlaşması, bir antijen ile daha iyi eşleşen farklı evlatların seçimi ve antijen ile bağlanan reseptörleri etkileyen mutasyon sürecinin birleşimidir. Afinite, antijen ile hücre reseptörünün bağlanma derecesine göre çalışır. Yüksek afiniteye sahip bir hücre, yabancı hücrenin daha iyi tanınmasını ve daha hızlı bir bağışık cevap verilmesini sağlar. Bağışık sistemin hücreleri, somatik hücreler olduğundan buradaki mutasyon da somatik mutasyon olarak adlandırılır. Somatik mutasyon, hücre afinitesiyle ters orantılı olup düşük afiniteli hücrelere daha yüksek mutasyon uygulanmasını gerektirir [48]. Somatik mutasyon ile ters orantılı olarak bir hücrenin çoğalma oranı da hücrenin antijen ile bağlanma derecesi ile orantılıdır. Bir yabancı hücre vücuda girdiğinde birçok lenfosit farklı afinite dereceleriyle yabancı hücreyi tanımaya çalışırlar. Yüksek bağlanma derecesine sahip hücrelerin daha çok kopyası oluşturur. Klonal seçim prensibi hem B hem de T hücrelerini etkilemesine rağmen, afinite olgunlaşması sadece B hücrelerini etkiler [45].

3.4.1.4. Öz/ Öz Olmayan Ayrımı

Bağıışıklık sistemde, sadece yabancı hücrelerin değıl aynı zamanda vücudun kendi hücrelerinin de tanınması gerekir. Kendi vücut hücrelerimiz öz hücreler, vücuda dışardan giren hastalığa sebep olabilecek yabancı hücreler ise öz olmayan hücrelerdir. Bağıışıklık sistemi öz hücreleri tanımazsa, bu hücrelere karşı da bir koruma mekanizması devreye girip vücut hücrelerine de zarar verilecektir. T hücreleri kemik iliğinde üretilip timüste olgunlaşırlar. Öz T hücrelerinin yabancı hücrelerden ayırt edilmesi için timüste negatif seçim işlemi uygulanır [46]. Reseptörleri öz antijenler ile bağlanan ve onları tanıyan timüsteki T hücreleri, negatif seçim işlemi ile elenir ve timüs dışına öz antijenleri tanımayan hücreler verilir. Böylece timüs içinde sadece öz antijenler bulunup, öz olmayan antijenler kan akışına verilir.

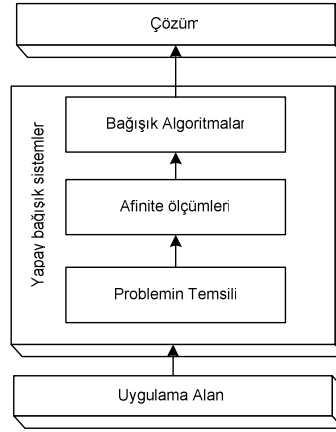
3.4.2. Yapay Bağıışık Sistemler

Yapay bağıışık sistemler, genetik algoritma ve yapay sinir ağıları gibi insan vücudundaki bazı biyolojik sistemlerin çalışma prensibinin gerçek dünyada uygulanmış şekli olup birçok alanda uygulanır. Yapay bağıışık sistemler, bilim ve mühendislik alanındaki problemleri çözmek için uygulanan prensipler, bağıışık fonksiyonlar ve teorik bağıışıklıktan esinlenilen hesaplama sistemleri olarak tanımlanır [9]. Teorik bağıışıklık üzerine yapılan çalışmalar genellikle laboratuvar deneyleri ve bağıışıklık sistemlerin çalışmasının daha iyi anlaşılmasını sağlarken, yapay bağıışık sistemler alanındaki çalışmalar ise mühendislik, bilim ve diğler araştırma alanlarındaki problemlere uygulanır. Bu özelliğı ile yapay bağıışık sistemler yumuşak hesaplama alanı olarak düşünülür.

Yapay bağıışık sistemlerin ilk ortaya çıkması, bilgisayar virüslerinin tespit edilmesi üzerine yapılan çalışmalardan dolayıdır. Bilgisayarlardaki virüs terimini ilk kez 1986'da kullanan Fred Cohen bu terimi "bilgisayardaki programlara kendini değıştirerek giren ve onlara zarar veren programlar" olarak tanımlamıştı. Bu virüslerin bilgisayara verdiği zarar, insan vücuduna giren patojenlerin davranışları ile benzerlik gösterir. Yapay bağıışık sistemler alanındaki çalışmaların başlangıcı daha eskiye dayamasına rağmen, ciddi olarak ilk kez 1996'da düzenlenen "Bağıışık Tabanlı Sistemler" isimli uluslar arası konferans ile başladı. Daha sonraki aşamalarda bazı araştırmacılar bağıışık sistemlerin modellerini çıkararak çalışma şeklinin daha iyi anlaşılmasını sağladı. Bağıışık sistemlerin birçok özelliğı, bu alandaki araştırmacılara yön vermiştir. Bağıışık sistemlerdeki örüntü tanıma, hafıza özellikleri ile daha önceden karşılaştıkları durumları hatırlayabilme ve sınıflandırma gibi özelliklerinden esinlenilerek geliştirilen yapay

bağışık sistemler ekonomi, dolandırıcılık tespiti, bilgisayar güvenliği gibi birçok alanda başarılı bir şekilde uygulanmıştır.

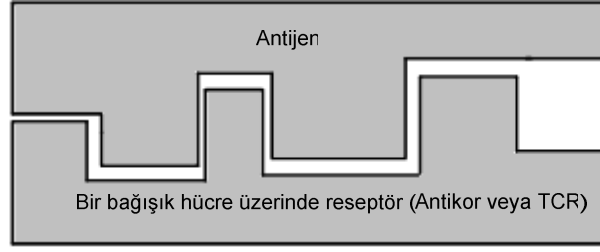
Yapay bağışık sistemler katmanlı bir yapı olarak düşünülebilir. Bu yapı şekil 3.17’de verilmiştir. Her sistemin temel adımı uygulama alanıdır. Problem uygun bir şekilde temsil edildikten sonra, ortam ile bireylerin etkileşimini sağlayacak afinite ölçümlerinin hesaplanması gerekir. Öklid ve hamming ölçümleri gibi farklı afinite ölçümleri mevcuttur. Son katman ise sistem dinamikleri ve davranışlarını yöneten algoritma ve işlemleri içerir. Bunlar bağışık ağ sistemleri, negatif seçim ve klonal seçim algoritmalarıdır.



Şekil 3.17. Yapay bağışık sistemlerin katmanlı yapısı

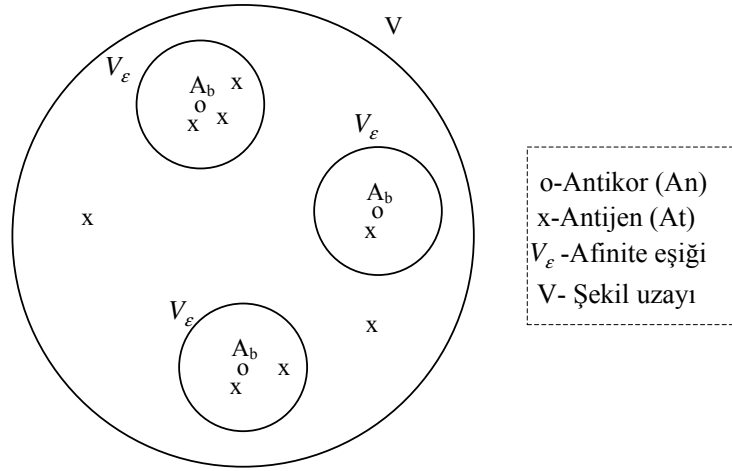
B ve T hücreleri, patojenleri tanıma ve öz hücreleri onlardan ayırt etmeyi sağlayan en önemli bağışıklık sistem hücreleridir. Bu iki hücre, yüzeylerinde bulunan reseptörler ve epitop olarak isimlendirilen antijen reseptörleri ile tümleyen biçiminde bağlanarak onları tanır ve onlara uygulanacak etki fonksiyonunu harekete geçirirler. Bu hücreler ve moleküller, yapay bağışık sistemleri oluşturmak ve modellemek için kullanılan elemanlardır.

Yapay bağışık sistemlerde şekil uzayı kavramını ilk kez 1979’da Perelson ve Oster [49] önermiştir. Önerdikleri yaklaşıma göre, antijenleri tanıyan hücrelerin yüzey reseptörleri ile antijenlerin bağlanması bir şekil uzayı ile gösterilebilir. Bağışıklık sistemlerde olduğu gibi şekil uzayında bir antijen ile reseptörün bağlanma derecesi tümleyen alanlar ile hesaplanır. Bu şekil 3.18’de gösterilmiştir.



Şekil 3.18. Tümleyen alanlar aracılığıyla tanıma

Bir antijen ile antikoru bağlanması, şekil 3.18’de de görüldüğü gibi bir anahtar kilit mekanizmasına benzer. Fakat bir anahtarın kapıyı açması için tam uyması gerekir. Buna karşın bu sistemde bir hata toleransı vardır. Bu şeklin daha genelleştirilmiş bir biçimi olan biçim uzayında N noktalı bir popülasyon sonlu bir V uzayında N adet nokta anlamına gelir [50]. Şekil 3.19’da bu biçim uzayının temsili bir gösterimi verilmiştir.



Şekil 3.19. Şekil uzayının gösterimi

Şekil 3.19’daki şekil uzayına göre, her antikor kendisine belirli bir V_ϵ gibi bir uzaklıkta olan antijenler ile bağlanır ve onları tanır. Aynı şekil uzayı içerisinde kalan antijenler aynı antikorlar tarafından tanınır.

Antikor ve antijenleri göstermek amacıyla vektörler kullanılabilir.

$$A_n = \langle A_{n1}, A_{n2}, \dots, A_{nL} \rangle \quad (3.22)$$

$$A_t = \langle A_{t1}, A_{t2}, \dots, A_{tL} \rangle \quad (3.23)$$

Şekil uzayları gerçek değerli, tam sayı, Hamming ve sembolik şekil uzayları olarak ifade edilir. Hücreler arasındaki etkileşim afinite olarak isimlendirilir. Afinite ölçümleri, çeşitli uzaklık ölçümlerine bağlıdır. Denklem (3.24)'te bir antijen ile antikor arasındaki öklid şekil uzaklığının denklemi verilmiştir.

$$M = \sqrt{\sum_{i=1}^L (A_{n_i} - A_{t_i})^2} \quad (3.24)$$

Gen kütüphanesi, kemik iliğinden antikorları üretmek için kullanılır. Bu özellik, reseptörleri gösteren dizgeleri oluşturmak için kullanılabilir. Antikorlar gen kütüphanelerinin her birinden rasgele seçilen farklı gen parçalarının etkileşiminden oluşur.

3.4.2.1. Negatif Seçim Algoritması

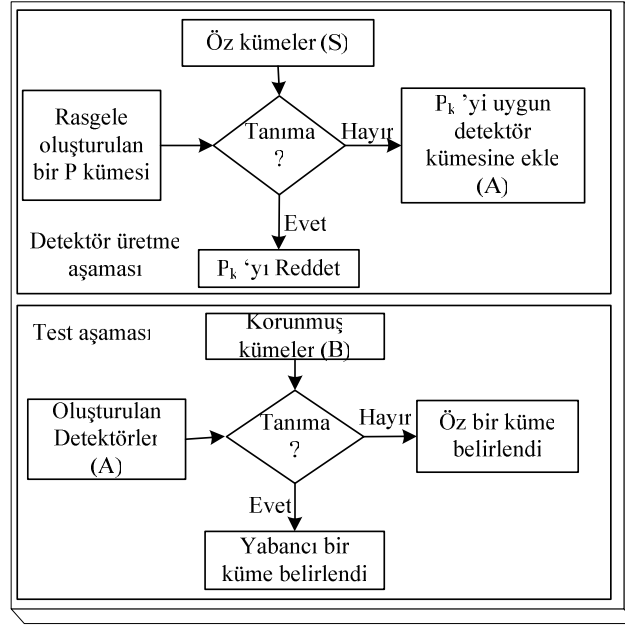
Kemik iliğinde üretilen T hücrelerinin gelişimi ve olgunlaşmasından timüs sorumludur. Timüs, öz olmayan yabancı antijenleri timüsün dışında tutan bir kan bariyeri ile korunur [9]. Bundan dolayı timüs içerisinde bulunan hücreler vücudun öz hücreleridir. T hücrelerinin timüs içinde olgunlaşması esnasında, buradaki öz antijenleri tanıyan T hücreleri elenir ve böylece timüs dışına verilen hücreler öz olmayan hücreleri tanıma kabiliyetine sahip olurlar. Negatif seçimin bu özelliği, mühendislik alanında kullanıldığında, bir sistemdeki farklı durumlar ve kural dışı çalışmaları tespit etmek amacıyla kullanılabilir.

Timüsteki T hücrelerinin negatif seçim mekanizmasından esinlenen Forrest ve diğ.[51], bu yöntemi bilgisayar virüslerini tespit etmek için uyarlamışlardır. Bu algoritmanın en önemli özelliği, sistem tarafından bilinmeyen örüntüler hakkında bilgi depolama yoluyla bu örüntülerin tanınmasının sağlanmasıdır.

Negatif seçim algoritması iki aşamadan oluşur. Birinci aşamada öz olmayan hücreleri tanıyan detektörler üretilir. İkinci aşama izleme aşaması olup, bir veri kümesi verilerek algoritmanın doğruluğu test edilir. Şekil 3. 20'de negatif seçim algoritması verilmiştir. Aşağıda algoritmanın çalışma adımları basit olarak verilmiştir.

1. Başlangıç: Rasgele bir dizge kümesi oluşturarak P kümesine yerleştir.
2. Afinite değerlendirme: P kümesi içerisindeki her bir dizge ile öz kümedeki dizgelerin afinitesini belirle.
3. Detektör üretimi veya silme: P kümesi içerisindeki bir dizge ile S kümesinde en az bir dizgenin afinitesi ile belirlenen afinite eşiğine eşit veya büyük ise bu dizge öz hücreleri

tanır ve elenmesi gerekir. Aksi taktirde P 'deki dizge öz olmayan bir kümeye ait olup detektör kümesine eklenir.



Şekil 3.20. Negatif seçim algoritması

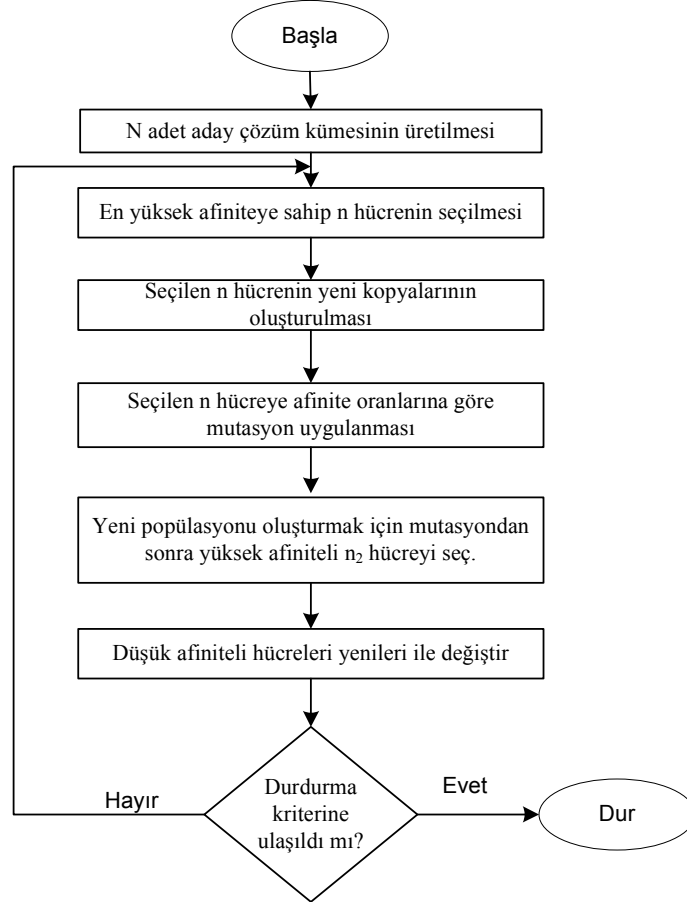
Şekil 3.21'de, eğitim aşamasında uygun detektörler üretildikten sonra test aşamasında bir korunmuş dizge ile en az bir detektör arasında bir tanıma varsa öz olmayan bir örüntü kümesi belirlenir. Tanıma oluşmazsa öz bir örüntü belirlenir.

3.4.2.2. Klonal Seçim Algoritması

Klonal seçim, vücuda yabancı bir hücre girdiğinde bağışıklık sisteminin davranışını tanımlamak amacıyla kullanılır. Klonal seçim algoritmasına göre sadece yabancı antijenleri tanıyan hücreler çoğalmakta ve diğerleri elenmektedir. Bir hücre ile antijenin bağlanma derecesini arttırmak için afinite olgunlaşması işlemi uygulanır. Normal bağışıklık sisteminde vücut bir antijen ile defalarca karşılaşabilir. İlk aşamada antijene verilen tepki ile daha sonraki aşamalarda verilen tepki aynı değildir. Afinite olgunlaşması ve mutasyon işlemleri ile öz hücreler antijenleri daha iyi tanıyacak bir yapıya kavuşurlar ve daha hızlı cevap verirler. Bu özellik takviyeli öğrenme stratejisi olarak bilinir [52].

Klonal seçim algoritması makine öğrenmesi, örüntü tanıma ve optimizasyon gibi alanlarda başarılı bir şekilde uygulanmaktadır. De Castro ve Zoben [52] klonal seçim

algoritmasını optimizasyon, basit bir ikili karakter tanıma ve gezgin satıcı problemini çözmek için kullanmışlardır. Klonal seçimde, antijenik afinite ile orantılı olarak en çok uyarılan hücreler seçilir ve kopyaları oluşturulur. Hiç uyarılmamış hücreler silinir. Yüksek afiniteye sahip hücreler, afinite olgunlaşması ile tekrar seçilir. Ayrıca popülasyonun farklılaşması da sağlanarak sonraki adımlarda çözüm kümesi genişletilir. Şekil 3.21 'de klonal seçim algoritmasının çalışma prensibinin temel adımları akış şeması olarak verilmiştir.



Şekil 3.21. Klonal seçim algoritmasının temel adımları

Klonal seçim algoritmasında hücrelerin kopyalama sayısı afinite ile doğru orantılıyken, mutasyon ile ters orantılıdır. Bu özellik sayesinde, yüksek afiniteye sahip hücrelerin popülasyonda temsilinin arttırılması ve düşük afiniteli hücrelere daha yüksek mutasyon uygulanarak afinitelerinin değişimi sağlanır. Klonal seçim algoritması, genetik algoritma ile benzerlik gösterir. Fakat temel bazı farklılıklar vardır. Klonal seçim algoritmasında sadece seçim değil aynı zamanda mutasyon da afinite orantılı olarak uygulanır. Ayrıca genetik algorithmada bulunan çaprazlama işlemi klonal seçimde yoktur.

4. VERİ MADENCİLİĞİ

4.1. Giriş

Bilgi teknolojilerindeki gelişmeler, bilgisayarların ve otomatik veri toplama araçlarının geniş bir alanda uygulamasını sağlamıştır. Yaygın bilgisayar kullanımı sonucunda, farklı ortamlarda çok büyük miktarlarda veriler birikmiştir. Bu verilerin büyük bir çoğunluğu bir daha kullanılmamak üzere veritabanlarında tutulurlar. Veri üzerinde hesaplama yaparak elde edilen sonuçların değerlendirildiği hesaplama araçlarının gelişimi, veri toplamaya göre oldukça yavaştır. İlgilenilecek veriler çok büyük boyutta olduğundan, geliştirilen hesaplama araçlarının hız açısından da performanslı olması beklenir. Astronomi veritabanlarında tutanak sayısı 10^{12} 'lere ulaşırken, sağlık sektöründeki uygulamalarda öznitelik sayısı 10^2 ile 10^3 arasında değişmektedir [53]. Veritabanındaki varlıkların tanımı ve aralarındaki ilişkilerin tutulduğu kısım veritabanının içsel boyutu olarak bilinir. İşletimsel kayıtlardan yola çıkarak tanımlanmış öz niteliklerin yeniden bilgi keşfi açısından düzenlenmesi de günümüz veritabanı teknolojilerinin önündeki en büyük sorunlardan biridir.

Veri miktarındaki artışın en önemli sebebi, elektronik veri toplayıcılarının kullanımının yaygınlaşmasıdır. Öyle ki artık günlük alışverişlerimiz, alışverişimiz sırasında aldığımız ürün çeşitleri, işyerine giriş çıkış saatlerimiz gibi çoğumuzun farkında olmadan yaptığı günlük işlemler büyük veritabanlarına kaydedilen verilerdir. Sadece uydu ve diğer uzay araçlarından dünyaya gönderilen görüntülerin saatte 50 gigabyte üzerinde olduğu düşünülürse, bu artışın boyutlarını daha açık bir şekilde göstermektedir. Bu kadar büyük veri yığınları arasında insan gözü mevcut verilerin ancak çok küçük bir kısmını görebilecek ve yorumlayabilecektir. Veri tabanı sistemlerinin kullanımının artışı ve hacimlerindeki muazzam büyüme ile birlikte bu verilerden nasıl faydalanabileceğimiz problemi ile karşılaşmıştır. Mevcut veri sistemlerinde kullanılan sorgu veya raporlama araçlarının söz konusu veri yığınları karşısında yetersiz kalması, veri tabanlarında bilgi keşfi adı altında yeni veri analiz yöntemleri arayışlarına neden olmuştur [31]. “Veritabanlarında Bilgi Keşfi” konferansları düzenlenerek veri tabanlarındaki uygulamaların akademik çevrelerce de etkin bir şekilde ele alınması hedeflenmiştir. Bu alandaki araştırmalar ile birlikte veri madenciliği sistemleri ortaya çıktı. Öyle ki veri madenciliği sistemleri bu konuda kullanılan birçok yöntemden faydalanmakta hatta bunların birkaçını bir arada kullanabilmektedir.

Veri madenciliği, doğal bilgi teknolojisindeki gelişmelerin bir sonucu olarak görülebilir. Bu gelişmeler veri toplama, veritabanı oluşturma, veri yönetimi ve verinin analiz edilerek

anlaşılması adımlarını kapsar. Basit olarak veri madenciliği, çok büyük miktardaki veriden bilgi çıkarılması olarak ifade edilir. Bu amaçla veri madenciliği teknikleri yapay zekâ, makine öğrenmesi, örüntü tanıma ve istatistik gibi teknikleri kullanır. Veri madenciliği veritabanlarında bilgi keşfi sürecinin temel adımını oluşturur [30].

Veri madenciliği finans, perakende satış işlemleri, biyomedikal veriler, zaman serileri, telekomünikasyon ve astronomi gibi birçok alanda uygulanmaktadır. Aşağıda bu uygulamalara bazı örnekler verilmiştir.

Perakende Satışlarda

- Satış tahmini
- Müşteri ürün ilişkisinin kurulması
- Ürün kataloglarının hazırlanması
- Reyon düzenlemeleri
- Müşteri satın alma alışkanlıklarının çıkartılması

Finans Alanında

- Kredi dolandırıcılıklarının tespiti
- Gelir düzeyine göre kişilerin kredi miktarlarının belirlenmesi

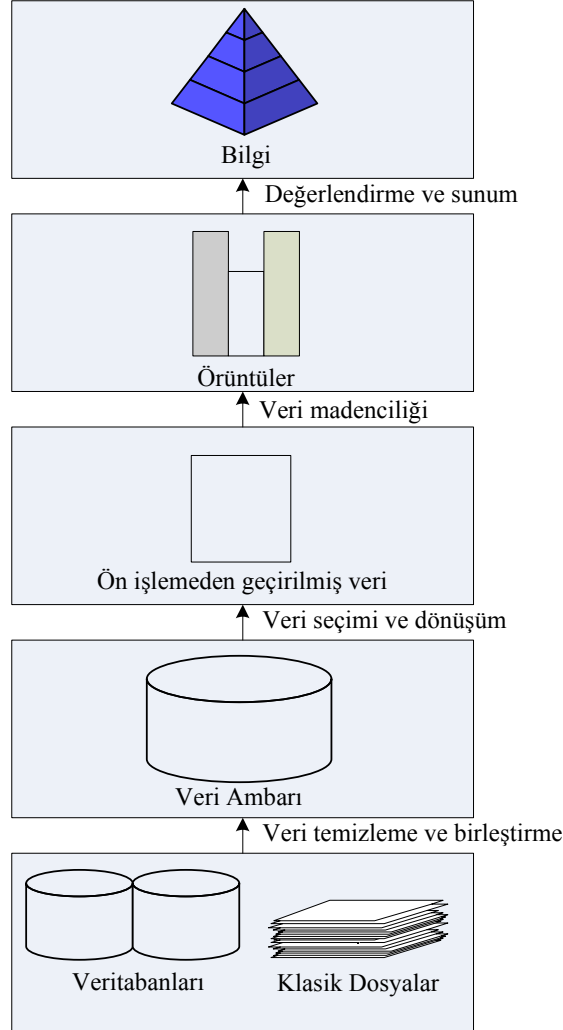
Zaman Serileri Alanında

- Yıllara göre ürün satışlarındaki benzerliklerin bulunması
- Zaman üzerinde oluşan depremlerin tahmini
- Ürün fiyatlarındaki keskin düşüş veya yükselmelerin belirlenmesi

4.2. Bilgi Elde Etme Süreci ve Veri Ön İşleme Adımları

Büyük boyutlardaki verilerden bilgi elde etme süreci olarak bilinen veritabanlarında bilgi keşfi, büyük miktardaki verileri işleyerek bu verilerden ilginç örüntüler çıkarır [54]. Veritabanlarında bilgi keşfi yapılırken şekil 4.1 'de verilen adımlar izlenir. Bu işlemlerden veri temizleme çelişkili, gereksiz, gürültülü ve tam olmayan verinin silinmesini sağlar. Böylece sorguların daha hızlı çalışması sağlanır. Örneğin hamilelik üzerine yapılan bir çalışmada hasta cinsiyetinin alınması gerekmediğinden, bu bilgi silinmelidir. Veri birleştirme adımı veri ambarları, veri küpleri ve düz dosyalar gibi veri kaynaklarındaki verilerin birleştirilmesi sağlanır. Veri seçimi, veritabanından sadece analizi yapılacak alan ile ilgili verinin alınmasıdır. Birçok veri tabanı temelde birbirleriyle ilgisi olan verilerden oluşmasına rağmen özelden gerekli verilerin belirlenerek seçimlerinin yapılması gerekmektedir. Örneğin kirada oturanlar derken, seçim için kirada oturma şartını öne sürmemiz bir kriterdir. Veri dönüşümü, probleme uygun olarak eldeki verinin kullanılması için normalleştirme, toplama ve genelleştirme gibi işlemlerin

uygulanmasıdır. Veri madenciliği, veri örüntülerini çıkarmak için uygulanan akıllı yöntemleri içerir. Büyük bir veritabanından en çok birlikte alınan ürün örüntülerinin bulunması veri madenciliğine bir örnek olarak verilebilir. Sistem tarafından tanımlanan örüntüler, insanların karar vermesine yardımcı olur. Ayrıca bu örüntüler verilerin sınıflandırılmasında, veritabanı içeriğinin özetlenmesinde veya gözlenen olayların açıklamasında daha sonra kullanılabilecek bilgiler şekline dönüştürülür.



Şekil 4.1. Bilgi keşfi sürecinde bir adım olarak veri madenciliği

4.3. Veri Madenciliğinde Karşılaşılan Problemler

Veri madenciliği için kullanılan makine öğrenmesi gibi algoritmalar küçük veri kümelerinde hızlı ve doğru bir biçimde çalışırken, çok büyük veritabanlarına uygulandığında

tamamen farklı davranabilir. Bir veri madenciliği sistemi tutarlı veri üzerinde mükemmel çalışırken, aynı veriye gürültü eklendiğinde kayda değer bir biçimde kötüleşebilir. Aşağıda veri madenciliği algoritmalarında en çok karşılaşılan problemler verilmiştir.

4.3.1. Veritabanı Boyutu

Günümüzde kullanılan veritabanı boyutları hızla arttığından, bu veritabanları üzerinde çalıştırılacak makine öğrenmesi algoritmaları çok yavaş çalışmaktadır. Küçük veri kümeleri üzerinde hızlı çalışan algoritmalar yüz binlerce hatta milyonlarca veri üzerinde uygulandığında yetersiz sonuçlar verirler [31]. Örneklemin büyük olması, örüntülerin gerçekten var olduğunu göstermesi açısından bir avantajdır ancak böyle bir örneklemden elde edilebilecek olası örüntü sayısı çok büyüktür. Bu yüzden veri madenciliği sistemlerinin karşı karşıya olduğu en önemli sorunlardan biri veritabanı boyutunun çok büyük olmasıdır. Dolayısıyla veri madenciliği yöntemleri ya sezgisel bir yaklaşımla arama uzayını taramalıdır ya da örneklemini yatay/dikey olarak indirgemelidir.

Yatay indirgeme, nitelik değerlerinin önceden belirlenmiş genelleme sıradüzenine göre, bir üst nitelik değeri ile değiştirilme işlemi yapıldıktan sonra aynı olan satırların çıkarılması işlemidir. Dikey indirgeme, artık niteliklerin indirgenmesi işlemidir. Özellik seçimi yöntemleri ya da nitelik bağımlılık çizelgesi uygulanarak yapılır.

4.3.2. Gürültülü Veri

Büyük veritabanlarında pek çok niteliğin değeri yanlış olabilir. Bu hata, veri girişi sırasında yapılan insan hataları veya girilen değerlerin yanlış ölçülmesinden kaynaklanır. Veri girişi ya da veri toplanması sırasında oluşan sistem dışı hatalara gürültü adı verilir [30]. Ancak günümüzde kullanılan ticari ilişkisel veritabanları, veri girişi sırasında oluşan hataları otomatik biçimde gidermek konusunda az bir destek sağlamaktadır. Hatalı veri, gerçek dünya veritabanlarında ciddi problem oluşturabilir. Bu durum, bir veri madenciliği yönteminin kullanılan veri kümesinde bulunan gürültülü verilere karşı daha az duyarlı olmasını gerektirir. Eğer veri kümesi gürültülü ise sistem bozuk veriyi tanımalı ve ihmal etmelidir.

4.3.3. Geçersiz Değerler

Veritabanlarında geçersiz veri, değeri birincil anahtarda yer almayan herhangi bir niteliğin değeri olabilir. Bir kayıta eğer bir nitelik değeri geçersiz ise o nitelik bilinmeyen ve uygulanamaz bir değere sahiptir [54]. Bu durum, ilişkisel veritabanlarında sıkça karşımıza

çıkılmaktadır. Bir ilişkide yer alan tüm kayıtlar aynı sayıda niteliğe, niteliğin değeri geçersiz olsa bile, sahip olmalıdır. Örneğin kişisel bilgisayarların özelliklerini tutan bir ilişkide bazı model bilgisayarlar için ses kartı modeli niteliğinin değeri geçersiz olabilir.

4.3.4. Eksik Veri

Evrendeki her nesnenin, ayrıntılı bir biçimde tanımlandığını ve bu nesnelerin alabileceği değerler kümesinin belirli olduğu varsayalım. Eğer her bir nesnenin tanımı kesin ve yeterli olsa idi, sınıflama işlemi basitçe nesnelerin altkümelerinden faydalanılarak yapılardı. Bununla birlikte veriler kurum ihtiyaçları göz önünde bulundurularak düzenlenip, toplandığından, mevcut veri gerçek hayatı yeterince yansıtmayabilir. Örneğin, hastalığın tanısını koymak için kurallar sadece çok yaşlı insanların belirtilerinin bulunduğu bir veri kümesi kullanılarak üretilseydi, bu kurallara dayanarak bir çocuğa tanı koymak pek doğru olmazdı. Bu gibi koşullarda bilgi keşif modeli belirli bir güvenlik derecesinde tahmini kararlar alabilmelidir.

4.3.5. Artık Veri

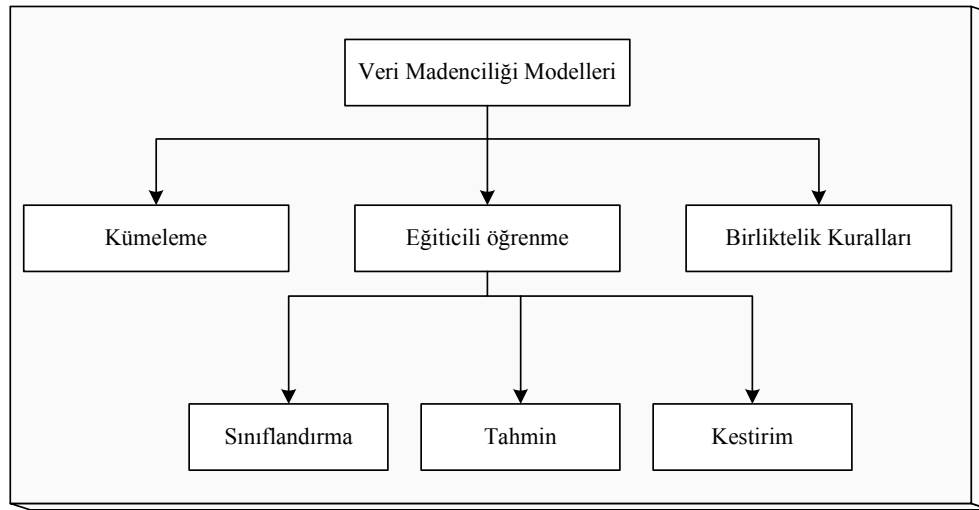
Verilen veri kümesi, eldeki probleme uygun olmayan artık nitelikler içerebilir. Bu durum pek çok işlem sırasında karşımıza çıkabilir. Örneğin eldeki problem ile ilgili veriyi elde etmek için iki ilişkiyi birleştirecek elde edilen ilişkide kullanıcının farkında olmadığı artık nitelikler bulunur. Artık nitelikleri elemek için geliştirilmiş algoritmalar özellik seçimi olarak adlandırılır.

4.3.6. Dinamik Veri

Veri madenciliği algoritmaları genellikle sabit veriler üzerinde çalışır. Fakat günümüzde veritabanlarına sürekli bilgi eklenmekte, dolayısıyla veri boyutu sürekli artmaktadır. Bu durum veri madenciliği algoritmalarının performansını ciddi biçimde kötüleştirir. İlk olarak sadece okuma yapan ve uzun süre çalışan bilgi keşfi metodu bir veritabanı uygulaması olarak mevcut veri tabanı ile birlikte çalıştırıldığında mevcut uygulamanın da performansı ciddi ölçüde düşer [30]. Diğer bir sakınca ise, veritabanında bulunan verilerin kalıcı olduğu varsayılıp, çevrimdışı veri üzerinde bilgi keşif metodu çalıştırıldığında, değişen verinin elde edilen örüntülere yansması gerekmektedir. Bu işlem, bilgi keşfi metodunun ürettiği örüntüleri zaman içinde değişen veriye göre sadece ilgili örüntüleri yığılmalı olarak günleme yeteneğine sahip olmasını gerektirir.

4.4. Veri Madenciliği Modelleri

Büyük boyuttaki veri kümelerinden bilgi çıkarımı amacıyla farklı modeller geliştirilebilir. Bu modeller tahmin edici ve tanımlayıcı olarak iki gruba ayrılabilir [53]. Tahmin edici yapıda, elde olan veriler üzerinde modeller oluşturularak bu modellere göre gelecekte değerleri bilinmeyen veritabanları için sonuç değerlerinin tahmin edilmesi sağlanır. Bir mağazanın zaman üzerinde geçmişteki satışlarına bakılarak, gelecekte oluşabilecek keskin düşüşler belirlenebilir. Tanımlayıcı yapıda ise var olan veritabanı üzerinden bilgi çıkaracak kararların üretilmesine dayalıdır. Veri madenciliği modelleri, kişilerin satın alma alışkanlıklarını çıkarılması için kullanılan birliktelik kuralları, sınıflandırma, tahmin ve kestirim için kullanılan eğitici öğrenme metotları ve eğitici kümelemeyi kapsar [54]. Veri madenciliği modelleri birçok alanda uygulandığından bu alanlar ile ilgili isimler almışlardır. Örneğin, web sayfalarındaki log bilgilerinden ilginç örüntülerin çıkarılması amacıyla web kullanım madenciliği, büyük tekst verilerinden ilginç örüntülerin aranıp bulunması ile tekst madenciliği veya zaman üzerinde alınan verilerden ilginç örüntülerin çıkarılması ile zaman serileri veri madenciliği gibi alanlar ortaya çıkmıştır. Şekil 4.2’de temel veri madenciliği modelleri verilmiştir.



Şekil 4.2. Veri madenciliği modelleri

4.4.1. Birliktelik Kuralları

Birliktelik kural madenciliği, geniş veri kümeleri içerisinde ilginç birliktelik ve bağıntıların bulunması amacıyla veritabanlarının taranmasıdır. Birliktelik kuralları, her bir kaydı

işlemlerden oluşan ve her bir işlemin de ürün birlikteliklerinden oluştuğu bir veritabanında en sık birlikte alınan ürün kümelerini aramak için veritabanını birçok kez tarar [55]. Büyük veri kümeleri içerisinde ilginç birlikteliklerin bulunması katalog tasarımı, reyon düzenlemesi gibi birçok karar verme sürecine katkıda bulunur. Birliktelik kural madenciliği için en temel örnek market sepet analizidir. Bu süreç müşterilerin alışveriş sepetlerindeki farklı ürünler arasındaki birliktelikleri bularak müşterilerin satın alma alışkanlıklarını analiz eder. Market sepet analizi sayesinde bir marketteki müşteriler için “süt alanların %80’i ekmek ve yumurta da alır” gibi bir kural çıkarılabilir. Birliktelik kuralları çıkarılırken bir işlem esnasında birden fazla ürün alınabildiği için, bir ürünün alınıp alınmadığı sıfır veya bir ile ifade edilir. Örneğin, “dizüstü bilgisayar alan kişiler aynı zamanda Windows Xp işletim sistemini de alır” şeklindeki bir kural denklem 4.1’deki gibi gösterilebilir.

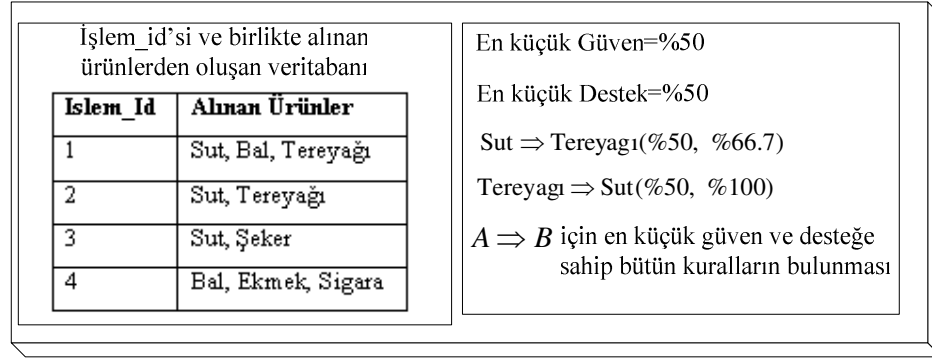
$$\text{Dizüstü Bilgisayar} \Rightarrow \text{Windows Xp} \text{ [Destek}=\%30, \text{Güven}=\%60] \quad (4.1)$$

Denklem (4.1)’deki destek ve güven değerleri, birliktelik kuralının ilginçliğini ölçmek için kullanılır. Buradaki %30 destek değeri, incelenen veritabanındaki kişilerin %30’u hem bilgisayarı hem de Windows Xp yazılımını tercih ettiği anlamındadır. Güven değeri ise bilgisayarı tercih edenlerin %60’ı aynı zamanda Windows Xp yazılımını da aldığı anlamındadır. Destek ve güven değerleri aşağıdaki denklemler ile ifade edilir.

$$\text{Destek}(A \Rightarrow B) = P(A \cup B) \quad (4.2)$$

$$\text{Güven}(A \Rightarrow B) = P(B | A) \quad (4.3)$$

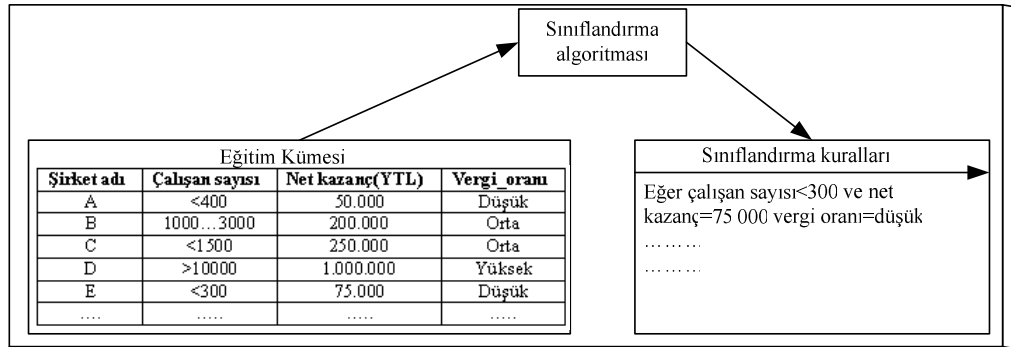
Bir birliktelik kuralının ilginç olabilmesi için belirlenen minimum destek ve güven değerlerini sağlaması gerekir. Denklem (4.2)’de destek değeri, A ve B’nin birlikte bulunabilme olasılığını ifade eder. Denklem (4.3)’teki güven değeri ise A ürününün yer aldığı işlemin aynı zamanda B’yi de içermesi olasılığını ifade eder. Şekil 4.3’te birliktelik kuralı için basit bir örnek verilmiştir.



Şekil 4.3 Birliktelik kuralları için bir örnek.

4.4.2. Sınıflandırma ve Tahmin

Sınıflandırma ve tahmin, önemli veri kümelerini tanımlayan modeller çıkarmak veya gelecekteki veri eğilimlerini tahmin etmek için kullanılır [54]. Sınıflandırma kategorik etiketler ile ilgilenirken, tahmin modelleri sürekli değerlikli fonksiyonlar ile ilgilidir. Örneğin, bir sınıflandırma modeli banka kredi uygulamalarını riskli veya güvenli olarak sınıflandırmak için kullanılabilir. Tahmin modelleri, gelir ve işi verilen müşterilerin giderlerini tahmin etmek için kullanılabilir. Sınıflandırma ve tahmin için makine öğrenmesi, istatistik ve uzman sistem kullanan birçok yöntem geliştirilmiştir. En çok kullanılanlar arasında yapay sinir ağları, bayesian sınıflandırma ve karar ağaçlarıdır [31]. Sınıflandırma işlemi temel olarak iki adımdan oluşur. İlk adımda veri sınıflarının ön tanımlı bir kümesini tanıyacak bir modelin oluşturulmasıdır. Bu eğitim aşaması olarak bilinir. Diğer adım ise gelen test verisinin hangi sınıfa ait olduğunu bulmaktır. Şekil 4.4'te sınıflandırma için eğitim kümesinden kuralların oluşturulmasının basit gösterimi verilmiştir.

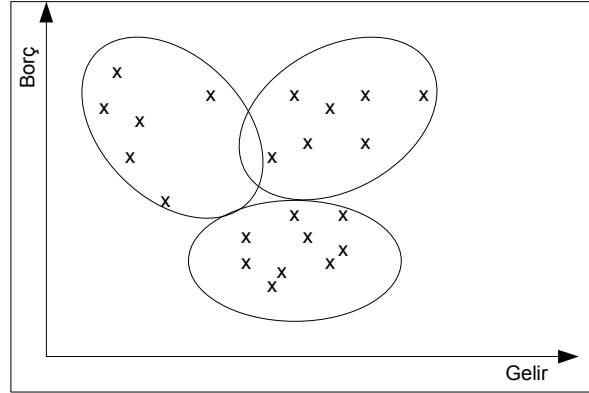


Şekil 4.4. Temel veri sınıflandırma süreci

Şekil 4.4'te gösterildiği gibi eğitim aşamasında sistem için sınıflandırma kuralları elde edildikten sonra gelen test verisinin en uygun düştüğü sınıf belirlenerek sınıflandırma işlemi yapılır.

4.4.3. Kümeleme

Kümeleme, verileri alt gruplara ayırmak için kullanılır. Sınıflandırmadan farklı olarak küme etiketleri yoktur. Aynı küme içerisindeki elemanlar birbirleri ile yüksek benzerlikte iken farklı bir kümedeki eleman ile benzerliği düşüktür [31]. Kümeleme için model tabanlı, hiyerarşik tabanlı, yoğunluk tabanlı ve grid tabanlı yöntemler kullanılır. Kümeleme örüntü tanıma, veri analizi, görüntü işleme gibi birçok alanda kullanılmaktadır. Şekil 4.5'te bir kümeleme örneği verilmiştir.



Şekil 4.5. Kümeleme örneği

Kümeleme modellerinde amaç, şekil 4.5' te görüldüğü gibi küme üyelerinin birbirlerine çok benzediği, ancak özellikleri birbirlerinden çok farklı olan kümelerin bulunması ve veri tabanındaki kayıtların bu farklı kümelere bölünmesidir.

Başlangıç aşamasında, veri tabanındaki kayıtların hangi kümelere ayrılacağı veya kümelemenin hangi değişken özelliklerine göre yapılacağı bilinmemekte, konunun uzmanı olan bir kişi tarafından kümelerin neler olacağı tahmin edilmektedir. Bu yüzden kümeleme, denetleyicisiz öğrenmeye girer. Yaygın kullanım alanları nüfusbilimi, astronomi ve zaman serilerinin sınıflandırılması gibi alanlardır.

4.5. Zaman Serileri Veri Madenciliği

Bir zaman serisi veritabanı, zaman ile değişen bir sistemden elde edilen verilerin belirlenen zaman aralıklarında kaydedilmesi ile oluşturulur [31]. Zaman serilerindeki veriler tipik olarak eşit zaman aralıklarında ölçülür. Zaman serisi veritabanına örnek olarak medikal veriler, bilimsel deneyler, bir stok marketin günlük ürün satışları, bir asenkron motorun akım ve hız bilgileri ve bir rasathaneden alınan deprem bilgileri verilebilir. Farklı web sayfalarının zaman üzerinde kullanımını kaydeden web verisi de zaman serisi olarak modellenebilir. Tipik olarak her zaman serisi, verilen bir veri toplama istasyonundaki zamanın bir fonksiyonu olarak (örneğin bir stokun fiyatı) bir nesnenin değişimini tanımlar. Gerçek dünyada zaman serileri, ticari veritabanlarında depolanan verinin geniş bir bölümü için kullanılır [30]. Zaman serilerinden özellik çıkarılması amacıyla kullanılan matematiksel yöntemler zaman serileri analizi olarak bilinir. Zaman serileri analizi, lineer ve lineer olmayan olarak iki gruba ayrılır. Lineer zaman serileri analiz yöntemini kullanarak bir zaman serisinin analiz edilmesi için serinin durağan hale getirilmesi gerekmektedir. Ayrıca hareketli ortalama yöntemi ve box-jenkins gibi zaman serileri analiz yöntemleri çok fazla matematiksel işlemler gerektirmesinden dolayı tercih edilmemektedir. Lineer olmayan zaman serileri analizi, kaos ve gömülme boyutu gibi tekniklere dayalı olup, zaman serisinin durgun hale getirilmesi gibi gereksinimlere ihtiyaç duymaz.

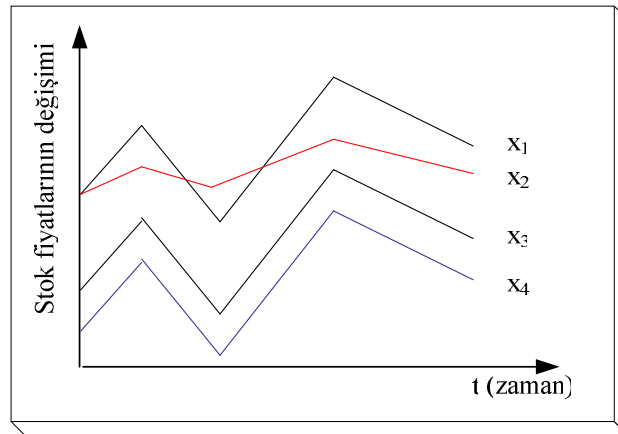
Zaman serisi verisinin önemi ve yaygınlığı, birçok araştırmacıyı bu verilerin analizine yönlendirmiştir. Zaman serisi üzerinde istatistik literatürü çok geniştir ve temel olarak araştırmacılar örüntü tanıma, eğilim analizi (yıllara göre satışlardaki lineer büyüme), mevsimsellik (kış satışları yaz satışlarının iki katı olması) ve öngörü gibi problemlere odaklanır [56]. Bilgisayar bilimcileri zaman serileri için benzerlik araması, indeksleme, olay tahmini ve sınıflandırma gibi konuları çalışmışlardır. Farklı zaman serilerinin davranışlarındaki benzerlikleri bulmak için benzerlik araması, zaman serilerinden daha hızlı bilgi çıkarılması için indeksleme ve boyut azaltma teknikleri zaman serileri veri madenciliğinde çalışılan konulardandır. Zaman serileri genelde lineer olmayan davranışlar gösterdiğinden, bu serilerin analizi için uygun bir modelin çıkarılması zordur. Zaman serilerin analizi ve özellik çıkarılması için lineer olmayan yöntemler geniş olarak kullanılmıştır. Bu yöntemler kullanılarak zaman serilerinden ilginç bilgiler çıkarılabilir. Bu yöntem için zaman serilerinden önemli olayların çıkarılması, aynı tipten zaman serilerinin davranışlarına göre sınıflandırılması gibi örnekler verilebilir. Zaman serisinden bilgi çıkarılması için faz alanı olarak isimlendirilen bir ön işleme adımının uygulanması gerekir. Ön işleme adımında faz alanını oluşturmak için uygun bir gömülme boyutu ve zaman gecikmesi seçilmesi gerekir. Daha sonra bu faz alanı verileri

üzerinde genetik algoritma gibi optimizasyon yöntemler kullanılarak olay tahmini yapılır ve bayesian sınıflandırma veya yapay sinir ağları gibi yöntemler kullanılarak sınıflandırma işlemleri yapılır.

4.5.1. Zaman Serilerinde Benzerlik Araması

Bir serinin zaman üzerindeki değişimine benzer davranışlar gösteren zaman serilerinin bulunması amacıyla geliştirilen teknikler zaman serilerinde benzerlik araması olarak bilinir. Bu yöntem ile örneğin bir marketteki ürün satışlarının bir önceki yıl ile benzerlik gösterip göstermediği veya hangi dönemlerde benzersizlik gösterdiği bulunabilir. Veritabanından verilen bir zaman serisine benzer veri sıralarının bulunması benzer seri eşleşmesi olarak bilinir [57].

Benzerlik eşleşmesi yapılırken, X ve Y gibi iki zaman serisi için belirlenen benzerlik ölçümüne göre $Ben(X,Y)$ gibi bir fonksiyon tanımlanır. Hedef, benzer gruplara farklı nesneleri kümeleyebilmek veya bilinen nesne örneklerinin bir kümesine dayalı olarak bir nesneyi sınıflandırabilmektir. Benzerlik modelinin keskin eşleşmelere izin vermesi gerektiğinden zor bir problemdir. Benzerlik probleminin ilginç bir uygulama alanı ise alt seri eşleşme problemidir. Burada amaç, verilen bir X zaman serisi ve daha kısa bir Y zaman serisi için Y örüntüsüne benzer olan X'in alt sıralarını bulmaktır. Bu tür problemler için zaman serileri arasında benzerliğin farklı görüşleri veri madenciliği araştırmalarında önerilmiştir. Önerilen teknikler veritabanları, işaret işleme, konuşma tanıma, örüntü eşleşme ve istatistik gibi geniş farklı disiplinlerden teknikleri kapsar [30]. Şekil 4.6'da farklı zaman serisi örnekleri verilmiştir.



Şekil 4.6. Zaman serisi örnekleri

Şekil 4.6'da, x_1 , x_3 ve x_4 'ün birbirine benzer davranışlar gösterdiği ve x_2 'nin bu serilere daha az benzediği görülmektedir. Benzerlik ile ilgili en çok kullanılan ölçümler, öklid normları

ve dinamik zaman eğriliği modellerine dayalıdır. Aşağıda bu yöntemlerden detaylı olarak bahsedilmektedir.

4.5.1.1. Benzerlik Araması için Öklid ve Manhattan Ölçümü

Zaman serileri için en basit benzerlik ölçülerinden biri öklid mesafesi ölçümüdür. Her ikisinin de boyutu n olan X ve Y gibi iki zaman serisi verilsin. Her bir seri n boyutlu öklid uzayında bir nokta olarak alınır ve X ile Y arasındaki benzerlik Ben(X,Y) olarak tanımlanır. İki nokta arasındaki mesafe ölçüm denklemi aşağıda verilmiştir.

$$\text{Ben}(X, Y) = \left(\sum_{i=1}^n |x_i - y_i|^p \right)^{1/p} \quad (4.4)$$

Denklem (4.4)'te p=1 için Manhattan mesafesi ve p=2 için öklid mesafesi ölçülmüş olunur. Bu tür bir benzerlik ölçümünün kullanımı, anlaşılması ve hesaplanması kolay olduğu için avantaj sağlar [30]. Ayrıca zaman serisi indeksleme ve kümeleme problemleri için zaman serisini ölçeklendirilebilmesine izin verir. Fakat buna rağmen birçok uygulamada uygun olmayan sonuçlar verir. En büyük dezavantajlarından biri, zaman serilerindeki farklı taban çizgilerine izin vermemesidir. Örneğin, şekil 4.5'te stok fiyatlarını gösteren x_1 serisi 70 YTL ve x_4 serisini 30 YTL civarında dalgalandığını düşünelim. Her iki zaman serisinin şekli birbirine çok benzer olmasına rağmen, ikisi arasındaki öklid mesafesi çok büyük olmaktadır. Bu iki seri benzer olmalarına rağmen farklıymış gibi bir durum ortaya çıkar.

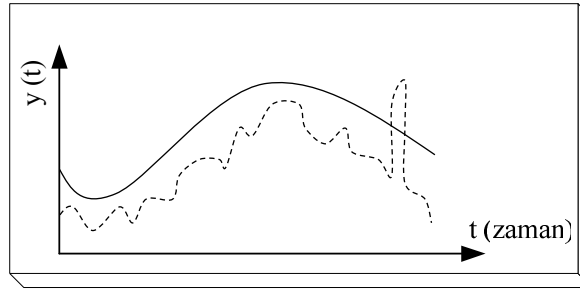
4.5.1.2. Benzerlik Araması için Normalleştirme Dönüşümü

Öklid mesafesi ölçümündeki dezavantajları gidermek için Goldin ve diğ. [58], benzerlik ölçümü olarak normalleştirme yöntemini önermişlerdir. Önerdikleri yönteme göre bir $X=\{x_1, \dots, x_n\}$ serisinin ilk olarak ortalaması ve standart sapması sırasıyla $\mu(X)$, $\sigma(X)$ hesaplanır ve normalleştirilmiş X' serisi oluşturulur. Normalleştirme formülü denklem (4.5)'te verilmiştir.

$$x'_i = (x_i - \mu(X)) / \sigma(X) \quad (4.5)$$

Benzer şekilde Y serisi de Y' serisine dönüştürülür. Böylece X ve Y serisi arasındaki benzerlik Ben(X',Y') olarak tanımlanır. Bu benzerlik tanımlaması öklid mesafe ölçümündeki

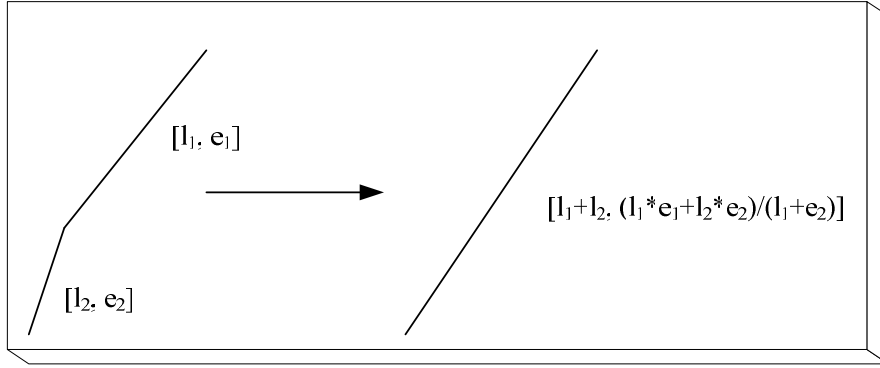
dezavantajları giderir. Bununla birlikte normalleştirme işlemi bazı dezavantajlara sahiptir. Bunlardan biri, zamanda faz kaymalarına çok duyarlı olmasıdır. Örneğin, X ve Y gibi iki seri düşünüldüğünde X'in sinüs olduğu yerde, Y kosinüs'e benzer. Kaydırma işlemi yapıldığında her iki serinin benzer olduğu görülecektir. Fakat kaydırma olmadan bu iki seri arasındaki öklid mesafesi çok büyük olacaktır. Öklid mesafesi zaman eksenini boyunca hızlanma ve yavaşlamaya izin vermez. Örneğin X ve Y gibi iki sinüs dalgamız olsun. Bu dalgalardan X'in frekansı Y serisinin frekansının iki katı olsun. Normalleştirme ile bile öklid mesafesi iki sinyal arasındaki benzerliği yanlış tespit eder. Ayrıca bu yaklaşımda, zaman serisi verilerine bir gürültü eklendiğinde benzerlik sonuçları yanlış çıkacaktır. Aşağıda bir zaman serisi ve bu seriye gürültü eklendiğinde oluşan serinin grafiği verilmiştir.



Şekil 4.7. Gürültülü ve gürültüsüz zaman serisi

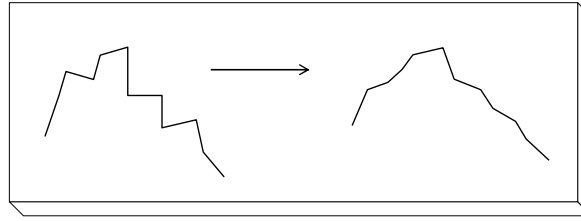
4.5.1.3. Genel Dönüşümler

Benzerlik hesaplamalarındaki şekil kavramının öneminin tanımlanmasında, alternatif bir yaklaşım Jagadish ve diğ. [59] tarafından sunulmuştur. Yaptıkları çalışmada bir dönüşüm kural dili içeren genel bir benzerlik çatısı tanımlanmıştır. Dönüşüm dilindeki her bir kural giriş olarak bir seri alır ve kural ile ilişkili olan bir maliyette bir çıkış serisi üretir. Y serisine X serisinin benzerliği, bu kuralların bir sırasını uygulayarak X'i Y'ye minimum dönüştürme maliyeti ile hesaplanır. Bir dönüşüm kuralı olarak bitişik iki parçayı tek bir parçaya dönüştürmek olabilir. Şekil 4.8'de bu kurala uygun bir hesaplama gösterimi verilmiştir.



Şekil 4.8. Dönüşüm kuralı uygulama örneği

Şekil 4.8’de ilk zaman serisi üzerine uygulanan bir dönüşüm kuralı ile ikinci zaman serisi oluşturulmuştur. Bu şekilde l_1 ve l_2 lineer iki parçadan oluşan zaman serisinin her bir parçasının uzunluğunu gösterirken, e_1 ve e_2 ise her bir parçanın eğimini gösterir. Dönüşüm için kullanılan kural ile dönüştürülmüş serinin uzunluğu iki parçanın uzunlukları toplamına ve yeni eğim ise önceki eğimlerin ağırlıklı ortalamasına eşittir. Bu kuralın uygulandığı örnek zaman serisi ve dönüşüm sonrası oluşan yeni zaman serisinin grafiği şekil 4.9 ’da gösterilmiştir.



Şekil 4.9. Dönüşüm kurallarının zaman serilerine uygulanması

Rafei ve diğ. [60] ölçekleme, kaydırma ve hareketli ortalamalar gibi dönüşümler ile uygulamaları tanımlamıştır. Hareketli ortalamalar, düzgünleştirilmiş zaman serisi için teknikler olarak bilinir. Örneğin, bir X stokunun belirli gün sayısı üzerinde bir stokun günlük kapanış fiyatını gösterebilir. Bir X serisi için üç günlük hareketli ortalama dönüşümü $x'_i = (x_{i-1} + x_i + x_{i+1}) / 3$ şeklinde olur. Ölçekleme dönüşümü, bir ölçekleme sabiti ile her bir elemanı çarpar. Kaydırma dönüşümü, sağa veya sola bir sayı ile her bir elemanı kaydırır.

4.5.1.4. Dinamik Zaman Eğriliği

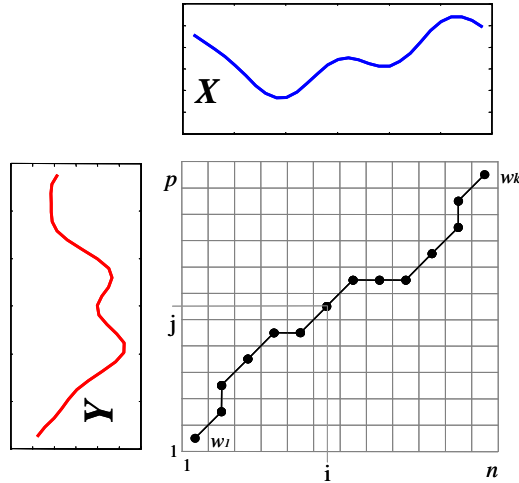
Zaman serilerinde benzerlik aramasında öklid mesafesi, normalleştirme ve genel dönüşümlerdeki eksiklikleri gidermek için dinamik zaman eğriliği (DZE) önerilmiştir [61]. Bu

yöntem, özellikle konuşma tanımada geniş olarak kullanılmaktadır. DZE zaman serilerinde hızlanma ve yavaşlamaya izin verir. Yani zaman üzerinde alınan verilerin frekanslarının değişimine izin verir.

X ve Y gibi iki seri düşünelim. $X=\{x_1, \dots, x_m\}$ ve $Y=\{y_1, \dots, y_m\}$ olsun. Her iki seride de tekrarlı elemanlara izin verilebilir. Örneğin $X'=\{x_1, x_2, x_2, x_3, x_3, \dots\}$ ve $Y'=\{y_1, y_1, y_2, y_2, y_2, y_2, \dots\}$ gibi yeni seriler oluşturulabilir. DZE(X,Y) mesafesi aşağıda denklem (4.6)'deki gibi hesaplanır.

$$DZE(X, Y) = \min_{\forall X', Y' \text{ için } |X'|=|Y'|} L_1(X', Y') \quad (4.6)$$

Şekil 4.10'da iki seri arasındaki dinamik zaman eğriliğini gösteren bir grafik verilmiştir. Burada iki seri arasındaki zaman eğriliğinde bazı sınırlamalar vardır. Zaman eğriliği monoton olmak zorunda olup aşağı veya sola gidemez. Aynı zamanda bu eğri sürekli olup serilerdeki hiçbir elemanın atlanmaması gerekir. Bazen hesaplama etkinliği, w boyutlu için sınırlı bir eğrilik yolu tanımlanabilir. Bu değer $li-jl \leq w$ eğrilik yolunda herhangi bir grid noktası içindir.



Şekil 4.9. İki seri arasındaki dinamik zaman eğriliği grafiği

DZE için geliştirilecek algoritmanın amacı, minimum maliyetli bir eğrilik yolunun bulunmasını sağlamak olmalıdır. Bu amaçla öz yinelemeli bir DZE algoritması kullanılabilir. $\{x_1, \dots, x_i\}$ ve $\{y_1, \dots, y_j\}$ gibi iki alt seri arasındaki mesafe öz yinelemeli olarak aşağıdaki gibi verilebilir.

$$D(i, j) = |x_i - y_j| + \min\{D(i-1, j), D(i-1, j-1), D(i, j-1)\} \quad (4.7)$$

DZE benzerlik ölçümü algoritması, ölçekleme ve farklı taban çizgilerine duyarlılık gibi eksikliklerden etkilenir. Bu eksikliklerin giderilmesi için serilerin normalleştirilmesi gerekir.

4.5.2. Zaman Serileri için İndeksleme Teknikleri

Bir indeks, veriyi hızlı bir şekilde elde etmeye izin veren veri organizasyon yapısıdır [30]. Büyük bir zaman serisini analiz etmek için, belirli bir özelliğe sahip olan zaman serisini bulmamız gerekir. Örneğin, belirli bir biçimde olan bir zaman serisi veya belirli bir değere yakın ortalamaya sahip bir zaman serisi olmalıdır. Böyle bir sorgunun sonuçları genellikle sadece verinin küçük bir kısmı ile döner. Bir indeks kullanmadan zaman serisi veritabanını sorguladığında, bütün zaman serisi verisi istenen özelliğe sahip olup olmadığını test etmek için bulup getirilir. Bu hem CPU hem de G/Ç maliyeti açısından aşırı derecede zaman ve enerji kaybıdır. Böylece indeksler zaman serisi verisinde büyük ölçekli bilgilerin keşfi için gereklidir. Zaman serisi indeksleme problemi özellikle son birkaç yılda araştırmacıların büyük ilgisini çekmiştir ve bu kısmen veritabanı boyutunun artışına bağlanabilir. Örneğin yeryüzü için toplanan uzay resim veritabanını düşünelim. Eğer kullanıcılara bu büyük veritabanlarını göstermeye izin vermek istiyorsak, ilgili veriyi doğru ve etkili bir biçimde organize etmek gerekir. Bir uygulama örneği olarak, bir yıl boyunca bir stokun değişimini gösteren zaman serisi içinde bugünün değişimine en yakın değişimi bulmak verilebilir. Bu değişim alt seri indeksleme problemi olup, her bir elemanı bir seri olan S kümesi ve X sorgu örüntüsü verildiğinde S 'te X 'e benzer alt serileri bulmaktır [58]. Bu problemleri çözmek için uygun indeksleme teknikleri kullanılmak zorundadır. Benzerlik problemi temel olarak indeksleme problemi ile ilişkilidir. Basit benzerlik ölçümleri genellikle indeksleri kolay oluştururken, daha karmaşık benzerlik ölçümlerinde indeksleme problemi zor ve ilginçtir.

Dönüşüm kuralları belirli uygulamalar için adapte edilebilen benzerlikleri tanımlamak için çok genel bir mekanizma sağlar. Bununla birlikte bazı dezavantajları vardır. Alt seri hesaplamaları zor olur. Çünkü bir X serisinden özellikler çıkarmak eğer kurallar X ve Y 'nin özel çiftlerine bağlı olarak uygulanırsa karmaşık olur. Hem de özellik uzayında öklid mesafeleri orijinal seri benzersizliğinin sağlanmasında iyi yaklaşımlar sunmayabilir.

4.5.3. Zaman Serilerinde Veri Azaltma Teknikleri

Veri azaltma, önemli bir veri madenciliği kavramıdır. Veri azaltma teknikleri, mümkün olduğu kadarıyla verinin karakteristiklerini koruyarak yönetilebilir veri yapısına büyük veriyi

düşürür. Bu kavram, büyük miktardaki veriyi keşfetmek ve hızlı analiz etmek için gereklidir [30]. Veri azaltma zaman serisinin iki karakteristiğinden dolayı büyük zaman serisi verisi için kullanışlıdır. Zaman serisi için veri azaltma, zaman serisinin boyutunu da azalttığı için zaman serisi için boyut azaltma olarak ta çağrılır. Veri madenciliği bakış açısından, zaman serileri iki önemli karakteristiğe sahiptir:

- **Yüksek Boyut:** Eğer zaman serisindeki her zaman noktasını bir boyut olarak varsayarsak, bir zaman serisi çok yüksek boyutlarda bir noktadır. 1000 boyutunda bir zaman serisi, 1000-boyutlu bir alanda bir noktaya karşılık gelir. 1000-boyutlu bir zaman serisi pratikte çok sık rastlanmasına karşın, 1000-boyutlu bir alanda veri işleme modern bilgisayar sistemleri olsa bile çok zordur.
- **Zamana Ait Düzen:** Bir zaman serisindeki ardışık değerler bir zaman serisinin zamana ait düzeninden dolayı ilişkilidir. Örneğin, finansal zaman serisi için ardışık değerler arasındaki farklar çoğu zaman tahmin edilebilir bir eşik içinde olacaktır.

Uzunluğu N olan bir zaman serisi n -boyutlu bir uzayda bir kayıt olarak düşünülebilir. Bu alanı direkt olarak indekslemek yüksek boyuttan dolayı yetersizdir. Temel fikir bir boyut azaltma tekniği kullanmaktır. Boyut azaltma tekniği n uzunluğunda bir X zaman serisini alır ve bu zaman serisini temsil eden özellikler ile ifade edecek bir boyutta $f(X)$ 'i elde edilir. Böylece X ve Y arasındaki benzerlik veya benzersizlik yaklaşık olarak $f(X)$ ile $f(Y)$ noktaları arasındaki öklid mesafesine eşittir. Boyut azaltmanın birkaç avantajı vardır. İlk, indeksleme yapıları daha düşük boyutlarda daha iyi çalışır. Mesafeyi hesaplama zamanı boyutların sayısı ile orantılı olduğundan mesafe hesaplamaları daha hızlı olmaktadır. Sonunda veritabanının tamamının boyutu daha küçük olur ve bu daha iyi performans sağlar.

4.5.4. Lineer Olmayan Zaman Serileri Analizi ve Veri Madenciliği Teknikleri

Lineer zaman serileri analiz yöntemleri, bir veri kümesi içindeki baskın frekanslar ve lineer bağıntılar gibi düzenli yapıları yorumlarlar. Bu yöntemlerde sistemin gerçek dinamikleri, küçük değişimlerin küçük etkilere yol açtığı lineer paradigmalar ile yönetilir [62]. Lineer denklemler sadece üstel veya periyodik olarak salınan çözümler üretirken, bir sistemin bütün dinamik davranışları sisteme eklenen harici bir rasgele değişkenden etkilenir. Lineer olmayan bu davranışlar sadece rasgele bir giriş ile değil, aynı zamanda sistemin deterministik denklemleri ile de oluşturulur. Zaman serilerinden bilgi çıkarmak amacıyla kullanılan benzerlik araması yönteminde zaman serilerinin benzerliği bir hata toleransına göre belirlenir. Dolayısıyla iki seri

arasındaki benzerlik, hata toleransı ne kadar düşükse o kadar yüksek olur. Buna rağmen iki seri arasında tam bir benzerliğin bulunması çoğu zaman zordur. Ayrıca zaman serileri yüksek boyutta olduklarından boyut azaltma teknikleri kullanılmaktadır. Fakat bu tekniklerde de genelde zaman serisi üzerine uygulanan dönüşümler zaman bilgisinin kaybolmasına neden olur. Ayrıca aynı olan iki zaman serisinin farklı zamanlarda başlaması veya farklı frekanslarda verilerin kaydedilmesi benzerlik aramasının performansını düşürür.

Lineer olmayan zaman serileri analiz yöntemi, bir zaman serisinin faz alanında incelenmesini gerektirir [29]. Faz alanı, bir zaman serisinin uygun bir zaman gecikmesi ve gömülme boyutu seçilerek oluşturulan matristir. Uygun bir zaman gecikmesi ve gömülme boyutu seçilerek oluşturulan faz alanı ile zaman serilerinin benzerliğinin bulunması, sınıflandırma ve olay tahmini gibi birçok zaman serisi problemi kolayca çözülebilir. Lineer zaman serileri analizi yöntemi ile bir serinin analiz edilebilmesi için o serinin durgun hale getirilmesi gerekmektedir. Buna karşın lineer olmayan zaman serileri analiz yönteminde bu tür gereksinimlere ihtiyaç yoktur.

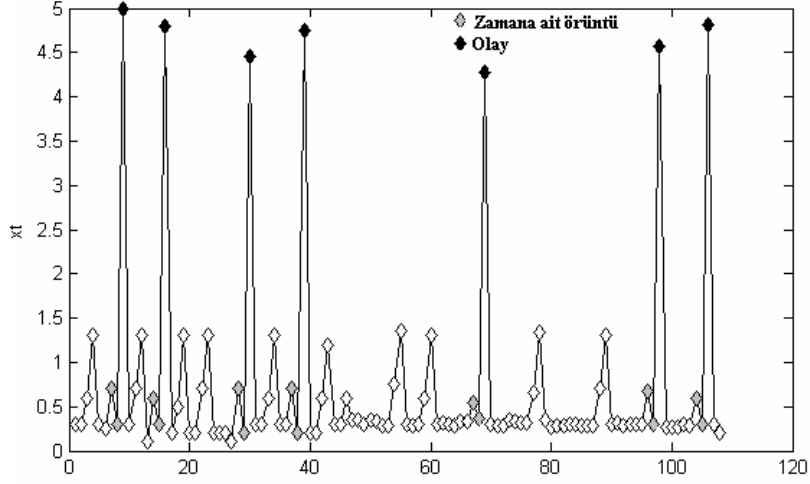
Veri madenciliği yöntemlerinin lineer olmayan zaman serilerinin analizi için kullanılması ile yeni bir zaman serileri veri madenciliği yöntemi Povinelli [29] tarafından geliştirilmiştir. Lineer olmayan zaman serileri analiz yönteminin en temel işlemleri olan zaman gecikmesi ve gömülme boyutunun seçilerek faz alanının oluşturulması işlemi, zaman serileri veri madenciliği yönteminin ön işleme adımını oluşturur. Faz alanı oluşturulduktan sonra, veri madenciliği algoritmaları zaman serisinden ilginç örüntülerin çıkarılması amacıyla uygulanır. Bu yöntem ile sismik bir zaman serisinde oluşan depremler, stok fiyatlarının keskin düşüşü gibi olayları önceden tahmin edecek zamana ait örüntü kümeleri bulunabilir. Ayrıca asenkron motorlardan zaman üzerinde alınan akım ve moment bilgileri ile motor arızaları sınıflandırılabilir. Uygulamalar sadece bunlarla sınırlı kalmayıp; benzer zaman serilerinin gruplandırılması, elektrokardiyogram cihazlarından zaman üzerinde alınan veriler ile hastalık teşhisi ve konuşma tanıma gibi birçok alanda uygulanabilir.

4.5.4.1. Zaman Serileri Veri Madenciliğinde Bazı Kavramlar

- **Olay, zamana ait örüntü ve zamana ait örüntü kümesi**

Bir zaman serisinde olay kavramı önemli bir oluş olarak ifade edilir [63]. Bir olayın tanımı zaman serileri veri madenciliğinin amacına bağlıdır. Örneğin, sismik bir zaman serisinde bir deprem olay olarak ifade edilirken, bir asenkron motora ait akım verilerini gösteren zaman serisinde bir arıza olay olarak ifade edilir. Ayrıca stok fiyatlarındaki keskin düşüşler veya ani

yükselmeler de birer olay olarak ifade edilir. Şekil 4.11’de sismik bir zaman serisi örneği verilmiştir.



Şekil 4.11. Sismik bir zaman serisi

Yukarıda sismik bir zaman serisindeki olaylar ve zamana ait örüntüler gösterilmiştir. Zamana ait bir örüntü, zaman serileri veri madenciliğinde diğer önemli bir kavramdır. Zamana ait bir p örüntüsü, zaman serisinde gizlenmiş yapılar olup olayların tahmini ve karakteristiğini tanımlar. Bu örüntüler $\mathcal{R}^Q$ ’da $Q \times 1$ boyutunda bir vektördür. Bir zaman serisinde bir olayın olacağını önceden tespit etmek için zamana ait örüntünün kullanılması pek uygun değildir. Çünkü yukarıdaki örnekte de görüldüğü gibi bütün depremlerin şiddeti aynı değildir. Ayrıca zamana ait örüntüler de gürültü içerebilirler. Olayların tahmini için zamana ait örüntüye alternatif olarak zamana ait bir P örüntü kümesi olay tahmini için kullanılabilir. P, p’nin bir komşusu olup belirli bir δ mesafesindeki bütün noktaları içeren kümedir ve aşağıdaki denklem ile ifade edilir.

$$P = \{a \in \mathcal{R}^Q : d(p, a) \leq \delta\} \quad (4.8)$$

Bu denklemde d faz alanında tanımlanmış bir mesafe ölçümüdür. Zamana ait örüntü kümesinin merkezini p oluşturmaktadır. Faz alanında bir vektör aşağıdaki denklemde verilmiştir.

$$X_t = (x_{t-(Q-1)\tau}, x_{t-(Q-2)\tau}, \dots, x_t) \quad (4.9)$$

Burada $Q > 0$ gömülme boyutunu, $\tau > 0$ zaman gecikmesini, t zaman indeksini ve X_t ise faz alanında bir noktayı gösterir. Bu gösterim kullanıldığında zaman ifadesi geçmiş, gelecek ve şu andaki durum olmak üzere üç kategoriye ayrılır. Zamana ait örüntü geçmişte oluşup içinde bulunduğumuz anda tamamlanırken, bir olay ise gelecekte oluşur.

- **Faz Alanı ve Zaman-Gecikmeli Gömülme**

Faz alanı, bir zaman serisinin seçilen gömülme boyutu ve zaman gecikmesine göre oluşturulur. Gömülme boyutu uygun seçildiğinde, faz alanı zaman serisini en iyi biçimde temsil edebilir [34]. Aşağıda örnek bir zaman serisi ve bu serinin faz alanı matrisi verilmiştir.

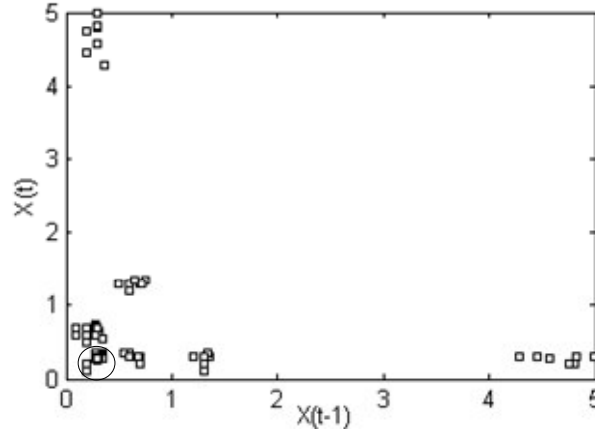
$$X = \{x_t, t=1 \dots N\} \quad (4.10)$$

$$F = \begin{bmatrix} X_{1+(Q-1)\tau} & \dots & X_{1+\tau} & X_1 \\ X_{2+(Q-1)\tau} & \dots & X_{2+\tau} & X_{1+\tau} \\ \vdots & & \vdots & \vdots \\ \vdots & \dots & \vdots & \vdots \\ \vdots & & \vdots & \vdots \\ X_N & \dots & X_{N-(Q-2)\tau} & X_{N-(Q-1)\tau} \end{bmatrix}_{(N-(Q-1)\tau) \times 1} \quad (4.11)$$

Yukarıdaki X zaman serisi $Q=3$ ve $\tau = 2$ için faz alanı matrisi denklem 4.11’de yerine konulursa aşağıdaki gibi oluşturulur.

$$F = \begin{bmatrix} X_{1+4} & X_{1+2} & X_1 \\ X_{2+4} & X_{2+2} & X_2 \\ \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots \\ X_N & X_{N-2} & X_{N-4} \end{bmatrix} \quad (4.12)$$

Denklem 4.12’de verilen faz alanı matrisine göre üç boyutlu bir faz alanı oluşturulur. Matrisin her bir satırı faz alanında bir noktaya karşılık gelir. Gömülme boyutu, ardışık olmayan zaman gecikme örnekleri ile faz alanı matrisinin oluşturulmasını sağlar. Şekil 4.10’da verilen deprem zaman serisinin $Q=2$ ve $\tau = 1$ için faz alanı şekil 4.12’de verilmiştir.



Şekil 4.12. Faz alanı

Şekil 4.12’de, daire içindeki noktalar zamana ait örüntü kümesini gösterirler. Kümenin merkezi ise zamana ait örüntüyü gösterir. Zaman-gecikmeli faz alanı, şekilden de görüldüğü gibi orijinal bir zaman serisinin Q-boyutlu bir faz alanına dönüştürülmesini sağlar.

- **Olay Tanımlama Fonksiyonu**

Gelecekte oluşan bir olay ile zamana ait örüntü kümesini ilişkilendirmek için olay tanımlama fonksiyonu $g(.)$ tanımlanır [29]. Olay tanımlama fonksiyonu, verilen bir zaman indeksi için gelecekte bir olayın oluşma şiddetini belirler. Bu fonksiyon zaman serisinin amacına göre tanımlanır ve bir zaman serisi veri madenciliği probleminin ilk adımını oluşturur. Şekil 4.10’da verilen sismik zaman serisinde depremi bir adım öncesinden tahmin edecek bir olay tanımlama fonksiyonu denklem (4.13) ‘de verilmiştir.

$$g(x_t)=x_{t+1} \quad (4.13)$$

Denklem (4.13)’deki olay tanımlama fonksiyonu, bir adım sonrasında oluşacak bir depremi tahmin etmek için kullanılabilir. Aynı şekilde, iki adım sonrasında oluşacak bir depremi tahmin etmek için ise $g(x_t)=x_{t+2}$ gibi bir olay tanımlama fonksiyonu kullanılabilir. Fakat bu tür bir fonksiyonda bazı ara değerler kaçırılabilceği için yanlış tahminler yapılabilir. Farklı zaman serisi problemleri için farklı olay tanımlama fonksiyonu kullanılır. Örneğin, bir marketteki stok günlük stok fiyatlarını tutan zaman serisi için, fiyatlardaki keskin düşüşü bulurken aşağıdaki olay tanımlama fonksiyonu kullanılabilir.

$$g(x_t) = \frac{x_{t+1} - x_t}{x_t} \quad (4.14)$$

- **Hedef Fonksiyonu**

Hedef fonksiyonu, olayları tanımlayacak zamana ait örüntü kümelerinin toplamının etkinliğini gösterir. Hedef fonksiyonu, bir zamana ait örüntü kümesinin gelecekte oluşacak olayları tahmin etme kabiliyetlerini tanımlar. Hedef fonksiyonunun seçimi genelde zaman serisi veri madenciliğinin amacına bağlıdır. Bu fonksiyon uygun seçilerek, olay tanımlama fonksiyonunun değerine göre gelecekte oluşacak olan olayları tahmin edecek zamana ait örüntü kümeleri belirlenir. Povinelli [29], yaptığı çalışmada deprem tahmini ve stok fiyatlarındaki düşüşlerin tahmini için aşağıdaki gibi bir hedef fonksiyonu tanımlamıştır.

$$f(v, \delta) = \begin{cases} \mu_M, & \|x_i - v\| < \delta \text{ için eğer } M \geq \beta N \text{ ise} \\ (\mu_M - g_0) \frac{M}{\beta N} + g_0, & \text{diğer} \end{cases} \quad (4.15)$$

Bu denklemde g_0 kümedeki en küçük olay değerini, M kümedeki noktaların sayısını, N toplam nokta sayısını, δ küme yarıçapını, v küme merkezini ve β ise minimum küme boyutunun oranını gösterir. Denklemde kullanılan μ_M ise aşağıdaki gibi tanımlanır ve küme içindeki olay tanımlama fonksiyonlarının ortalamasını gösterir.

$$\mu_M = \frac{1}{M} \sum_{i=1}^M g(X_i) \quad (4.16)$$

Povinelli [29] tarafından önerilen bu hedef fonksiyonu ve örüntü kümesinin en önemli dezavantajı kümenin klasik küme gibi keskin olmasıdır. Yani faz alanında bir nokta ya kümenin içindedir veya dışındadır gibi keskin bir mantık vardır. Ayrıca kümenin merkezindeki bir nokta ile merkeze uzak bir noktanın bir olayı tanımlama derecesi aynıdır. Bu eksikliği gidermek için Feng ve diğ. [64], bulanık bir zamana ait örüntü kümesi önermiştir. Gaussian biçimli bir bulanık üyelik fonksiyonu tanımlanarak faz alanındaki her bir noktaya gelecekteki bir olayı tanımlama derecesi verilmiştir. Dolayısıyla olayı tahmin edecek bir örüntü kümesinin merkezindeki bir nokta en yüksek bir olay tanımlama değerine sahipken, merkezden uzakta bir nokta daha düşük bir olay tanımlama değerine sahip olacaktır. Denklem 4.17’de bulanık zamana ait örüntü kümesi verilmiştir.

$$BP = \{(X_t, \mu_{BP}(X_t)) \mid X_t \in \mathbb{R}^Q\} \quad (4.17)$$

Bu denklemde $\mu_{BP}(X_t)$ BP örüntü kümesine X_t 'nin ait olma derecesini gösterir. Kullanılan Gaussian biçimli fonksiyon aşağıda verilmiştir.

$$\mu_{BP} = e^{-\frac{\|v - X_t\|^2}{2\delta^2}} \quad (4.18)$$

Bulanık örüntü kümesi BP'nin merkez noktası v ve kümenin yarıçapı δ olarak seçilmiştir. Faz alanında bir X_t noktasının v merkez noktasına olan uzaklığı $\|v - X_t\|$ ifadesi ile verilir ve bir X_t noktasının kümeye ait olma derecesini gösterir. Hedef fonksiyonu da bu denkleme göre yeniden düzenlenerek aşağıdaki gibi yazılabilir.

$$f(v, \delta) = \max_{v, \delta} \left\{ \sum_{i=1}^N \mu_{BP}(X_i) g(X_i) \right\} \quad (4.19)$$

Burada hedef fonksiyonu, olay tanımlama fonksiyonunun faz alanındaki her bir noktanın kümeye ait olma derecesi ile ağırlıklandırılarak elde edilmiştir. Dolayısıyla sadece küme içindeki noktalar değil aynı zamanda küme dışındaki noktalarda olayı tahmin etmek için kullanılır.

4.5.4.2. Zaman Serileri Veri Madenciliği Algoritmasının Genel Çatısı

Zaman serileri veri madenciliğinde ilk aşama, bir zaman serisinden elde edilecek bilginin ne olacağını belirlemektir. Zaman serilerinden farklı amaçlar için farklı bilgiler çıkarılabilir. Eğer zaman serisinde gelecekte oluşacak bir olayın önceden tahmin edilmesi ile ilgili bir uygulama yapılmak isteniyorsa kullanılacak olay tanımlama fonksiyonunun belirlenmesi gerekir. Zaman serisi üzerinde eğer gerekiyorsa bir ön işleme veya filtreleme işlemi yapılır. Daha sonra hedef fonksiyonu belirlenmeli ve genetik algoritma, klonal seçim gibi yapay zeka optimizasyon algoritmaları kullanılarak olayları tahmin edecek örüntü kümeleri bulunabilir. Eğer zaman serilerinin sınıflandırılması ile ilgili bir uygulama yapılacaksa; olay tanımlama fonksiyonunu tanımlamaya gerek yoktur. Gerekli filtreleme ve ön işleme adımlarından sonra, gömülme boyutu ve zaman gecikmesi parametreleri seçilerek faz alanı

oluşturulur. Bu faz alanında bayesian sınıflandırma, yapay sinir ağları, bulanık mantık gibi yapay zeka sınıflandırma algoritmaları kullanılarak zaman serileri sınıflandırılır. Aşağıda bir zaman serisinden ilginç olayların çıkarılması için kullanılan algoritmanın genel adımları verilmiştir. Algoritma eğitim ve test aşaması olmak üzere iki aşamadan oluşur.

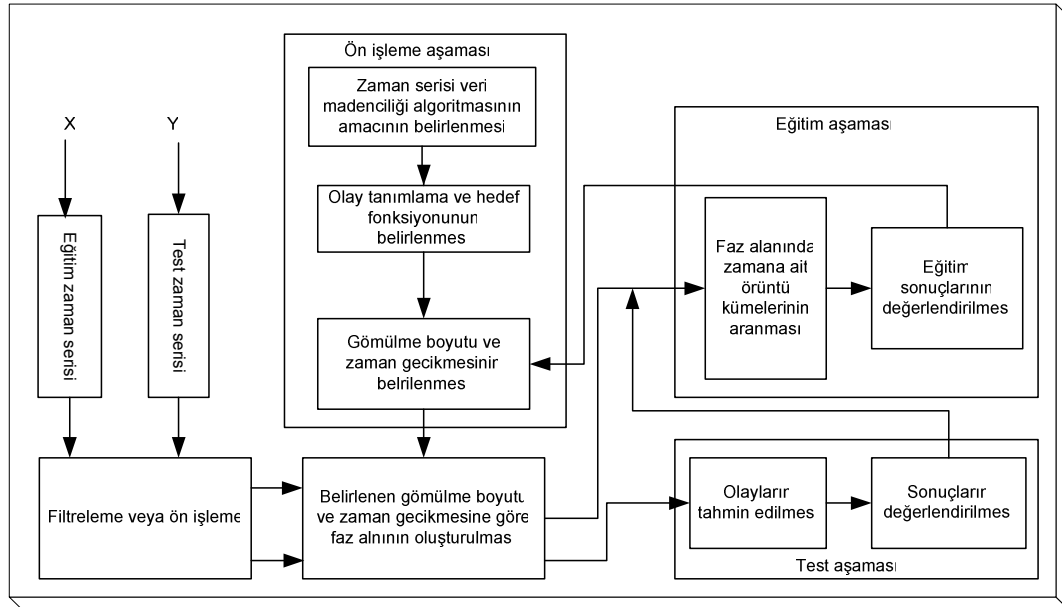
1. Eğitim Aşaması

- a. Zaman serisinden çıkarılacak bilginin belirlenmesi. Bu amaçla uygun bir olay tanımlama fonksiyonunun $g(X_t)$ 'nin tanımlanması.
- b. Hedef fonksiyonunun tanımlanması.
- c. Faz alanı için gömülme boyutu Q ve zaman gecikmesi τ 'nun seçilmesi.
- d. Faz alanında hedef fonksiyonunu maksimum yapacak optimal zamana ait örüntü kümelerinin aranması.
- e. Bulunan zamana ait örüntü kümelerinin değerlendirilmesi.
- f. Uygun sonuç bulunmamış ise d. adıma git.

2. Test Aşaması

- a. Test için kullanılacak zaman serisi için faz alanının oluşturulması.
- b. Eğitim aşamasında elde edilen zamana ait örüntü kümelerinin gelecekte oluşacak olayları tahmin edip etmeyeceğinin test edilmesi.

Test sonuçları değerlendirildikten sonra, sonuçların kabul edilebilir olup olmadığı karar verilir. Eğer kabul edilebilir sonuçlar elde edilmezse eğitim aşaması tekrarlanır. Aşağıda olay tahmini için kullanılan yukarıdaki algoritmanın blok diyagramı verilmiştir.



Şekil 4.13. Zaman serisi veri madenciliği algoritmasının blok şeması

Yukarıdaki şemada görüldüğü gibi faz alanı oluşturulmadan önce önışleme aşamasındaki adımların yapılması gerekir. Faz alanı oluşturmak için kullanılan temel bileşenler olan gömülme boyutu ve zaman gecikmesinin rasgele seçilmesi algoritmanın performansını kötü yönde etkileyebilir. Uygun bir zaman gecikmesi ve gömülme boyutunun seçilmesi gerekir. Zaman serisinin örnekleme zamanı ile karşılaştırıldığında zaman gecikmesi τ 'nın küçük seçilmesi faz alanındaki noktaların bir köşegen boyunca dağılımına neden olur. Diğer taraftan τ çok büyük seçilirse, noktalar bütün faz alanı boyunca dağılacığından zamana ait örüntü kümelerini birbirinden ayırt etmek zor olacaktır. Zaman gecikmesinin tahmini için karşılıklı bilgi ve oto korelasyon yöntemleri kullanılır. Denklem (4.10)'da verilen zaman serisi için faz alanında X_t ve $X_{t-\tau}$ gibi iki nokta arasındaki oto korelasyon aşağıdaki gibi hesaplanır.

$$C(X_t, X_{t-\tau}) = \frac{E(X_t X_{t-\tau}) - E(X_t)^2}{E\{[X_t - E(X_{t-\tau})]^2\}} \quad (4.20)$$

Otokorelasyon yönteminde, yukarıdaki denklemin ilk sifıra yaklaştığı nokta zaman gecikmesi olarak alınır. Bu denklemde X_t ve $X_{t-\tau}$ faz alanında bir noktaya karşılık gelir. Karşılıklı bilgi yöntemi ise aşağıdaki denklem ile verilebilir.

$$K(X_t, X_{t+\tau}) = \sum_{i,j} p_{ij}(\tau) \ln p_{ij}(\tau) - 2 \sum_{i,j} p_i \ln p_i \quad (4.21)$$

Burada p_i , X_t 'nin histogramda i. kutuda kalma olasılığı ve p_{ij} X_t 'nin i. kutuda kalma ve $X_{t+\tau}$ 'nin j. kutuda kalma olasılığıdır. Bu yöntemde de karşılıklı bilgi fonksiyonunun ilk minimuma ulaştığı gecikme zaman gecikmesi olarak alınır [62].

Faz alanı için minimum bir gömülme boyutunun belirlenmesi faz alanında benzersizlik için önemlidir. Bu özelliğin belirlenmesi için yanlış en yakın komşu algoritması kullanılır [64]. Faz alanındaki bütün noktalar yanlış komşudan arıncaya kadar, yanlış en yakın komşu algoritması faz alanının boyutunu adım adım arttırır. Elde edilen minimum gömülme boyutlu faz alanı orijinal zaman serisinin aynı topolojik özelliklerine sahip olacaktır. Yanlış en yakın komşu algoritması ile gömülme boyutunu aşağıdaki denkleme göre belirlenir.

$$M_n(Q)^2 = \sum_{i=0}^{b-1} [X_{n-i\tau}(Q) - X_{n-i\tau}^{EYK}(Q)]^2 \quad (4.22)$$

Burada $X_n(Q)$, Q boyutu ve τ zaman gecikmesi ile faz alanında bir noktadır. $X_n^{EYK}(Q)$, faz alanında $X_n(Q)$ noktasının en yakın komşusu ve $M_n(Q)$ ise faz alanında bir noktanın en yakın komşusu ile arasındaki mesafeyi gösterir. İki komşu noktanın birbirinden nasıl uzaklaştığı aşağıdaki denklem ile verilebilir. Bu mesafe bir denklem (4.24) 'e göre normalleştirilir.

$$M(Q+1)^2 - M(Q)^2 = [X_{n-b\tau}(Q) - X_{n-b\tau}^{EYK}]^2 \quad (4.23)$$

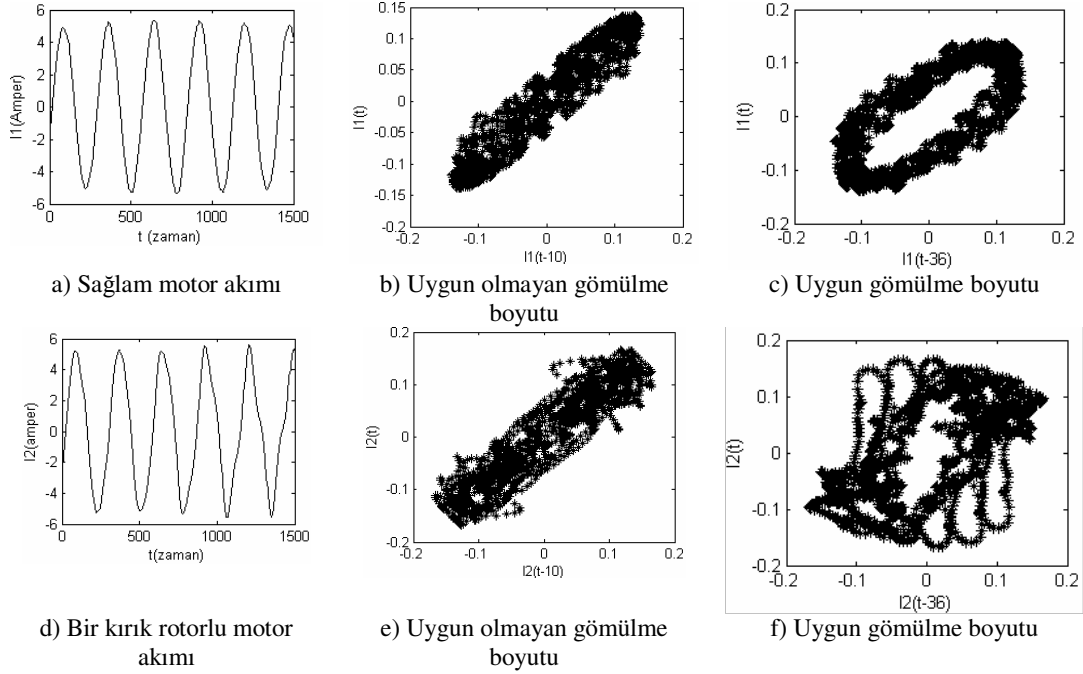
$$r_i = \frac{|M_n(Q+1)^2 - M_n(Q)^2|}{M_n(Q)} \quad (4.24)$$

Denklem (4.24)'teki r_i değeri, verilen bir v eşik değerini aştığında $X_{n-i\tau}(Q)$ yanlış bir en yakın komşuya sahip olarak işaretlenir. Yanlış en yakın komşuların yüzdesi aşağıdaki denklem ile hesaplanır.

$$\left(\frac{1}{N - (Q-1)\tau} \right) \sum_{n=1+(b-1)\tau}^N \left(\frac{|M_n(Q+1)^2 - M_n(Q)^2|}{M_n(Q)} - r_i \right) \quad (4.25)$$

Denklem (4.25)'te N sinyalin veri noktaları sayısını gösterir. Denklemde elde edilen oran, 0.001 ile 0.01 arasında değişen bir eşik değeri ile karşılaştırılır. Karşılaştırma şartı gerçekleştiğinde uygun gömülme boyutu bulunur. Faz alanı için zaman gecikmesi ve gömülme boyutu belirlendikten sonra, bu alanda veri madenciliği teknikleri kullanılarak zaman serileri analiz edilir.

Faz alanının problemi en iyi şekilde temsil etmesi ve yanlış sonuçlardan kaçınılması için gömülme boyutu büyük öneme sahiptir. Aşağıda bu durumu gösteren bir grafik verilmiştir.



Şekil 4.14 Farklı gömülme boyutları için iki zaman serisi örneği.

Yukarıda sağlam ve bir kırık rotorlu aynı tip motorun akım grafikleri ve rasgele seçilmiş uygun olmayan bir gömülme boyutu ile oluşturulmuş faz alanları ve karşılıklı bilgi ile gömülme boyutu belirlenerek oluşturulmuş faz alanı grafikleri verilmiştir. Normalde zaman serileri birbirine çok benzemektedir. Dolayısıyla daha önce anlatılan benzerlik araması yöntemi iki seriyi benzer olarak alacaktır. Fakat normalde motorda oluşan bir arıza durumu vardır. Zaman serileri veri madenciliğinde önemli olan bu küçük değişimlerin de bulunmasıdır. Şeklin (b) ve (e) seçeneklerinde her iki durum için rasgele küçük bir gömülme boyutu, (c) ve (f) seçeneklerinde ise karşılıklı bilgi yöntemi ile elde edilen bir gömülme boyutu seçilmiştir. Gömülme boyutu rasgele seçilerek oluşturulan iki faz alanına bakıldığında zaman serilerini ayırt edecek bir yöntemin geliştirilmesi zordur. Fakat (c) ve (f) seçeneklerine bakıldığında faz alanının noktalarının dağılımı iki zaman serisinde oldukça farklı bir görünüm vermektedir. Böylece zaman serileri veri madenciliğinin en önemli konusu faz alanının oluşturulması olmuştur.

5. SİMÜLASYON VE DENEYSEL SONUÇLAR

Deneysel ve simülasyon çalışmaları ile ilgili verileri elde etmek için, tek fazlı ve üç fazlı olmak üzere iki farklı asenkron motor tipi kullanılmıştır. Tek fazlı asenkron motorda simülasyon ile kısa devre olmuş sarım arızaları, mil yatağı sürtünmesi, kırık rotor ve eksantriklik arızaları oluşturulmuştur. Deneysel veriler için iki adet üç fazlı asenkron motor kullanılmıştır. Verileri bilgisayara aktarmak için Advantech'in 100 Khz'lik veri toplama kartı kullanılmıştır. İlk adımda, referans olarak farklı çalışma hızlarında sağlam motor verileri alınmış ve daha sonra ikinci motorda sırasıyla bir kırık, iki kırık ve üç kırık rotor arızaları oluşturularak veriler alınmıştır. Simülasyon ve deneysel olarak alınan veriler ayrı ayrı algoritmalarda kullanıldığı gibi iki veri tipi birleştirilerek de kullanılmıştır. Yumuşak hesaplama ve zaman serileri veri madenciliği yöntemleri kullanılarak gerçekleştirilen dört farklı arıza teşhis algoritması önerilmiştir. Aşağıda bu uygulama çalışmalarının her biri sonuçları ile detaylı olarak verilmiştir.

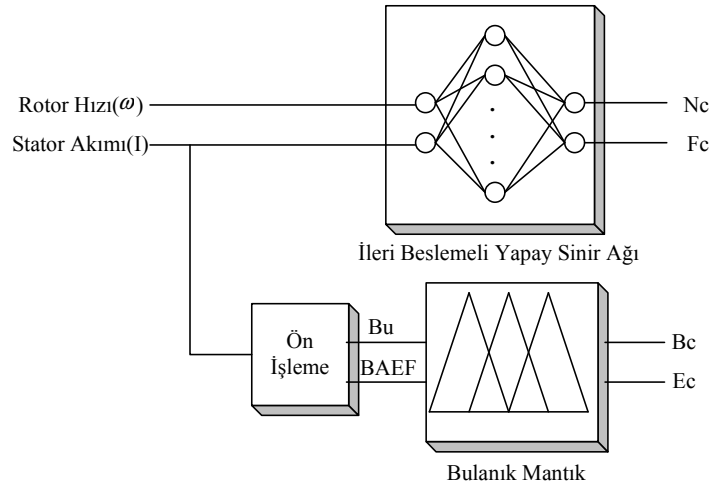
5.1. Uygulama Çalışması 1

Bu uygulama çalışmasında; tek fazlı sincap kafes asenkron motorların sarım, sürtünme, hava boşluğu eksantrikliği ve kırık rotor çubuğu arızalarının teşhisinin gerçekleştirilmesi için, yumuşak hesaplama tabanlı bir yaklaşım sunulmuştur. Sarım ve sürtünme durumlarının sınıflandırılması için ileri beslemeli yapay sinir ağı kullanılmıştır. Hava boşluğu eksantrikliği ve kırık rotor arızalarının belirlenmesi için, bulanık mantık kullanılmıştır. Tek fazlı asenkron motorun stator akımı ve rotor hızı bilgileri kullanılarak yapay sinir ağının girişleri, stator akımının Fourier dönüşümleri kullanılarak bulanık mantık modelinin girişleri oluşturulmuştur. Sağlam ve arızalı motor verileri ile önerilen yumuşak hesaplama tabanlı arıza teşhis yaklaşımının performansı gerçekleştirilen matlab programı ile doğrulanmıştır.

5.1.1. Arıza Teşhisi İçin Önerilen Yumuşak Hesaplama Tabanlı Yaklaşım

Bir asenkron motorun sarım ve sürtünme durumundaki kötüleşmeler, rotor hızı ve stator akımını etkilemektedir. Bu arızalar oluştuğunda, motorun stator akımının efektif değeri ve rotor hızı arızaya bağlı olarak azalıp artabilir. Aradaki bu ilişki, lineer olmadığından bu tür arızaları tespit etmek amacıyla ileri beslemeli yapay sinir ağı arıza teşhis şemamızın ilk modülünü oluşturacaktır. Buna karşın kırık rotor ve eksantriklik arızaları direkt olarak stator akımının efektif değeri ve rotor hızından tespit edilemez. Stator akımının bir ön işlemeye tabi tutulması

gerekir. Bu önışleme ile stator akımının hızlı fourier dönüşümü alınarak motor akımı frekans spektrumuna dönüştürülecektir. Frekans spektrumunda, hat frekansının kırık rotor ve eksantriklik ile ilgili yan bantları elde edilecektir. Bu yan bant genliklerinin heuristik yorumlanması ve arıza teşhisinin yapılabilmesi için bulanık mantık ikinci modülümüz olacaktır. Şekil 5.1’de kurulacak sistemin şeması gösterilmiştir. Girişler stator faz akımı ve ortalama rotor hızıdır. Burada N_C sarım, F_C sürtünme, B_C kırık rotor, E_C eksantriklik arızaları için çıkışları göstermektedir.



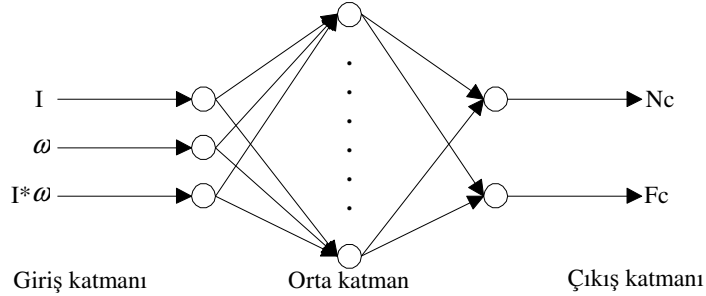
Şekil 5.1. Yumuşak hesaplama tabanlı arıza teşhis şeması

5.1.1.1. İleri Beslemeli Yapay Sinir Ağı Modülü

Bir asenkron motor ile ilgili matematiksel denklemler, yüksek dereceden lineer olmayan davranışlar gösterdiğinden motorun matematiksel modelinin oluşturulması ve buna dayalı olarak arıza teşhisinin yapılması oldukça zordur. Ayrıca motor belli bir süre çalıştıktan sonra çalıştığı ortamdan dolayı fiziksel yapısı bozulacaktır ve matematiksel model motorun durumunu tam olarak ifade etmeyecektir.

Yapay sinir ağları motorun matematiksel modeline ihtiyaç duymadan motor parametreleri arasındaki bir ilişkiyi öğrenebilir. Bu çalışmada, üç katmanlı ileri beslemeli yapay sinir ağı modeli kullanılmıştır. Yapay sinir ağının girişlerini stator akımının efektif değeri, rotor hızı ve yapay sinir ağının daha hızlı öğrenebilmesi için bu iki değerin çarpımı oluşturmaktadır. Bu girişler kullanılarak asenkron motordaki sarım (N_C) ve sürtünme (F_C) arızaları teşhis edilecektir. Yapay sinir ağının girişleri [0,1] aralığında normalleştirilmiştir. Yapay sinir ağı lineer olmayan sistemi öğrendikten sonra en yakın komşu algoritması kullanılarak bu sürekli değerler [0.1 0.5 0.9] ayırık değerlere dönüştürülmüştür. Bu değerler sarım ve sürtünme

durumları için sırayla kötü, orta ve iyi değerlerine karşılık gelmektedir. Aşağıda şekil 5.2 'de arıza teşhisi için oluşturulan yapay sinir ağı verilmiştir. Burada I değeri stator akımını ve ω ise rotor hızıdır.



Şekil 5.2. Arıza teşhisi için kullanılan yapay sinir ağı modeli

5.1.1.2. Bulanık Mantık Modülü

Asenkron motor arızalarının teşhisinde, bulanık mantık ile uzman bilgilerinden kurallar çıkararak arızalar belirlenebilir. Bu çalışmada kırık rotor ve eksantriklik ile ilgili arızalar bulanık mantık ile teşhis edilmeye çalışılmıştır. Bulanık mantığın girişlerini oluşturmak amacıyla statorun faz akımı bir ön işlemeye tabii tutulmuştur. Motorun stator faz akımı denklem (5.1) 'de Hızlı Fourier Dönüşümü kullanılarak frekans spektrumuna dönüştürülmüştür. Bu denklemde $0 \leq n \leq N-1$, $0 \leq \omega \leq N-1$, $0 \leq k \leq N-1$ ve $\omega_k = 2\pi k / N$ olur.

$$X(e^{j\omega}) = \sum_{n=0}^{N-1} x[n]e^{-j\omega n} \quad (5.1)$$

Akım sinyalleri kontrol ve durum izleme amacıyla kolayca elde edilebilir. Buradaki problem farklı motor durumlarını ayırmak ve akım sinyallerinden tam özelliklerin çıkarılmasındaki zorluklardır. Motor akım spektrum analizi, kırık rotor ve eksantriklik arızalarını tespit etmek için en çok kullanılan parametrik olmayan spektrum analiz metodudur. Bu spektrumda, denklem (5.2) ile kırık rotor hatalarının etkilediği frekanslar elde edilecektir. Bu frekanslar hat frekansının yan bandı olarak ta bilinir.

$$f_b = (1 \pm 2s)f_s \quad (5.2)$$

Burada f_b kırık rotor ile ilgili yan bantları, s motor kayması ve f_s hat frekansını gösterir. Denklem (5.2) 'deki kayma (s), motorun senkron hızı (n_s) ve mekaniksel hızı (n_m) kullanılarak denklem (5.3) 'teki gibi hesaplanabilir.

$$s = \frac{n_s - n_m}{n_s} \quad (5.3)$$

Motorun senkron hızı denklem (5.4) 'te gösterildiği gibi hat frekansı ile ilişkilidir. Burada P motorun kutup sayısını göstermektedir.

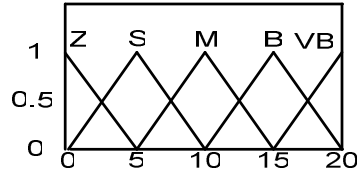
$$n_s = \frac{120f_s}{P} \quad (5.4)$$

Asenkron motorlarda oluşan diğer bir arıza türü ise hava boşluğu eksantrikliğidir. Pratikte dinamik ve statik eksantriklik daima ani olarak oluşur. Fakat çoğu zaman ikisi aynı anda oluştuğundan, bu çalışmada sadece statik eksantriklik arızaları teşhis edilmiştir.

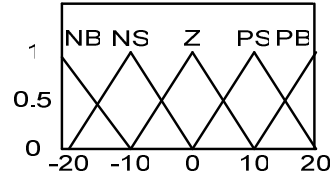
$$f_e = RS * CH \pm nf_s \quad n = 1,3,5.... \quad (5.5)$$

Eksantriklik arızalarını teşhis etmek amacıyla aynı spektrumdaki eksantriklik ile ilgili yan bantlar incelenmiştir. Denklem (5.5) 'te, RS rotor sayısı ve CH motorun çalışma hızının frekansdır. Eksantriklik arızalarının tespiti için bu eşitlikte düşük yan bandın genliği kullanılmıştır.

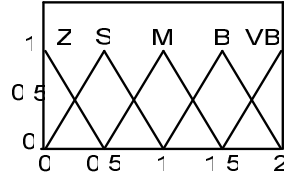
Bulanık mantık ile ilgili girişlerin ön işlemeden geçirilmesi ile bulanık mantık için ilgili yan bantlar elde edilmiştir. Bulanık mantık iki girişten ve iki çıkıştan oluşmaktadır. Bulanık mantığın ilk girişi kırık rotor üst yan bant frekansının genliğidir (B_U). İkinci giriş ise kırık rotor alt yan bandının genliğinden eksantriklik yan bandının genliği çıkarılarak oluşturulmuştur (BAE_F). Şekil 5.3 bulanık mantığın giriş ve çıkış üyelik fonksiyonları verilmiştir.



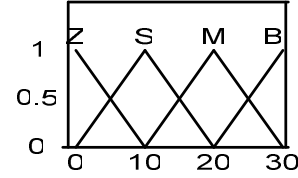
a) B_U için bulanık üyelik fonksiyonları



b) BAE_F için bulanık üyelik fonksiyonları



c) B_C için çıkış üyelik fonksiyonları



d) E_C için çıkış üyelik fonksiyonları

Şekil 5.3. Bulanık mantık modülünün giriş ve çıkış üyelik fonksiyonları

Şekil 5.3 (c) 'de, üyelik fonksiyonları kırık rotor sayısını göstermektedir. Bu şekilden de görülebileceği gibi 0 ile 2 arasındaki kırık rotor arızaları tespit edilebilir. Eksantriklik ile ilgili arızalar ise % olarak verilmiş olup %0-%30 arasındaki eksantriklik arızası tespit edilebilir. Şekil 5.4 'te bulanık modülün kural tabanı verilmiştir. Bu şekilde görüldüğü gibi çizelgenin her bir hücresindeki üst kısım eksantriklik ile ilgili (E_C) çıkışları, alt kısım ise kırık rotor ile ilgili (B_C) çıkışları gösterir.

Bu	Z	S	M	B	VB
BAEF	E_C B_C	B	B	B	B
NB	Z	S	S	M	B
NS	M	M	M	M	M
Z	Z	S	M	B	B
PS	Z	S	S	S	S
PB	Z	Z	Z	Z	Z

Şekil 5.4. Bulanık modül kural tabanı

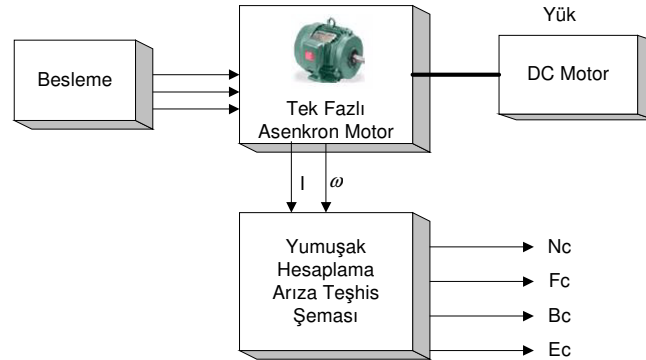
5.1.2. Simülasyon Sonuçları

Önerilen yumuşak hesaplama tabanlı arıza teşhis algoritmasını gerçekleştirmek için Matlab programı kullanılmıştır. Arıza teşhisi için kullanılan motorun parametreleri çizelge 5.1’de verilmiştir.

Çizelge 5.1. Tek fazlı sincap kafes bir asenkron motor için parametre değerleri

Özellik	Değer
Hp	¼
Gerilim v_s	110
Rotor Sayısı	36
Frekans(Hz)	60
Stator Rezistansı r_s (Ohm)	2.02
X_{ls} (ohms)	2.79
X_M (ohms)	66.8
X_{lr} (ohms)	2.12
r_r (ohms)	4.12
J (kg m ²)	0.015

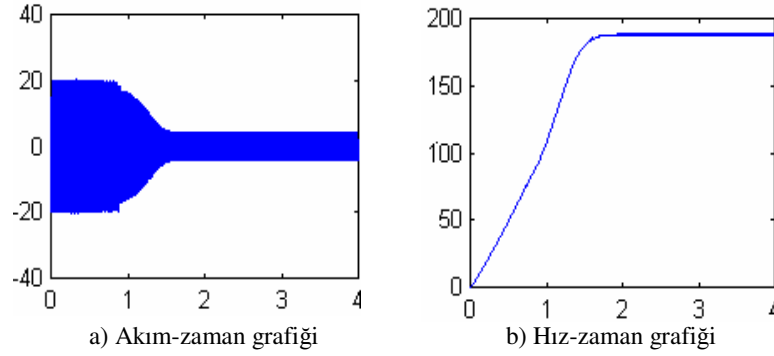
Çalışmada kullanılan motorun kararlı durumda ve sabit bir yük momentini ile bilinen ortam sıcaklıklarında çalıştığı varsayılacaktır. Şekil 5.5’te arıza teşhisinin deney düzeneği verilmiştir.



Şekil 5.5. Arıza teşhis şemasının deney düzeneği

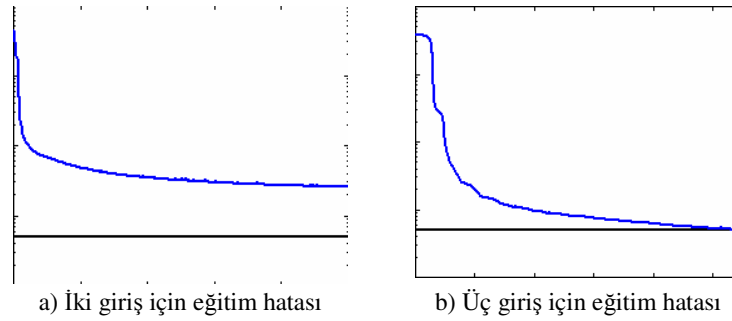
Arıza teşhisinin ilk modülünü oluşturan üç katmanlı ileri beslemeli yapay sinir ağı geriye yayılım öğrenme algoritması ile eğitilmiştir. Öğrenme oranı 0.1 ve momentum katsayısı 0.7 seçilmiştir. Gizli katmanda 10 düğüm kullanılmıştır. Yapay sinir ağı Matlab’da traingdx (eğim düşme fonksiyonu) ile eğitilmiştir. Bu eğitimde ortalama hata 0.0005 olarak seçilmiştir. Bu hata değerine yapay sinir ağı 50000 iterasyon sonunda ulaşmaktadır. Eğim düşme algoritması kullanıldığından yapay sinir ağı daha hızlı öğrenmektedir. Şekil 5.6 ’da yapay sinir

ağının girişlerini oluşturan akım ve hız değerlerinden bir durum için akım ve hızın zamana göre grafikleri verilmiştir. Bu eğitim verisinden stator akımı rotor ve eksantriklik arızaları için bulanık modülünde girişini oluşturmaktadır.



Şekil 5.6. Yapay sinir ağı için eğitim verisi örneği

Başlangıçta yapay sinir ağına girişleri olarak, stator akımı ve rotor hızı seçilmiştir. Buna göre ağı eğitim hatasının grafiği şekil 5.7 (a) 'daki gibidir. Yapay sinir ağına daha hızlı öğrenebilmesi için rotor hızı ile stator'un efektif akım değerinin çarpımı yapay sinir ağına diğer bir girişini oluşturmuştur. Diğer bütün parametreler aynı olmak koşuluyla yapay sinir ağı tekrar eğitilmiş ve daha iyi sonuçlar alındığı gözlemlenmiştir. Şekil 5.7 (b) 'de bu yapay sinir ağına hata grafiği verilmiştir.



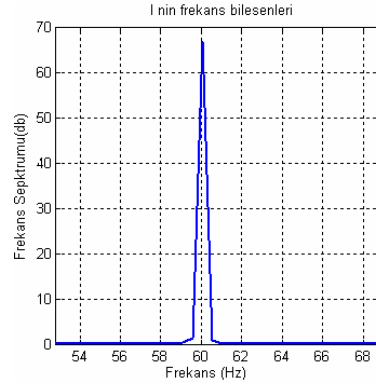
Şekil 5.7. Yapay sinir ağı için eğitim hatası grafikleri

Yapay sinir ağı eğitildikten sonra, çizelge 5.2 'de verilen akım ve hız değerlerine göre test edilmiş ve buna göre sarım ve sürtünme ile ilgili sonuçlar alınmıştır.

Çizelge 5.2. Yapay sinir ağı test sonuçları

I(A)	ω (rad/sec)	N _C	F _C
3.34	170.45	0.9	0.1
5.67	171.35	0.9	0.1
8.40	180.20	0.9	0.1
6.38	190.10	0.1	0.9
4.77	167.15	0.9	0.1
10.15	188.70	0.5	0.5
2.23	175.30	0.9	0.5
8.67	178.45	0.9	0.1

Arıza teşhis şemasının ikinci kısmını oluşturan bulanık mantık modülü kullanılarak, kırık rotor ve eksantriklik arızalarını bulmak amaçlanmıştır. Bu amaçla stator faz akımının frekans spektrumu incelenmiştir. Kırık rotor ve eksantriklik arızalarını bulmak amacıyla yan bantlar değerlendirilmiştir. Böylece aynı spektrumda hem kırık rotor hem de eksantriklik arızaları aynı anda tespit edilebilmiştir. Şekil 5.8 'de sağlam bir motora ait akım spektrumu gösterilmiştir.

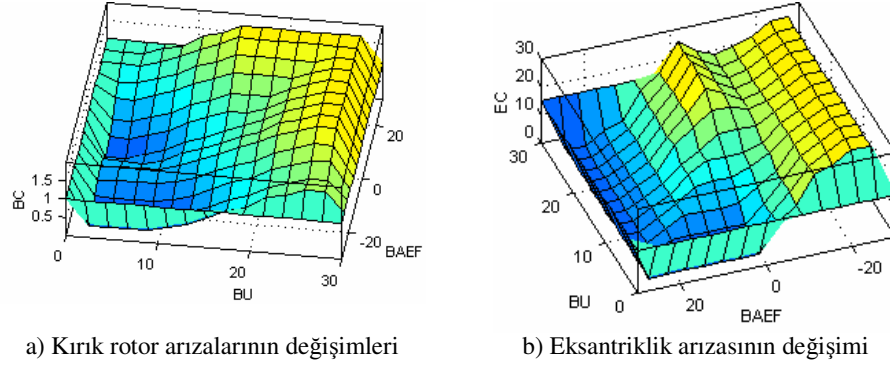


Şekil 5.8. Sağlam bir motorun akım spektrumu

Mamdani tipi bulanık mantık kullanılmış ve durulandırma için centroid metodu kullanılmıştır. AND için min ve OR için max operatörü seçilmiştir. Kırık rotor arızaları düşük olduğunda, kırık rotor ve eksantriklik arasındaki etkileşim ile üretilen bileşenlerin değeri çok küçüktür. Bu bilgiler ışığında bulanık mantık kullanılarak ilgili kurallar çıkarılmıştır. Toplam 25 kuraldan aşağıda birkaç kural verilmiştir:

1. IF B_u is M and BEA_F is PS THEN B_C is B and E_C is S
2. IF B_u is B and BE_F is PB THEN B_C is VB and E_C is Z
3. IF B_u is S and BE_F is NS THEN B_C is S and E_C is M

Aşağıda şekil 5.9'da kırık rotor ve eksantrikliğin bu girişlere göre değişimi verilmiştir.



Şekil 5.9. Girişlere göre arızaların değişimi

Şekil 5.9 (a) 'da görüldüğü gibi kırık rotor ile ilgili arızalar, kırık rotor üst yan bandı maksimum olduğunda ve fark maksimum olduğunda hata maksimum olmaktadır. Şekil 5.9 (b)'de fark değeri sıfır civarında ve üst yan bant maksimum olduğunda eksantriklik değeri maksimum olmaktadır. Çizelge 5.3 'te bulanık modül ile ilgili simülasyon sonuçları verilmiştir.

Çizelge 5.3. Bulanık mantık simülasyon sonuçları

B_U	BAE_F	BC	$EC(\%)$
1.08	-27.8	0.0	30
15.2	-11.3	0.9	22.3
6.51	19.05	1.13	0.7
27.1	-7.59	2	23.1
3.25	0.25	0.02	0.001
28.9	-6.14	2	25

Kırık rotor arızaları oluştuğunda, hat frekansının düşük ve yüksek yan bantlarının genlikleri değişecektir. Eksantriklik ile ilgili yan bandın genliği de oluşan eksantriklik değerine göre değişecektir.

Asenkron motorlar, endüstride en çok kullanılan motor tipleri olduğundan bu motorlarda oluşabilecek arızaların erken bir aşamada ve sistem tamamen arızaya uğramadan teşhis edilmesi önemli olmaktadır. Bu çalışmada tek fazlı bir asenkron motorda oluşabilecek sarım, sürtünme, kırık rotor ve eksantriklik arızaları yumuşak hesaplama teknikleri ile tespit edilmiştir. Yapay sinir ağlarının lineer olmayan sistemleri öğrenme yeteneğinden faydalanılarak, rotor hızı ve stator akımı ile sarım ve sürtünme durumu arasındaki doğrusal olmayan ilişkiyi öğrenmek için yapay sinir ağı eğitilmiştir. Bundan faydalanarak sarım ve sürtünme arızaları tespit edilmiştir. Aynı akım verileri üzerinden akım spektum analizi ile elde edilen yan bantların genliği bulanık mantık ile değerlendirilerek kırık rotor ve eksantriklik arızaları tespit edilmiştir.

5.2. Uygulama Çalışması 2

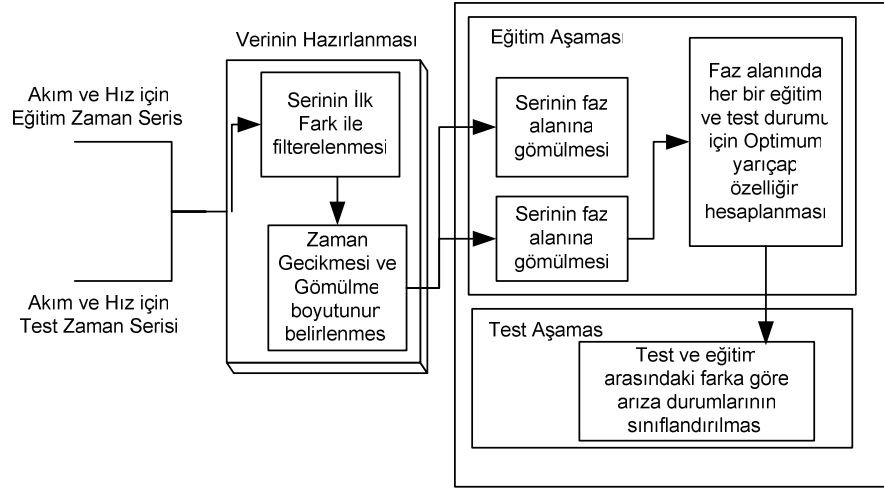
Bu uygulamada; asenkron motorlardaki sarım ve sürtünme arızalarının teşhisi için zaman serileri veri madenciliğine dayalı bir yöntem sunulmuştur. Sarım ve sürtünme durumundaki arızaları belirlemek için stator akımı ve rotor hızı bilgileri kullanılmıştır. Bu iki verinin çarpımı, ilk fark zaman serisi metodu ile filtrelenmiştir. Elde edilen yeni zaman serisi, lineer olmayan zaman serisi metotlarından zaman gecikmesine dayalı gömülme yöntemi kullanılarak faz alanına dönüştürülmüştür. Faz alanında zaman serisi üzerinde veri madenciliği teknikleri kullanılarak, her bir arızalı ve sağlam durum için tek olan bir yarıçap özelliği elde edilmiştir. Bu özellik kullanılarak farklı arıza durumları sınıflandırılmıştır. Arızalı ve arızasız motor verileri ile önerilen zaman serileri veri madenciliği tabanlı arıza teşhis yönteminin performansı uygulama sonuçları ile doğrulanmıştır.

5.2.1. Arıza Teşhisi için Önerilen Zaman Serileri Veri Madenciliği Yaklaşımı

Sarım ve sürtünme arızalarının belirlenmesi için motorun stator akımı ve rotor hızı giriş olarak alınmıştır. Bu arızalar, stator akımı veya rotor hızının efektif değerinde değişmelere sebep olduğundan kararlı durumdaki zamana ait iki grafiğin çarpımı faz alanı için girişi oluşturmuştur. Uygun bir filtrelemeyi sağlamak için elde edilen zaman serisinin ilk fark filtrelemesi denklem (5.6)'da hesaplanmıştır.

$$\Delta i(t) = i(t) - i(t-1) \quad (5.6)$$

Farklı sarım ve sürtünme arızalarını belirlemek amacıyla önerilen yaklaşım yöntemi şekil 5.10'da verilmiştir.



Şekil 5.10. Önerilen zaman serileri veri madenciliği arıza teşhis yönteminin blok diyagramı

Sarım ve sürtünme arızalarını belirlemek amacıyla denklem (5.6)'daki Δi , zaman serisinin faz alanına dönüştürülmesi için ilk adımdır. Zaman gecikmesinin belirlenmesi için dördüncü bölümde bahsedilen karşılıklı bilgi yöntemi kullanılmış ve zaman gecikmesi 38 olarak hesaplanmıştır. Faz alanının gömülme boyutu 2 olarak alınmıştır. Asenkron motorun her bir arızalı ve sağlam durumun grafiklerinden şeklin merkez noktaları, denklem (5.7)'deki formüle göre hesaplanmıştır. Faz alanındaki diğer noktaların merkez noktaya olan uzaklıklarının ortalamasının hesaplanması için öklid mesafesi kullanılmıştır. Bu özellik denklem (5.8)'de verilmiştir. Her bir arıza ve sağlam motor durumlarında faz alanındaki dönme yarıçapını hesaplamak için denklem (5.9) kullanılmıştır.

$$m_1 = \frac{\sum_{t=1}^N i_t}{N-1} \quad (5.7)$$

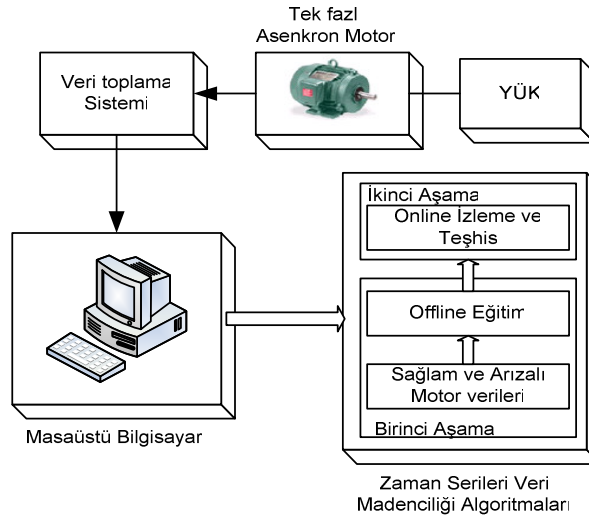
$$d(t)^2 = (I_t - m_0)^2 + (I_{t-1} - m_1)^2 \quad (5.8)$$

$$r = \sqrt{\frac{\sum_{t=1}^N d(t)^2}{N}} \quad (5.9)$$

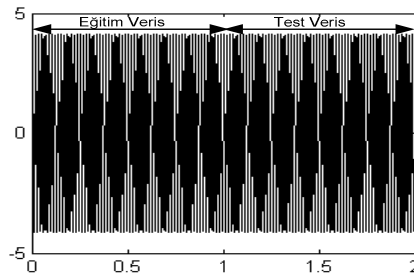
Denklem (5.8)'deki m_0 zaman gecikmesi sıfır olduğunda ve m_1 zaman gecikmesi l olduğunda hesaplanan değerlerdir. Sonuçta faz alanımız iki boyutlu olduğundan m_0 x-koordinatını ve m_1 y-koordinatını gösterir. I_t faz alanında x-koordinatında bir noktayı, I_{t-l} y-koordinatında bir noktayı göstermektedir.

5.2.2. Uygulama Sonuçları

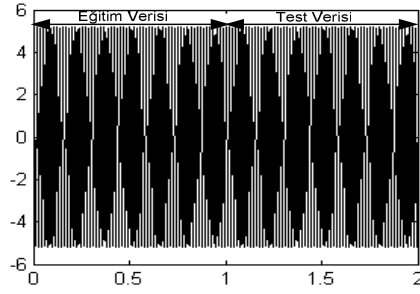
Önerilen zaman serileri veri madenciliği algoritmasını gerçekleştirmek için matlab programı kullanılmıştır. Deneyde tek fazlı sincap kafes asenkron motor kullanılmıştır. Kullanılan motorun parametreleri çizelge 5.1’de verilmiştir. Deneyde her bir arıza durumu için akım ve hızın kararlı durumdaki ilk 1000 örneği eğitim için geriye kalan 1000 örneği test verisi olarak kullanılmıştır. Aşağıda şekil 5.11’de arıza teşhisinin deney düzeneği verilmiştir. Arızalı ve sağlam motorun eğitim ve test veri kümesi için kullanılan akım grafiklerinden ikisi şekil 5.12 ve şekil 5.13’te verilmiştir.



Şekil 5.11. Arıza teşhis yönteminin deney düzeneği

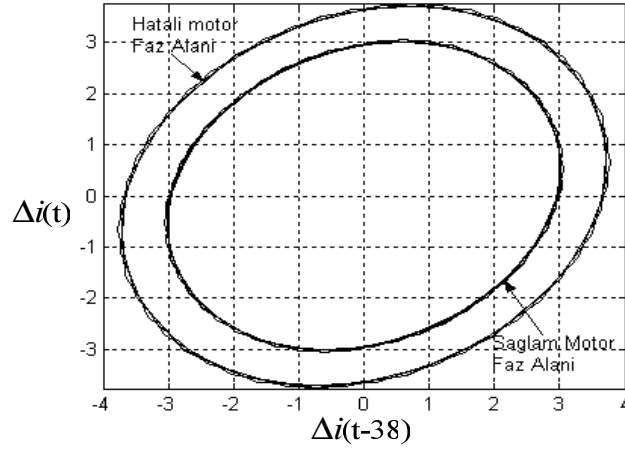


Şekil 5.12. Sağlam motor için akımın eğitim ve test verisi



Şekil 5.13. Arızalı motor için akımın eğitim ve test verisi

Hız zaman verisi 0 ile 4 arasında normalleştirildikten sonra, akım verisi ile çarpılarak ilk fark yöntemi ile filtrelenmiştir. Şekil 5.14'te sağlam ve arızalı bir motorun faz alanı grafiği verilmiştir.



Şekil 5.14. Motorun sağlam ve arızalı durumu için faz alanı grafikleri

Eğitim ve test yarıçaplarının arasındaki fark çizelge 5.4'te verilmiştir.

Çizelge 5.4. Eğitim ve test verileri için yarıçap farkları

		EĞİTİM VERİSİ				
TEST VERİSİ		N100B0	N100B12	N100B20	N90B0	N90B20
	N100B0	0.0001	0.1029	0.3605	0.7071	0.9540
	N100B12	0.1100	0.0072	0.2142	0.5970	0.8439
	N100B20	0.3604	0.2576	0.0036	0.3466	0.5965
	N90B0	0.7072	0.6043	0.3829	0.0001	0.3392
	N90B20	0.9570	0.8541	0.6327	0.2516	0.0030

Çizelge 5.4'te, N sarım durumunu ifade etmekte olup elimizdeki sağlam motor için değeri 100'dür ve B sürtünme durumunu ifade edip 0 ile 0.0020 arasında değişmektedir. Her bir arıza değeri için elde edilen yarıçap değerlerine göre arıza durumu sınıflandırılmıştır. Aynı arıza durumunun eğitim ve test verisi arasındaki yarıçap farkı, farklı durumlar için elde edilen fark ile karşılaştırıldığında oldukça küçük olmaktadır. Elde edilen yarıçap değerleri birbirine çok yakın arıza değerleri için yakın değerlere sahip olmaktadır. Bu yüzden eğitim ve test verileri sadece çizelge 5.4'te verilenlerle sınırlı tutulmamalıdır. Önerilen yöntemin avantajlarından biri, motor dinamikleri hakkında çok fazla bilgiye sahip olmaya gerek kalmadan sadece akım ve hız değerleri bilinerek bu değerler üzerinden arıza teşhisi yapılabilmesidir. Yöntemin getirmiş olduğu bir avantajda böyle bir arıza teşhis yönteminin uygulanma maliyetinin ucuz olmasıdır.

5.3. Uygulama Çalışması 3

Bu çalışmada; bir asenkron motorda oluşan kırık rotor çubuğu arızalarını tespit etmek amacıyla negatif seçim algoritması ve motor akım imza analizi kullanılmıştır. Motordan elde edilen akım sinyalleri, motor akım imza analizi yöntemi ile akım spektrumuna dönüştürülmüştür. Bu spektrumda elde edilen yan bantlar negatif seçim algoritmasına giriş olarak alınmış ve kırık rotor arızaları teşhis edilmiştir. Geliştirilen arıza teşhis algoritmasının uygulanabilirliği, bir asenkron motordaki arızalar gerçek zamanlı teşhis edilerek gösterilmiştir. Ayrıca önerilen arıza teşhis yaklaşımı diğer arızaların teşhisi için uyarlanabilir.

5.3.1. Arıza Teşhisi için Önerilen Bağışıklık Tabanlı Algoritma

Yapay bağışıklık tabanlı arıza teşhis sistemi iki bileşenden oluşmaktadır. İlk bileşen, motorun üç fazından elde edilen akım sinyalleri üzerinde ön işleme yapmak için kullanılan motor akım imza analizidir. Kırık rotor arızaları tek bir faz akımı incelenerek teşhis edilebilir. Motorun A fazından alınan akım bilgileri denklem (5.1)'deki formül kullanılarak frekans spektrumuna dönüştürülmüş ve güç spektrum yoğunluğu hesaplanarak akım spektrumu oluşturulmuştur. Akım spektrumundan kırık rotor ile ilgili yan bantların elde edilmesi için denklem (5.2)'deki hesaplama formülü aşağıdaki gibi değiştirilerek kullanılmıştır.

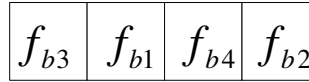
$$f_b = (1 \pm 2ks)f_h \quad k = 1, 2, \dots, N \quad (5.10)$$

Burada f_b kırık rotor ile ilgili yan bantların frekansını, f_h hat frekansını ve s ise motor kaymasını gösterir. Burada $k=1,2$ için elde edilen yan bantlar incelenmiştir. Yüksek yan bantlar

$f_{b1} = (1+2s)f_h$ ve $f_{b2} = (1+4s)f_h$, düşük yan bantlar ise $f_{b3} = (1-2s)f_h$ ve $f_{b4} = (1-4s)f_h$ olarak hesaplanır. Kayma değerinin hesaplanması için denklem (5.11)'deki gibi bir formül kullanılmıştır.

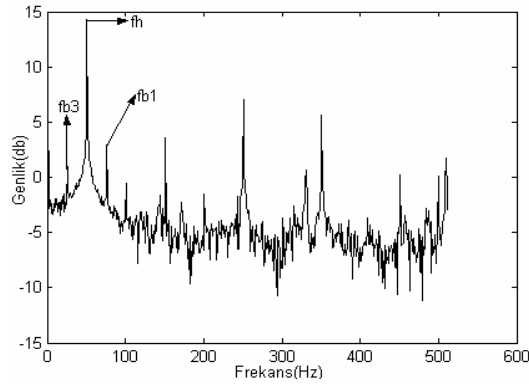
$$s = 1 - \frac{1}{N} \left[p \left(1 + \frac{f'}{f_h} \right) \right] \quad (5.11)$$

Burada N rotor sayısını, p motorun çift kutup sayısını ve f' ise ilk slot harmoniğinin frekansını gösterir. Motor akım imza analizi ile akım spektrumu oluşturulduktan ve kırık rotor çubuğu ile ilgili yan bantlar hesaplandıktan sonra negatif seçim algoritması için gerekli girişler oluşturulmalıdır. Kırık rotor çubuğu arızalarında f_{b1} ve f_{b3} daha etkilidir. Diğer iki yan bant kırık rotor arızaları oluştuğunda daha az değişmektedirler. Veriler [0,1] aralığında normalleştirilmiştir ve reel kodlama kullanılmıştır. Şekil 5.15'te kodlama verilmiştir.



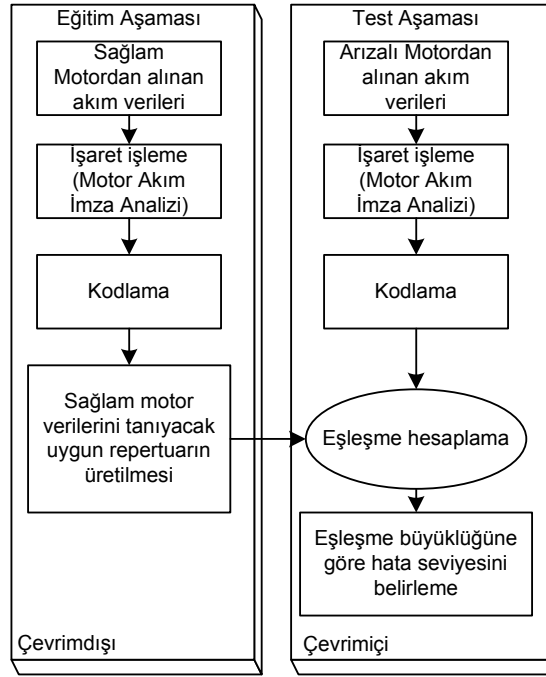
Şekil 5.15. Negatif seçim için kodlama

Akım spektrumu ile ilgili bir örnek şekil 5.16'da gösterilmiştir. Şekilden de görüldüğü gibi f_h hat frekansını olup bu motor için 50 Hz 'dir. Hat frekansının sağında olan yan bantlar yüksek ve solunda olan yan bantlar ise düşük yan bantlardır. Bu şekilde bir düşük (f_{b3}) ve bir yüksek (f_{b1}) yan bant gösterilmiştir.

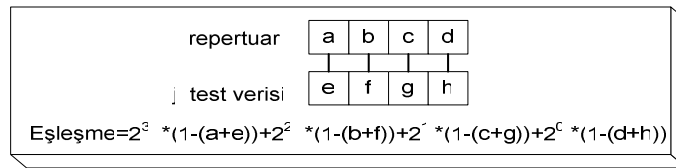


Şekil 5.16. 180 V ile beslenen bir kırık rotorlu motorun akım spektrumu

Arıza teşhis şeması, eğitim ve test olmak üzere iki kısımdan oluşur. Eğitim aşamasında, sağlam motor verilerini tanıyacak repertuarlar çevrimdışı olarak üretilir. Test aşamasında ise uygun şekilde kodlanan veriler ile bu repertuarlar eşleştirilerek arızalar teşhis edilir. Yapay bağışıklık tabanlı arıza teşhis şeması şekil 5.17’de verilmiştir. Bir veri ile repertuar arasındaki eşleştirmenin genel formülü şekil 5.18’de verilmiştir. Bu eşleşmeyi kullanarak yapılan bir hesaplama şekil 5.18’de verilmiştir.



Şekil 5.17. Yapay bağışıklık tabanlı arıza teşhis şeması



Şekil 5.18. Test ve repertuar arasındaki eşleşme denklemi

Reel kodlama kullanıldığından ve veriler [0,1] arasında normalleştirildiğinden dolayı sağlam bir motor ile ilgili verileri tanıyacak repertuarın üretilmesi için her bir hücrenin eşleniği alınmıştır.

a	0 36801	0 45626	0 2365	0 61296
t	0 5835	0 60244	0 74543	0 19002
$\text{Eşleşme} = \text{abs}(1 - (a(1) + t(1)))^2 * 2 + (\text{abs}(1 - (a(2) + t(2))))^2 * 2 + (\text{abs}(1 - \text{abs}(a(3) + t(3))))^2 + (\text{abs}(1 - (a(4) + t(4))))$				
Eşleşme=0 895				

Şekil 5.19. Örnek bir eşleştirme

Şekil 5.19'daki örnekte, kullanılan abs fonksiyonu mutlak değeri ifade etmektedir. Böyle bir eşleştirme kullanmanın getirmiş olduğu en önemli avantaj, hat frekansına yakın olan yan bantlar arızası belirlemede daha etkin olmasıdır. Ayrıca reel kodlama kullanıldığından keskin bir geçiş söz konusu değildir. Eşleştirme sonucu sıfıra yakın olması, normal sistem verilerinin geldiğini gösterir. Sıfırdan uzaklaştıkça arızanın derecesi artar.

5.3.2. Deneysel Verilerin Elde Edilmesi

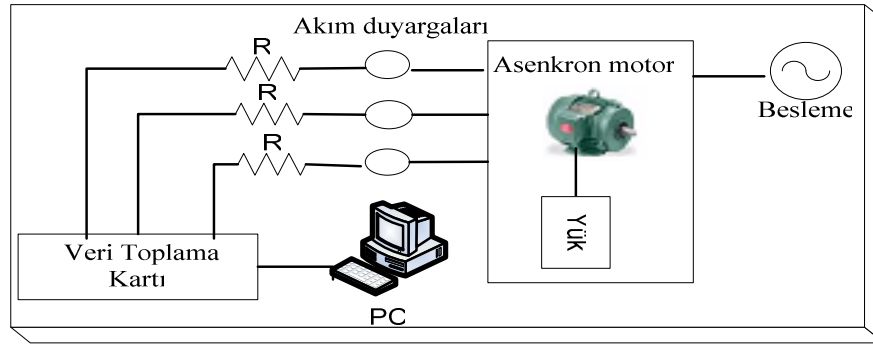
Gerçek bir asenkron motordan elde edilecek verileri kullanarak arızaları teşhis etmek için bir deney düzeneği oluşturulmuştur. Deney düzeneğinde üç fazlı asenkron motorlar kullanılmıştır. Kullanılan asenkron motorların özellikleri çizelge 5.5'te verilmiştir. Motorun faz akımları üç adet akım duyargası kullanılarak ölçülmüştür. Akım duyargalarından geçirilen faz akım bilgileri üç ayrı direnç üzerinden veri toplama kartının analog girişlerine verilmiştir. Şekil 5.20'de kurulan deney düzeneği verilmiştir. Kurulan deney düzeneğinin şeması şekil 5.21'de görülmektedir.

Çizelge 5.5.. Deneyde kullanılan asenkron motorun özellikleri

Özellik	Değer
Güç	0.37 Kw
Giriş gerilimi	380 V
Tam yük akımı	1.2 A
Frekans	50 Hz
Kutup sayısı	4
Rotor slot sayısı	22
Tam yükte çalışma hızı	1390 Rpm



Şekil 5.20. Arıza teşhisi için kurulan deney düzeneği.



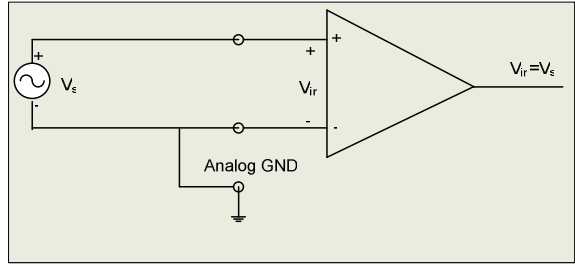
Şekil 5.21. Deney düzeneği şeması

Asenkron motorlar yedi farklı hızda çalıştırılarak, akım verileri bilgisayara kaydedilmiştir. Motora verilen besleme gerilimleri 140, 180, 220, 260, 300, 340 ve 380 V olarak seçilmiştir. Motorun besleme gerilimini değiştirmek için bir trafo kullanılmıştır. Motorun her bir durumu için saniyede 1000, 3000, 5000, 8000 örnek 3 saniye boyunca alınarak bilgisayara kaydedilmiştir. Ayrıca uygulanacak algoritmaların doğruluğunu test etmek amacıyla, 8000 örnek/saniye ile her bir durum için iki örnek alınmıştır. Sağlam motordan alınan veriler referans oluşturmak amacıyla kullanılmıştır. Kırık rotor arızaları oluşturmak için bir motorun üç adet rotor çubuğu delinmiştir. Şekil 5.22’de bir çubuğu delinmiş rotor görülmektedir.



Şekil 5.22. Bir çubuğu kırk rotor

Asenkron motordan verileri elde etmek amacıyla, Advantech 'in 16 bitlik bir veri toplama kartı kullanılmıştır. Veri toplama kartı, saniyede 100.000 veri (örnekleme frekansı: 100 Khz) alabilme özelliğine sahiptir. Kartın 16 adet analog girişinden ilk üçü faz akımlarını almak için kullanılmıştır. Analog girişlere akım sinyalleri diferansiyel giriş konfigürasyonu ile bağlanmıştır. Diferansiyel giriş konfigürasyonunda her bir kanal için iki sinyal kablosu kullanılır. Bu giriş tipi iki sinyal kablosu arasındaki gerilim farkına karşılık gelir. Şekil 5.23'de bu bağlantı şekli verilmiştir.



Şekil 5.23. Veri toplama kartının analog girişine sinyalin bağlanması

Motordan üç faz akım bilgisini okumak amacıyla, Delphi'de bir ara yüz programı tasarlanmıştır. Programı yazmak için ilk olarak Activex bileşeni olan Activedaq Delphi programına eklenmiştir. Aşağıda Delphi'de yazılan programın bir ekran görüntüsü verilmiştir.

ASENKRON MOTOR VERİ TOPLAMA PROGRAMI

Aygıt Seçimi
Aygıt No: 0
Aygıt Adı: Advantech PCL 816

Özelliklerin Ayarlanması
Başlama Kanalı: 0
Kanal Sayısı: 3
Giriş Aralığı:
☒ Overall Gain
Veri Tipi: Real Data
Transfer Modu: Software Mode
Tetikleme Kaynağı: Internal
Saat Kaynağı: Internal
Örneklem Oranı: 3000 veri/sn
Veri Sayısı: 9000
FIFO Ayarları
☒ Aktif Size: 1
☒ Olay Aktivitesi ☒ Periyodik Mo
Veri Okumak İstediğiniz Kanal sayısı: 3

VERİ ALMA İŞLEMLERİ
Veri Okuma Süresi: 3 sn
Başla Bitir
Durum

A FAZ AKIMI	B FAZ AKIMI	C FAZ AKIMI
1.5017	-0.94636	-0.57561
1.4867	-1.066	-0.42364
1.4367	-1.1755	-0.26067
1.3909	-1.2781	-0.10717
1.3183	-1.3577	0.057016
1.2328	-1.424	0.19862
1.1416	-1.4609	0.34175

ALINACAK VERİ İLE İLGİLİ AÇIKLAMALAR
Örneklem oranı: 1000 örnek/sn
Süre: 3 sn

VERİ SAYISI
A fazı veri sayısı: 3000
B fazı veri sayısı: 3000
C fazı veri sayısı: 3000

VERİNİN KAYDEDİLECEĞİ DOSYA ADI
B340R1V8 MATLAB DOSYASI OLUŞTUR

Hata Mesajı
Hata Kodu
Hata Mesajı

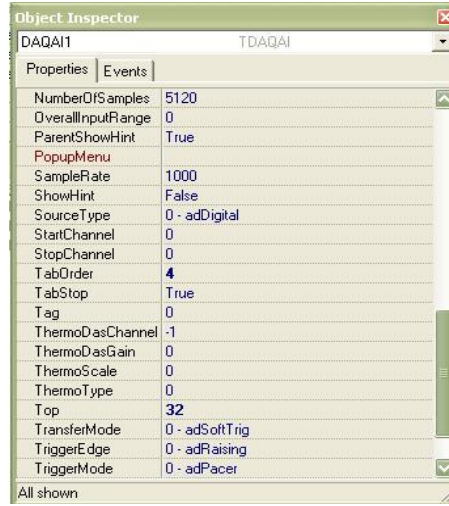
Başlama Zamanı: 12:30:43
Bitiş Zamanı: 12:30:46
Geçen Süre: 3 sn

PROGRAMI KAPAT

Şekil 5.24. Motordan akım verilerini almak için yazılan program

Şekil 5.24'ten de görüldüğü gibi, aygıt seçim butonu ile sistemde yüklü olan Advantech'in PCL-816 veri toplama kartı seçilmektedir. Hangi kanaldan başlayarak veri alınacağı işlemi başlama kanalı ile belirlenip, kanal sayısı ile toplam kaç kanaldan veri alınacağı belirlenir. Kurulan deney düzeneğinde 0., 1. ve 2. kanal veri alımı için kullanılmıştır. Verileri noktalı sayı olarak almak için, veri tipinden real data seçeneği seçilmelidir. Veri 3 farklı seçenek ile alınabilir. Kesme modu kullanıldığında, her bir örnek veri alındıktan sonra diğer verilerin alımına başlanması için veri toplama kartına bir kesme gönderilir. DMA modunda veri alınırken, veriler 1024'lük diziler halinde alınmaktadır. Biz verileri almak için software modunu kullandık. Çünkü kullanımı daha basit olup ne kadar süre veri alınacağını ayarlanması daha kolaydır. Veri toplama kartının üç kanalından alınan veriler programda bir dizeye kaydedilmektedir. Kanallardan veri dizisi alınırken, her bir kanal verisinden birer örnek alınır ve dizide ardışık sırada yazılır. Örneğin dizinin birinci elemanı 0. kanaldan gelen veriyi, ikinci elemanı 1. kanaldan gelen veriyi ve üçüncü elemanı ise 2. kanaldan gelen veriyi gösterir. Dolayısıyla üç kanaldan elde edilen veriler tek bir dizide karışık olarak bulunmaktadır. Dizideki bu veriler ayrıştırılarak üç ayrı liste kutusuna kaydedilmektedir. Örneklem oranı seçeneği saniyede kaç örnek alınacağını göstermektedir. Örneğin, saniyede 3000 örnek alınıyorsa ve 3 saniye boyunca veriler alınacaksa toplam alınacak veri sayısı 9000 olarak belirlenir. Buradaki 3000 değeri eğer tek kanal seçiliyse o kanal için örneklem oranını gösterir. Eğer üç kanal seçiliyse örneklem oranı 3'e bölünür. Dolayısıyla her bir kanal için saniyede 1000 örnek

alınacağını gösterir. Programda veri okuma süresi kutucuğu ile kaç saniye veri alınacağı belirlendikten sonra başla butonuna tıklanarak veri alımına başlanır ve belirlenen süre bittiğinde veri alımı kendiliğinden sonlanır. Belirlenen süre için veriler alındıktan sonra liste kutularındaki veriler bilgisayar ortamında kaydedilirken verilerin karışmasını önlemek amacıyla bir isimlendirme kullanılmıştır. Örneğin, sağlam motordan 380V besleme gerilimi ile alınan veriler isimlendirilirken “A380SV8a.dat” gibi bir ifade kullanılmıştır. Burada A birinci motoru, 380 besleme gerilimini, S verinin sağlam durumda alındığını, V8 verinin 8000 örnek/saniye ile alındığını ve a ise motorun hangi fazındaki veri olduğunu göstermektedir. Delphi ile analog girişlerden veri almak için kullanılan “Aidaq” Activex bileşeninin bazı özellikleri aşağıdaki şekilde verilmiştir.



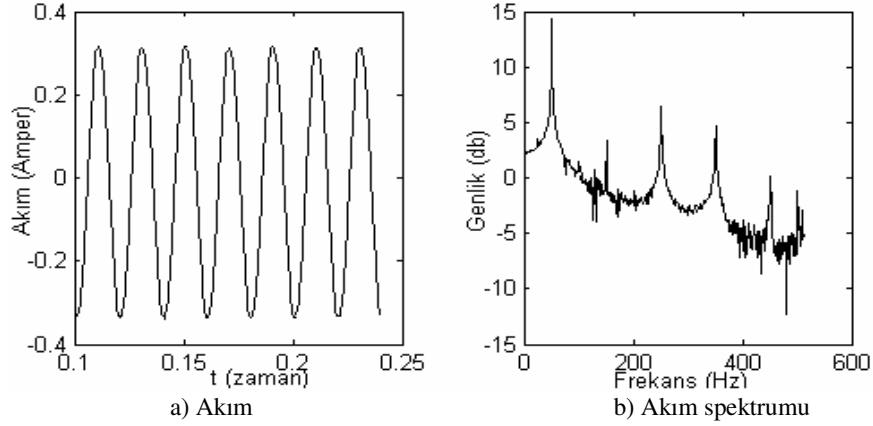
Şekil 5.25. DAQAI Activex bileşeninin bazı özellikleri

Şekil 5.25'te bazı özellikleri verilen DAQAI veri toplama kartının analog girişinden veri almak için kullanılan activex bileşenidir. Bu bileşenin özellikleri kullanılarak veri alımına hangi kanaldan başlanılacağı, veri örnekleme oranı ve veri transfer modu gibi özellikler kolayca ayarlanabilir.

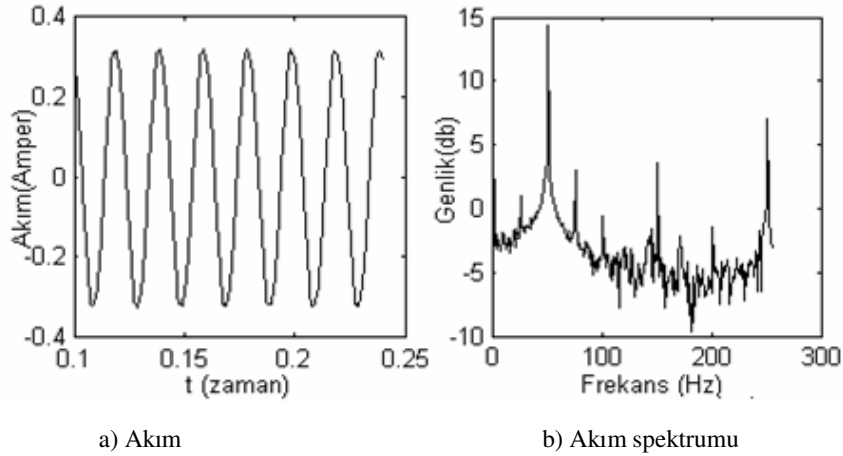
5.3.3. Deneysel Sonuçlar

Önerilen yapay bağımsızlık tabanlı arıza teşhis algoritmasını uygulamak için MATLAB programı kullanılmıştır. Analiz için üç fazlı gerçek bir asenkron motordan akım değerleri alınmıştır. Kırık rotor arızalarını oluşturmak amacıyla üç adet rotor çubuğu delinmiştir. İlk önce bir kırık rotor arızası oluşturulmuş, sonra bu rotorun yanındaki rotor çubuğu ve daha sonra

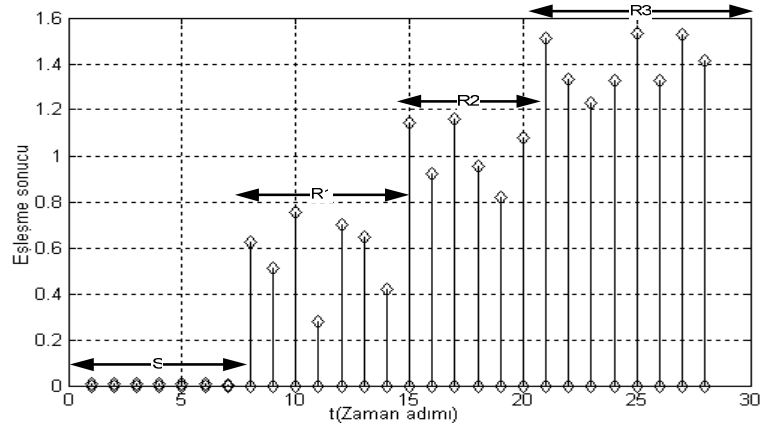
bunlardan uzakta olan bir rotor çubuğu delinmiştir. Üç faz akımından sadece a fazından elde edilen akım verileri arıza teşhisi için kullanılmıştır. Farklı hızlarda elde edilen sağlam ve arızalı motorun akım verileri, şekil 5.16'daki gibi motor akım imza analizi aracılığıyla akım spektrumuna dönüştürülmüştür. Akım spektrumları elde edildikten sonra denklem (5.10) kullanılarak sağlam motor için kırık rotor ile ilgili yan bantlar elde edilmiştir. Eğitim için kullanılan sağlam ve bir rotoru kırık motorların 180V gerilim ile beslenme durumunda akım ve spektrum grafikleri şekil 5.26 ve 5.27'de verilmiştir. Sağlam motor ile ilgili yan bantlar şekil (5.15)'te gösterilen kodlama ile kodlanarak, yapay bağışıklığın girişi için uygun hale getirilmiştir. Eğitim aşamasında sağlam motor verilerini tanıyan repertuarlar üretilmiştir. Test aşamasında sağlam motor, bir, iki ve üç kırık rotora sahip kodlanmış veriler üretilen repertuarlar ile eşleştirilmiştir. Şekil 5.28 'te test aşamasında test verileri ile repertuar arasındaki eşleşmeye göre arıza teşhisinin grafiği verilmiştir.



Şekil 5.26. 180 V ile beslenen sağlam motorun akım ve spektrum grafikleri



Şekil 5.27. 180 V ile beslenen bir kırık rotorlu motorun akım ve spektrum grafikleri



Şekil 5.28. Algoritmanın test sonuçları

Test aşamasında, hem sağlam hem de arızalı motor verileri kullanılmıştır. Şekil 5.28'den de görüldüğü gibi S ile gösterilen ilk yedi zaman adımında eşleştirme sonuçları 0'a çok yakındır. Böylece bu veriler, bağışıklık algoritması tarafından tanınan sağlam motor verileridir. S, R1, R2 ve R3 aralığındaki eşleştirmelerin ortalaması sırasıyla 0.0078, 0.5622, 1.0126 ve 1.399 olarak hesaplanmıştır. Bu bilgi arızaları sınıflandırmak için kullanılabilir. Böylece R1, R2 ve R3 ile gösterilen zaman adımı aralıkları sırasıyla bir, iki ve üç kırk ile ilgili verileri göstermektedir. Eşleştirme işlemi şekil 5.18'deki denklem kullanılarak hesaplanmıştır. Kırk rotor sayısı arttıkça eşleştirme denkleminin sonucunun arttığı görülmektedir.

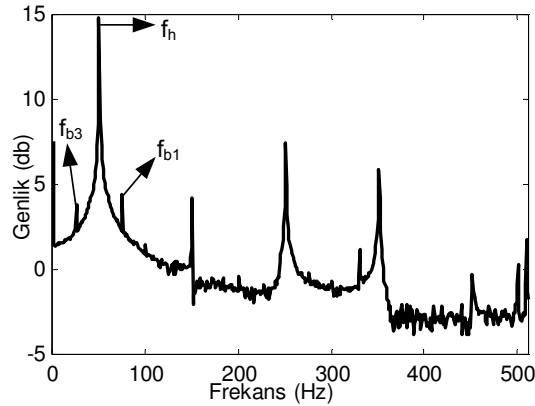
Daha önce arıza teşhisi için yapay bağışıklık sistemini kullanan çalışmalarda direkt zaman üzerinden izleme yapılmıştır. Bu çalışmalar, özel bir arızayı teşhis etmeyip sadece sistem tarafından kabul edilmeyen bir çalışma durumunu belirlerler. Motorun akımında zaman üzerinde meydana gelen düşmeler veya yükselmeler birçok sebebe bağlı olabilir. Örneğin, stator sarımları aşırı ısındığında motor akımı yükselir. Dolayısı ile akımın zaman üzerinde izlenmesi arıza teşhisi için yeterli bir yöntem değildir. Bu çalışmada motor akım imza analizi kullanılarak kırk rotor ile ilgili yan bantlar elde edilmiş ve arıza teşhisi için bu değerler kullanılmıştır. Akım spektrumunda kırk rotor ile ilgili yan bantlar arızanın şiddeti arttıkça artarlar. Bir diğer avantaj ise eşleşmede kullanılan denklemdir. Kırk rotor arızaları hat frekansına yakın frekansların genliklerini daha çok etkilerler. Bu yüzden eşleştirme denkleminde bu frekansların genliklerinin daha çok ağırlığa sahip olmaları sağlanmıştır.

5.4. Uygulama Çalışması 4

Bu çalışmada; insanın bağışıklık sisteminden esinlenilerek geliştirilen yapay bağışık sistemlerinin negatif seçim algoritması bulanık mantık ile birleştirilerek arıza teşhisi için kullanılmıştır. Önerilen algoritma ile asenkron motorlarda oluşan stator sarım arızası, mil yatağı sürtünme arızası ve kırık rotor çubuğu arızaları teşhis edebilmektedir. Bunun için giriş olarak alınan motor akım sinyalleri akım imza analiz yöntemi ile akım spektrumuna dönüştürülerek negatif seçim algoritması uygulanmıştır. Bu işlemin sonucu ile akım spektrumunun değişim yönü ve bir önceki arıza şiddetinin durumu bulanık sistemin girişlerini oluşturmuş, çıkış olarak arıza türü ve şiddeti elde edilmiştir. Önerilen algoritmanın performansı, simülasyon ve deneysel olarak elde edilen sağlam ve arızalı veriler kullanılarak doğrulanmıştır.

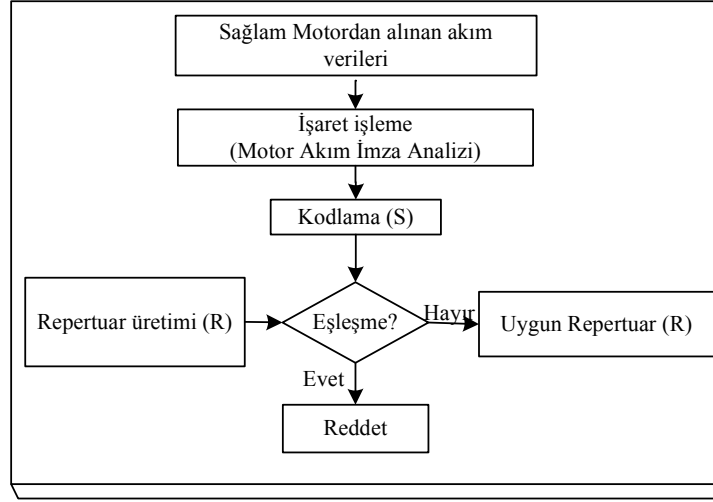
5.4.1. Önerilen Bağışık Sistem Tabanlı Bulanık Arıza Teşhis Algoritması

Arıza teşhis algoritmasının ilk adımında, stator akımı motor akım imza analizi yöntemi ile spektrum alanına dönüştürülür. Fourier dönüşümü uygulandıktan sonra güç spektrum yoğunluğu hesaplanır ve akım spektrumu oluşturulur. Bu spektrumda kırık rotor ile ilgili yan bantlar denklem (5.10)'a göre hesaplanır. Kayma değerinin hesaplanması için denklem (5.11) kullanılmıştır. Motor akım imza analizi ile akım spektrumu oluşturulduktan ve kırık rotor ile ilgili yan bantlar hesaplandıktan sonra negatif seçim algoritması için gerekli girişler oluşturulmalıdır. Negatif seçim algoritması için şekil 5.15'teki gerekli kodlama kullanılmıştır. Şekil 5.29'da, 220V ile beslenen motorun akım spektrumu gösterilmiştir. Burada f_h 50 Hz olup hat frekansını, f_h 'ın sağında olan yan bantlar yüksek ve solunda olan yan bantlar ise düşük yan bantlardır. Bu şekilde bir düşük (f_{b3}) ve bir yüksek (f_{b1}) yan bant gösterilmiştir.



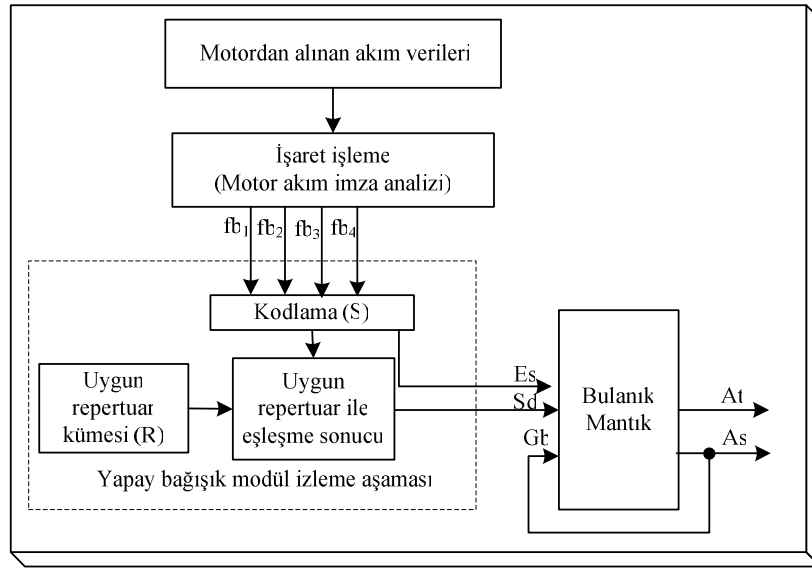
Şekil 5.29. 220 V ile beslenen bir kırık rotorlu motorun akım spektrumu

Yapay bağışık modülün eğitim aşamasında, sağlam motor verileri alınarak şekil 5.30'da görüldüğü gibi bir repertuar kümesi oluşturulur. Bu aşama çevrimdışı olarak yapılır ve sağlam motor verilerini tanıyan repertuarlar üretilir.



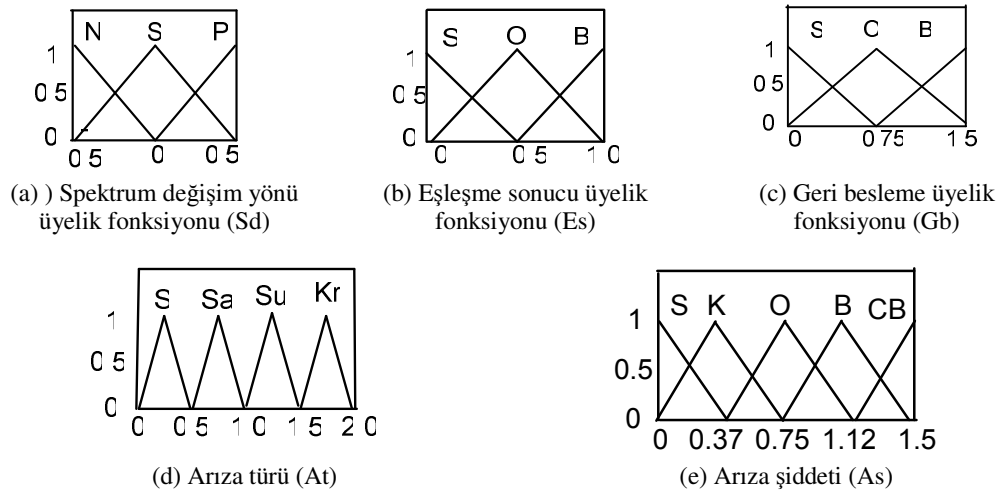
Şekil 5.30. Yapay bağışık modülünün eğitim aşaması

Eğitim aşamasında üretilen repertuar kümesi ile çevrimiçi olarak alınan veriler, arasındaki eşleşme sonuçları bulanık mantık modülünün birinci girişini oluşturur. İkinci giriş, akım spektrumunda 0–200 Hz arasındaki frekansların genliklerindeki değişim miktarının toplamıdır. Bu değişimin pozitif veya negatif olması arıza durumu etkileyecektir. Son giriş ise bir önceki adımdaki arıza şiddetidir. Bu giriş, motorda performans düşüklüğünün gerçek bir arızadan mı kaynaklandığını yoksa geçici bir durum mu olduğunu belirlemek için alınmıştır. Aşağıda şekil 5.31’de önerilen bağışık sistem tabanlı bulanık arıza teşhis şeması verilmiştir.



Şekil 5.31. Bağışık sistem tabanlı bulanık arıza teşhis şeması

Bağışık sistem tabanlı algoritma, şekil 5.31’de de görüldüğü gibi gerçek zamanda arıza teşhisi yapmaktadır. Burada Es, eşleştirme sonucunu, Sd spektrum değişim yönünü, Gb geri beslemeyi, As arıza şiddetini ve At arıza türünü gösterir. Aşağıda üç giriş, iki çıkışa sahip bulanık modülün üyelik fonksiyonları şekil 5.32’de ve bazı kural örnekleri çizelge 5.6’da verilmiştir. Bir veri ile repertuar arasındaki eşleştirmeyi hesaplamak için şekil 5.17’deki denklem kullanılmıştır.



Şekil 5.32. Bulanık modül üyelik fonksiyonları

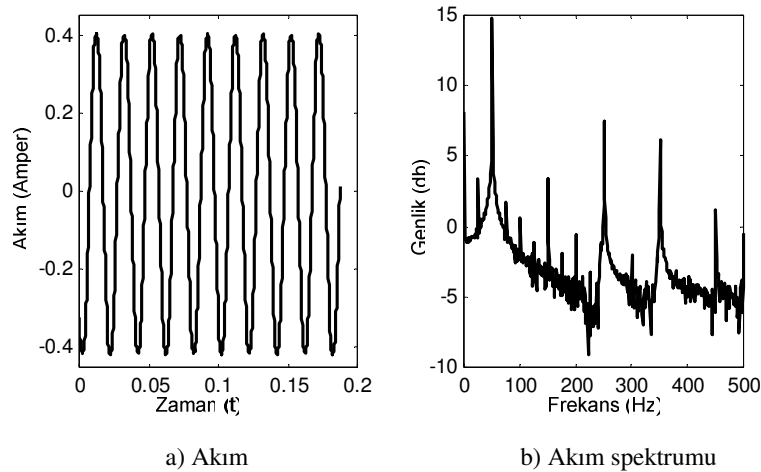
Çizelge 5.6. Bulanık modül kural örnekleri

Kural Örnekleri
1. If (Es is S) and (Sd is N) and (Gb is S) then (Ht is S) and (Hs is S)
2. If (Es is O) and (Sd is P) and (Gb is S) then (Ht is Kr) and (Hs is O)
3. If (Es is B) and (Sd is S) and (Gb is S) then (Ht is Kr) and (Hs is O)
4. If (Es is S) and (Sd is S) and (Gb is B) then (Ht is S) and (Hs is S)
5. If (Es is S) and (Sd is P) and (Gb is B) then (Ht is Su) and (Hs is B)

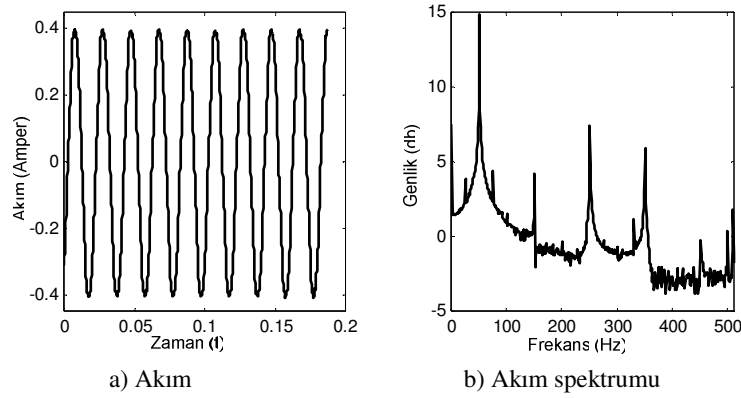
Çizelge 5.6'da Sa sarım arızasını, Su sürtünme ve Kr kırık rotor arızası durumunu gösterir. Şekil 18'den de görüldüğü gibi eşleşme denkleminde hat frekansına yakın yan bantların genlikleri daha büyük katsayılarla çarpıldığından daha etkili olmaktadır.

5.4.2 Simülasyon ve Deneysel Sonuçlar

Önerilen yapay bağışık tabanlı bulanık arıza teşhis algoritmasını uygulamak için MATLAB programı kullanılmıştır. Kırık rotor çubuğu ile ilgili motor akım verileri, üç fazlı 0.37 Kw'lık bir motordan deneysel olarak alınmıştır. Sarım ve sürtünme ile ilgili veriler ¼ hp gücündeki tek fazlı sincap kafes bir motordan simülasyon ile elde edilmiştir. Her iki motor tipi için de sağlam motor verileri referans olarak kullanılmıştır. Kırık rotor arızalarını oluşturmak amacıyla üç fazlı motorun rotor çubuklarından biri delinmiştir. Tek fazlı motorun stator sargısındaki yüz sarımdan on tanesinin kısa devre olduğu sarım arızası ve mil yatağında sürtünme arızaları oluşturulmuştur. Şekil 5.33 ve 5.34'te 220 V ile beslenen sağlam ve bir kırık rotor arızasına sahip üç fazlı motorların akım ve spektrum grafikleri verilmiştir.

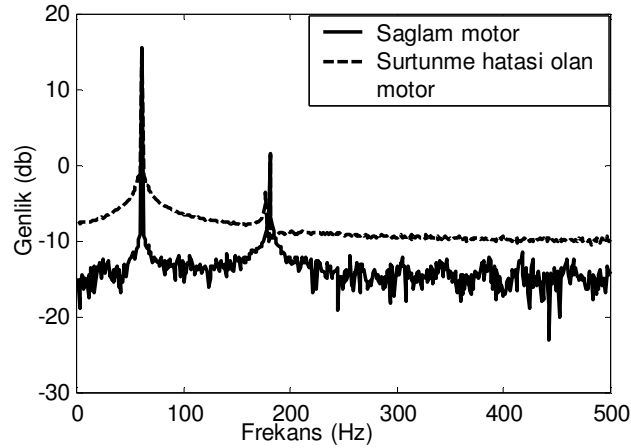


Şekil 5.33. 220 V ile beslenen sağlam motorun akım ve spektrum grafikleri



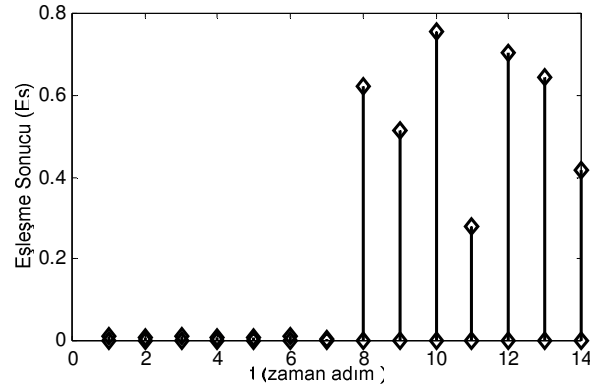
Şekil 5.34. 220 V ile beslenen bir kırık rotorlu motorun akım ve spektrum grafikleri

Sağlam bir üç fazlı motor 140, 180, 220, 260, 300, 340 ve 380 V besleme gerilimleri ile çalıştırılarak akım verileri alınmıştır. Daha sonra bu akım verileri şekil 5.30'daki algoritma ile eğitilerek sağlam motor verilerini tanıyan repertuarlar üretilmiştir. Bulanık mantığın diğer girişi için ise spektrumun değişim yönü kullanılmıştır. Şekil 5.35'te sağlam ve sürtünme arızalarının olduğu tek fazlı bir motora ait spektrumlar verilmiştir.

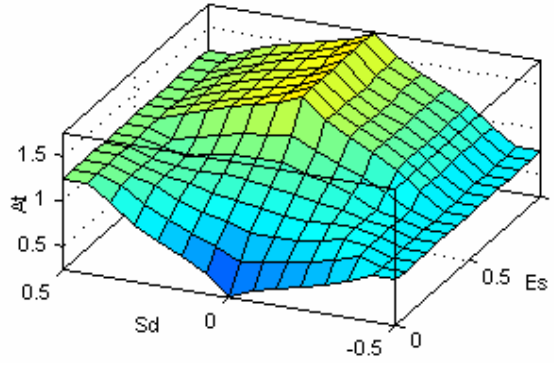


Şekil 5.35. Sağlam ve sürtünme arızası olan bir motora ait akım spektrum grafiği

Şekil 5.35'te hem kırık rotor ile ilgili yan bantlar hem de 0–200 Hz arasındaki frekansların genlik farklarının toplamı [0, 1] aralığında normaliz edilmiştir. Şekil 5.36'da, yapay bağışık sistem için bir izleme esnasında oluşan eşleşme grafiği verilmiştir. Burada eşleşme değeri arttıkça arıza şiddeti de artmaktadır. Bulanık modülün arıza şiddeti (A_s)'nin girişlere göre değişim grafiği şekil 5.37'de verilmiştir.



Şekil 5.36. Yapay bağışık sistemde izleme esnasında eşleşme sonuçları



Şekil 5.37. İki girişe göre arıza türünün değişimi

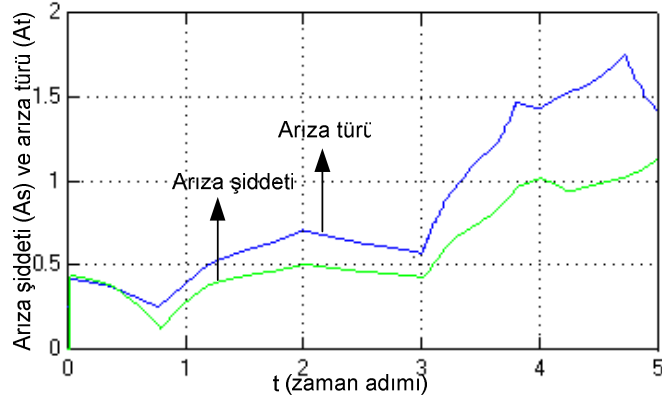
Şekil 5.37'den de görüldüğü gibi arıza türünün 0.5 ile 1.0 arasında sarım arızaları, 1.0 ile 1.5 arasında sürtünme ve 1.5 ile 2.0 arasındaki çıkışlar kırık rotor arızalarını gösterir. Aşağıda çizelge 5.7'de girişlere göre bulanık modülün çıkışları verilmiştir. Çizelge 5.8'de verilen girişlere göre algoritmanın çıkışı şekil 5.38'de verilmiştir.

Çizelge 5.7. Simülasyon sonuçları

Es	Sd	Gb	At	As
0.818	-0.392	1.050	0.794	1.14
0.049	-0.483	0.073	0.491	0.482
0.032	0.016	0.918	0.421	0.295
0.967	0.033	0.467	1.690	0.941
0.082	-0.450	0.934	0.750	0.857

Çizelge 5.8. Simülasyon girişleri

Es	Sd
0.0066	0.0366
0.0102	0.1968
0.0035	0.2952
0.6230	-0.1128
0.9961	-0.4500



Şekil 5.38. Arıza şiddeti ve arıza türü grafiği

Çizelge 5.7'nin birinci satırında; Es değeri büyük, Sd negatif ve Gb büyük ise bir sarım arızası vardır. Ayrıca arıza şiddeti (As) büyük olduğundan aynı zamanda bir kırık rotor arızası da vardır. Şekil 5.38'deki grafiğe göre; ilk adımda herhangi bir hata olmadığı için arıza türü ve şiddeti düşmüştür. İkinci ve üçüncü adımlar arasında Sd'nin değeri pozitif yönde arttığından bir sarım hatası vardır. Dördüncü ve beşinci adımlarda ise hem arıza şiddeti hem de arıza türünde bir artış olduğundan, kırık rotor ve sürtünme arızaları oluşmuştur. Algoritma gerçek zamanda arıza teşhisi yapabilmektedir. Bulanık sistemde; bir arızanın olup olmadığı, bir önceki adımda arıza şiddeti kontrol edilerek yapılmaktadır. Çünkü gerçek zamanda çalıştırılan bir motorda bazı geçici durumlardan dolayı da motor performansı düşebilir. Fakat gerçek bir arızada durumunda motorun performansının bir önceki durumuna göre daha da kötüleşmesi veya bulunduğu performansta çalışması beklenir.

Bir sistemin normal davranışları ve karakteristikleri yapılan gözlemlerle ortaya konulabilir. Kural dışı çalışmanın tespit edilmesi problemi, sistemin karakteristik davranışlarındaki sapmanın ortaya çıkarılması ile mümkün olabilir. Bağışıklık tabanlı arıza teşhis yaklaşımı da, bağışıklık sisteminin herhangi bir yabancı hücreyi vücudun kendi hücresinden ayırt etme özelliğini kullanır. Böyle bir sistem arıza teşhisi için kullanıldığında, arızanın tipi belirlenmeyip sadece kural dışı çalışmalar belirlenebilir. Fakat motorlarda birçok

arıza tipi oluşmaktadır. Bu yüzden sistemin bu kural dışı çalışmasının sebebinin belirlenmesi de önemli bir problemdir. Bu problemi çözmek için çalışmada bulanık mantık kullanılmıştır. Motorlarda en çok görülen rotor, mil yatağı ve stator arızalarının teşhisi için bir yaklaşım sunulmuştur. Önerilen algoritma aynı anda gelen iki arıza türünü teşhis edebildiği gibi arıza şiddeti çıkışı ile motorun durumu hakkında da bilgi vermektedir. Ayrıca bulanık sistemin bir önceki durumu kontrol edilerek, geçici performans düşüklüklerinin teşhis algoritmasını yanlış yönde etkilemesi önlenmiştir.

6. SONUÇLAR

Asenkron motorlar endüstride devamlılığın sağlanması ve işlerin aksamaması açısından büyük öneme sahip araçlardır. Bu tür motorların basit yapıları, kullanışlı olmaları, maliyetlerinin ucuz olması ve daha az bakıma ihtiyaç duymalarından dolayı diğer motor tiplerine göre daha çok tercih edilmektedir. Çalışma ortamları bakımından bu tür motorların sürekli sağlıklı ortamlarda çalıştıklarını söylemek mümkün değildir. Bulundukları ortamda nem ve aşındırıcı sebepler bu motorların ömrünü azaltmaktadır. Bu motorların maliyeti ucuz olmasına rağmen, kullanıldıkları uygulamalar üretimin ve çalışmanın devamı açısından önemlidir. Bu tür motorların herhangi bir kısmında başlangıç aşamasında oluşmaya başlayan arızalar ciddi performans düşüklüklerine sebep olmazlar. Fakat başlangıç aşamasındaki bu arızalar teşhis edilmezse daha büyük arızalara sebep olurlar. Arıza teşhisinde amaç erken safhadaki arızaları teşhis ederek motorların bakıma alınmasını ve gerekiyorsa yeni bir motor ile değiştirilmesini sağlamaktır. Böylece fabrikadaki üretimin devamlılığı sağlanır ve performans düşüklükleri engellenmiş olunur.

Yumuşak hesaplama teknikleri, matematiksel modeli çıkarılamayan veya sistemin giriş ile çıkışı arasında lineer olmayan bir ilişki olduğunda uygun çözüm bulmak amacıyla kullanılırlar. Lineer olmayan bir problemi çözmek için yapay sinir ağları kullanıldığında, sistem davranışlarını gösteren karmaşık matematiksel modellerin bilinmesine gereksinim olmaz. Tek gereken yapay sinir ağının eğitimini sağlayacak, sistemin giriş ve çıkışları arasındaki ilişkiyi gösteren eğitim örnekleri ile yapay sinir ağının eğitmektir. Yapay sinir ağı eğitildikten sonra gelen test örnekleri için doğru sonuçları verecektir. Bulanık mantık keskin doğruluğun çok önemli olmadığı, belirl bir hata toleransına izin verilebilecek lineer olmayan sistemlerin çözümü için kullanılır. Bulanık mantık bir problemi çözmek amacıyla uzman bilgisinden faydalanır. Uzamandan alınan bilgiler ışığında problemi çözecek giriş ve çıkış üyelik fonksiyonları belirlenerek kural tabanı oluşturulur. Yeni bir yumuşak hesaplama yaklaşımı olan yapay bağışık sistemler insanın bağışıklık sistemindeki koruma mekanizmasının benzetimi olup, mühendislik ve bilgisayar ile ilgili problemlerin çözümünde kullanılır. Mühendislik alanında özellikle optimizasyon, güvenlik, anomali tespiti, örüntü tanıma ve sınıflandırma gibi alanlarda geniş bir şekilde kullanılmıştır.

Büyük veri kümelerinden kullanışlı ve faydalı bilgi çıkarılması amacıyla kullanılan veri madenciliğinin zaman üzerinde alınan bilgilere uygulanması ile oluşan veri madenciliği yöntemi zaman serileri veri madenciliğidir. Veri madenciliğindeki kullanışlı veya faydalı bilgi, zaman serileri veri madenciliğinde olay veya zaman serisindeki değişimlerin bulunması olarak değişim

gösterir. Zaman serileri arasındaki benzerliklerin bulunması, zaman serilerinde oluşan ilginç olayların çıkarılması veya zaman serilerinin sınıflandırılması gibi zaman serileri veri madenciliği yöntemleri araştırmacılar tarafından farklı alanlara uygulanmıştır.

Yumuşak hesaplama teknikleri asenkron motorların arıza teşhisi için kullanıldığında motorun matematiksel dinamiklerine ihtiyaç duyulmaz. Arıza teşhisi için motorun akım, gerilim, hız ve moment gibi bilgileri kullanılır. Motorda herhangi bir arıza oluşmaya başladığında bu sinyallerde değişimler olur. Zaman üzerinde alınan bu bilgiler genelde bir ön işlemeden sonra yumuşak hesaplama yöntemlerine verilerek arıza teşhisi yapılır. Bazı uygulamalarda bu bilgilerin direkt zaman üzerinde alınan değerleri kullanılmakla birlikte, bu tür yöntemler ile arızaların türünü belirleyecek teşhisler yapılamayıp, motordaki kural dışı bir çalışma durumunu belirlenebilir.

Zaman serileri veri madenciliği asenkron motorlarda akım ve moment bilgisini kullanarak arıza teşhisi yapmak amacıyla kullanılabilir. Motordan alınan akım bilgileri bir zaman serisi olarak düşünülebildiğinden bu yöntemin kullanılarak farklı arıza durumlarının sınıflandırılması mümkün olmaktadır. Bir motordan alınan akım veya moment zaman serisi lineer olmayan zaman serileri analizi ile faz alanına dönüştürülerek veri madenciliği algoritmaları ile arızalar sınıflandırılabilir.

Asenkron motor arızalarını teşhis etmek amacıyla iki farklı motor tipi kullanılmıştır. Sarım ve mil yatağı sürtünme arızaları ile ilgili akım ve hız bilgileri tek fazlı bir asenkron motordan simülasyon ile elde edilmiştir. Kırık rotor arızaları ile ilgili faz akımı verileri deneysel olarak üç fazlı bir asenkron motordan farklı hızlarda çalıştırılarak bilgisayara aktarılmıştır.

Yapılan birinci uygulama çalışmasında tek fazlı bir asenkron motordaki sarım ve sürtünme arızalarını teşhisi için yapay sinir ağının lineer olmayan sistemleri öğrenme özelliğinden faydalanılmıştır. Stator akımı ile rotor hızı, farklı arıza durumları için lineer olmayan bir şekilde değiştiğinden yapay sinir ağının girişi olarak bu iki bilgi kullanılmış ve çıkış olarak da arızaların durumları verilmiştir. Çevrimdışı olarak yapay sinir ağı eğitildikten sonra, gerçek zamanda arızalar teşhis edilmiştir. Yöntemin ikinci modülü olan bulanık mantık modülünde ise kırık rotor ve hava boşluğu eksantrikliği arızaları teşhis edilmiştir. Stator akımı bir ön işlemeye tabii tutularak akım spektrumu oluşturulmuş ve bu spektrumda yan bantlar incelenerek kırık rotor ve eksantriklik arızaları teşhis edilmiştir. Sunulan yöntemin getirmiş olduğu en önemli avantaj farklı arıza tiplerinin aynı anda teşhis edilebilmesidir.

İkinci uygulama çalışmasında sarım ve sürtünme arızalarını sınıflandırılması için zaman serileri veri madenciliği teknikleri kullanılmıştır. Zaman serileri veri madenciliğini kullanarak arıza teşhisi yapan daha önceki çalışmalar moment gibi motordan elde edilmesi pahalı ve

hassas ölçümler gerektiren bilgileri kullanmıştır. Önerilen çalışmada ise sadece akım ve hız bilgileri kullanılmış ve her bir arızalı durumu kolayca ayırt edebilecek bir dönme yarıçapı bulunmuştur.

Yapay bağışık sistemler kullanan arıza teşhis yaklaşımları genelde direkt akım üzerinden veya titreşim gibi hassas ve maliyetli ölçümler gerektiren veriler kullanılmıştır. Temel olarak yapılan sağlam motor durumlarını tanıyacak repertuarların üretilmesi ve herhangi bir kural dışı çalışma olduğunda onu tespit edilmesini sağlamaktır. Üçüncü uygulama çalışmasında önerilen yaklaşım ile kırık rotor arızalarını teşhis etmek amacıyla akım bilgisini direkt olarak kullanmak yerine akım bir ön işlemeden geçirildikten sonra kırık rotor ile ilgili yapay bağışıklık modeli oluşturulmuştur. Önerilen yaklaşımın bir avantajı da arızayı belirlemede daha etkili olan yan bantların kodlamada büyük ağırlığa sahip olmasıdır. Yöntem sadece tek faz akımını giriş olarak alır ve bir, iki ve üç kırık rotor arızalarını teşhis eder.

Bütün arıza türleri için kullanılabilecek bir çalışma ise üçüncü uygulamada verilmiştir. Bu çalışmada farklı iki motor tipi kullanılarak farklı arıza türleri teşhis edilmiştir. Sınıflandırma için bulanık mantık kullanılmıştır. Önerilen yöntem ile üç fazlı bir asenkron motordaki kırık rotor arızaları ve tek fazlı bir asenkron motordaki sarım ve sürtünme arızaları teşhis edilmiştir. Sunulan yöntem ile sadece arızalar teşhis edilmeyip aynı zamanda arızanın şiddeti de teşhis edilmiştir.

Sonuç olarak yumuşak hesaplama ve veri madenciliği tabanlı teknikler arıza teşhisi için kullanıldığında, motorun iç yapısı ve modelinin bilinmesine gerek kalmadan referans olarak sağlam motor bilgileri de bilinmek şartıyla farklı motor tiplerine uygulanabilir. Önerilen algoritmaların performansı deneysel ve simülasyon sonuçlarına göre başarılı görülebilir.

KAYNAKLAR

- [1] Benbouzid, M. E. H. and Kliman, G. B., 2003, What Stator Current Processing-Based Technique to Use for Induction Motor Rotor Faults Diagnosis?, IEEE Transaction on Energy Conversion, Vol. 18, No. 2, 238-244.
- [2] Chow, M. Y., 1997, Methodologies of Using Neural Network and Fuzzy Logic Technologies for Motor Incipient Fault Detection, World Scientific, 138 s.
- [3] Chow, M. Y., Mangum, P. M. and Yee, S. O., 1991, A Neural Network Approach to Real-Time Condition Monitoring of Induction Motors, IEEE Transaction on Industrial Electronics, Vol. 38, No. 6, 448-453.
- [4] Kowalski, C.T. and Kowalska, T. O., 2003, Neural Networks Application for Induction Motor Faults Diagnosis, Mathematics and Computers in Simulation, Vol. 63, 435-448.
- [5] Chow, M. Y. and Sharpe R. N., 1993, On the Application and Design Consideration of Artificial Neural Network Fault Detector, IEEE Transaction on Industrial Electronics, Vol. 9, No. 2, 181-198.
- [6] Benbouzid, M. E. H, 2000, A Review of Induction Motors Signature Analysis as a Medium for Faults Detection, IEEE Transaction on Industrial Electronics, Vol. 47, No. 5, 984-993.
- [7] Filippetti, F., Franceschini, G., Tesson, C., and Vas, P., 2000, Recent Developments of Induction Motors Drives Fault Diagnosis Using AI Techniques, IEEE Transactions on Industrial Electronics, Vol. 47., No. 5, 994-1004.
- [8] Zadeh, L., 1994, Soft Computing and Fuzzy Logic, Software IEEE, Vol. 11, Issue 6, 48-56.
- [9] De Castro, L. N. and Timmis, J., 2002, Artificial Immune Systems: A New Computational Intelligence Approach, Springer, 357 s.
- [10] Mohammed, A., Awadallah, M., and Morcos, M., 2003, Application of AI Tools in Fault Diagnosis of Electrical Machines and Drives- An Overview, IEEE Transaction on Energy Conversion, Vol. 18., No.2, 245-251.
- [11] Ayhan, B., Chow, M.Y., Trussell, H. J. and Song M. H. , 2003, A Case Study on Comparison of Non-parametric Spectrum Methods for Broken Rotor Bars Fault Detection, Industrial Electronics Society (IECON'03-The 9th Annual Conference of the IEEE), Vol. 3, 2835-2840.
- [12] Haji, M. and Toliyat, A. H., 2001, Pattern Recognition- A Technique for Induction Machines Rotor Fault Detection "Eccentricity and Broken Bar Fault", IEEE Transactions on Energy Conversion, IEEE Transaction on Energy Conversion, Vol. 16, Issue 4, 312-317.
- [13] Benbouzid, M. E.H. and Nejari, H., 2001, A Simple Fuzzy Logic Approach for Induction Motors Stator Condition Monitoring, IEEE Electric Machines and Drives Conference, 634-639.

- [14] Ayhan, B., 2000, Fault Detection of Induction Motors Using Neural Networks, Ms. Dissertation, Boğaziçi Üniversitesi, Fen Bilimleri Enstitüsü, 94 s.
- [15] Gülez, K., 1995, Motor Hata Tespit ve Analizi için Yapay Sinir Ağlarının Tasarımı, Yüksek Lisans Tezi, Yıldız Teknik Üniversitesi, Fen Bilimleri Enstitüsü, 78 s.
- [16] Kolla, S. and Varatharasa, L., 2000, Identifying Three-Phase Induction Motor Faults Using Artificial Neural Networks, Elsevier ISA Transactions 39, 433-439.
- [17] Zidani, F., Benbouzid, M. E. H., Diallo, D. and Said, M. S., 2003 , Induction Motor Stator Faults Diagnosis by a Current Concordia Pattern-Based Fuzzy Decision System, IEEE Transaction on Energy Conversion, Vol. 18, No.4, 469-475.
- [18] Altug, S., Chow, M. Y. and Trussel, H.J., 1999, Fuzzy Inference Systems Implemented on Neural Architectures for Motor Fault Detection and Diagnosis, IEEE Transaction on Industrial Electroncis, Vol. 46, No. 6, 1069-1079.
- [19] Alford, T. and Xu, M., 1998, Motor Current Analysis and Its Applications in Induction Motor Fault Diagnosis, ENTERACT'98 Conference.
- [20] Sin, M. L., Soong, W. L. and Ertugrul N., Induction Machine On-line Condition Monitoring and Fault Diagnosis-A Survey, University of Adelaide.
- [21] Benbouzid, M. E. H., Beguenane, R., Nejari, H. and Vieira, M., 1997, Three Phase Induction Machines Asymetrical Faults Identification by Means of Stator Current Signature Analysis, Proceeding of IEEE International Syposium on Diagnostics for Electrical Machines,Power Electronics and Drives, Vol.1 ,9-14.
- [22] Zhongming, Y. and Bin, W., 2000, A Review of Induction Motor Online Fault Diagnosis, IEEE Power Electronics and Motion Control Conference, Vol.3, 1353-1358.
- [23] Acosta, G. G., Verucchi, C.J. and Gelso, E.R., 2006, A Current Monitoring System for Diagnosing Electrical Failures in Induction Motors, Mechanical Systems and Signal Processing, Volume 20, Issue 4, 953-965.
- [24] Ayhan, B., Chow, M.Y. and Song, M. H., 2005, Multiple Signature Processing-Based Fault Detection Schemes for Broken Rotor Bar in Induction Motors”, IEEE Transaction on Energy Conversion, Vol. 20 , No. 2, 336-343.
- [25] Branco, P. J. C., Dente, J. A. and Mendes, R. V., 2003, “Using Immunology Principles for Fault Detection”, IEEE Transaction on Industrial Electronics, Vol. 50, No.2, 362-373.
- [26] Gao X. Z., Ovaska S. J., Wang X. and Chow M.Y., 2004, Neural Network Based Negative Selection Algorithm with Applications in Fault Diagnosis”, IEEE International System Man and Cybernetics Conference, 3408-3414.

- [27] Martins, J. F., Branco, P. J., Pires, A. J. and Dente J. A., 2000, Fault Detection Using Immune-Based Systems and Formal Language Algorithms, 39th IEEE Conference on Decision and Control, 2633-2638.
- [28] Shulin, L., Jiazhong, Z., Wengang S. and Wenhui, H., 2002, Negative-Selection Based Approach for Fault Diagnosis of Rotary Machinery, IEEE Proceedings of the American Control Conference, 2955-3960.
- [29] Povinelli, R. J., 1999, Time Series Data Mining: Identifying Temporal Patterns for Characterization and Prediction of Time Series Events, Ph.D. Dissertation, Marquette University, Milwaukee, WI.
- [30] Ye, N., 2003, The Handbook of Data Mining, Lawrence Erlbaum Associates Publishers, London, Arizona State University, 703 s.
- [31] Han, J. and Kamber, M., Data Mining Concept and Techniques, 2001, Morgan Kaufmann Publishers, Simon Fraser University, 550 s.
- [32] Bangura, J. F., Povinelli, R.J., Demerdash, N. A. O. and Brown, R. H., 2003, Diagnostics of Eccentricities and Bar/End-Ring Connector Breakages in Polyphase Induction Motors Through a Combination of Time-Series Data Mining and Time-Stepping Coupled FE-State-Space Techniques, IEEE Transaction on Industry Applications, Vol. 39, No.4,1005-1013.
- [33] Yeh, C., Povinelli, R. J., Mirafzal, B. and Demerdash, N.A.O., 2004, Diagnosis of Stator Winding Inter-Turn Shorts in Induction Motors Fed By PWM Inverter Drive Systems Using a Time-Series Data Mining Technique, International Conference on Power System Technology, Powercon2004, Singapore.
- [34] Takens F., 1980, Detecting Strange Attractors in Turbulence, Proceedings of Dynamical Systems and Turbulence, Warwick, 366-381.
- [35] Siddique, A., Yadava, G.S. and Singh, B., 2005, A Review of Stator Fault Monitoring Techniques of Induction Motors, IEEE Transaction on Energy Conversion, Vol. 1, 200-206.
- [36] Kim, K. and Parlos, A. G., 2002, Induction motor Fault Diagnosis Based on Neuropredictors and Wavelet Signal Processing, IEEE/ASME Transaction on Mechatronics, Vol. 7, No. 2, 201-219.
- [37] Nandi, S. and Toliyat, A. H., 1991, Condition Monitoring and Fault Diagnosing of Electrical Machines- A review, IEEE Industry Application Conference (34th IAS Annual Meeting), Vol. 1, 197-204.
- [38] Coley, P., 2004, Fault Zone Analysis "ROTOR", Motor Reliability Technical Conference, 1-4.
- [39] Kecman, V., 2001, Learning and Soft Computing, The Massachusetts Institute of Technology Press Cambridge, London-England, 541 s.

- [40] Kulkarni, D. A., 2001, Computer Vision and Fuzzy-Neural Systems, Prentice Hall Ptr, New Jersey, 504 s.
- [41] Öztemel, E., 2003, Yapay Sinir Ağları, Papatya Yayıncılık, 232 sf.,
- [42] Nabiyev, V. V., 2003, Yapay Zeka-Problemler, Yöntemler, Algoritmalar, Seçkin Yayıncılık, 724 sf.,
- [43] Karnin, E.D., 1990, A Simple Procedure for Pruning Back-propagation Trained Neural Networks, IEEE Transactions on Neural Networks, Vol. 1, Issue. 2, 239-242.
- [44] Mendel, J. M., 1995, Fuzzy Logic Systems for Engineering: A tutorial, Proceedings of IEEE, Vol. 83, No. 3, 345-377.
- [45] de Castro, L. N., and Timmis, J. I., 2003, Artificial Immune Systems as A Novel Soft Computing Paradigm, Soft Computing Journal, Vol. 7, Issue 7.
- [46] Karaboğa, D., 2004, Yapay Zeka Optimizasyon Algoritmaları, Atlas Yayın Dağıtım, İstanbul, Yayın No. 38, 199 s.
- [47] de Castro, L. N. and Von Zuben, F. J., 1999, Artificial Immune Systems: Part I – Basic Theory and Applications, Technical Report – RT DCA 01/99, 95 S.
- [48] Berek, C. and Ziegner, M., 1993, The Maturation of the Immune Response, Imm. Today, Vol. 14, Issue 8, 400–402.
- [49] Perelson, A. S. and Oster, G. F., 1979, Theoretical Studies of Clonal Selection: Minimal Antibody Repertoire Size and Reliability of Self-Nonself Discrimination, Journal of Theoretic Biology, Vol. 81, 645-670.
- [50] Perelson, A. S., 1989, Immune Network Theory, Immn. Rev., Vol. 110, 5-36.
- [51] Forrest, S., Perelson Allen, L. and Cherukuri, R., 1994, Self-Nonself Discrimination in a Computer, Proceeding of IEEE Symposium in Security and Privacy, 202-212.
- [52] De Castro, L.N. and Von Zuben, F.J., 2000, The clonal Selection Algorithm with Engineering Applications, GECCO 2000, Las Vegas, Nevada, USA.
- [53] Fayyad, P. S. U. M, Piatetsky-Shapiro, G. and Uthurusamy, R., 1996, Advances in knowledge discovery and data mining, Cambridge, MA: MIT Press.
- [54] Roiger, R. and Geatz, M., 2002, Data Mining A Tutorial Based Primer, Addison Wesley Publisher, 408 s.
- [55] Agrawal, R., Imielinski, T., Swami, A., 1993, “Mining Association Rules between Sets of Items in Large Databases”, In ACM SIGMOD Conference Management of Data.
- [56] Karagöz, M., 1998, İstatistik Yöntemleri, Özmert Ofset Matbaacılık ve Form Baskı Tesisleri-Malatya, 461 s.
- [57] Moon, Y. S., Whang, K. Y. and Loh, W. K., 2001, Efficient time series subsequence matching using duality in construction Windows, Elsevier Information Systems, 279-293.

- [58] Goldin, D. Q., Kanellakis, P. C., 1995, On Similarity Queries for Time Series Data: Constraint Specification and Implementation, Springer Principles and Practice of Constraint Programming Conference, 137-153.
- [59] Jadagish, H. V., Mendelzon, A.O. and Milo, T., 1995, Similarity Based Queries. Proceedings of ACM Symposium on Principles of Database System, 36-45.
- [60] Rafiei, D. and Mendelzon, A., 1998, Efficient retrieval of similar time sequences using DFT, FODO'98 Conference.
- [61] Mark, L., Abraham, K. and Horst, B., 2004, Data Mining in Time Series Databases, World Scientific, Machine Perception Artificial Intelligence, Vol 57, 192 S.,
- [62] H. Kantz and Schreiber, T., 1997, Nonlinear Time Series Analysis, Cambridge University Press, 304 s.
- [63] Povinelli, R. and Feng, X., 2003, A New Temporal Pattern Identification Method for Characterization and Prediction of Complex Time Series, IEEE Transaction on Knowledge and Data Engineering, Vol. 15, No.2, 339–352.
- [64] Feng, X., Huang, H., 2005, A Fuzzy Set Based Reconstructed Phase Space Method for Identification of Temporal Patterns in Complex Time Series, IEEE Transactions on Knowledge and Data Engineering, Vol. 17, No. 5, S. 601-613.

ÖZGEÇMİŞ

İlhan AYDIN

Fırat Üniversitesi
Mühendislik Fakültesi
Bilgisayar Mühendisliği Bölümü
23114, ELAZIĞ

E_Posta : iaydin@firat.edu.tr

1981	Palu’da doğdu.
1994-1997	Atatürk Lisesinden mezun oldu.
1997-2001	Fırat Üniversitesi Bilgisayar Mühendisliği bölümünden mezun oldu.
2001-	Kemaliye Hacı Ali Akın Meslek Yüksek Okulunda Öğretim Görevlisi olarak görev yapmaktadır.

FÜBAP-1140 Nolu proje tarafından desteklenmiştir.

1. İlhan AYDIN, Mehmet KARAKÖSE, Erhan AKIN, Asenkron Motorlarda Hata Teşhisi İçin Yeni Bir Yumuşak Hesaplama Yaklaşımı, TOK'05 Otomatik Kontrol Konferansı, İstanbul, 2005.
2. İlhan AYDIN, Mehmet KARAKÖSE, Erhan AKIN, Zaman Serileri Veri Madenciliği Tekniği Kullanılarak Asenkron Motorlarda Sarım ve Sürtünme Hatalarının Teşhisi, EMO2005 11. Elektrik ve Bilgisayar Mühendisliği Kongresi, İstanbul, 2005.
3. İlhan AYDIN, Mehmet KARAKÖSE, Erhan AKIN, Bulanık Mantık Tabanlı Zaman Serisi Veri Madenciliği Kullanan Tahmin Algoritması, MBGAK'05, İstanbul, 2005.
4. Bilal ALATAS, İlhan AYDIN, Erhan AKIN, Asenkron Motorların Hata Teşhisinde Yapay Bağışıklık Sistemi Yaklaşımı, II. Mühendislik Bilimleri Genç Araştırmacılar Kongresi (MBGAK 2005), İstanbul
5. İlhan AYDIN, Mehmet KARAKÖSE, Erhan AKIN, Yapay Bağışıklık Sistemi Tabanlı Arıza Teşhis Algoritması, IEEE 14. Sinyal İşleme ve İletişim Uygulamaları Kurultayı(SİU2006), Antalya, 2006
6. İlhan AYDIN, Mehmet KARAKÖSE, Erhan AKIN, Bağışık Sistem Tabanlı Bulanık Arıza Teşhis Algoritması, ASYU-INISTA 2006, İstanbul, Haziran 2006.

ADNANTECH PCL-816 Veri Toplama Kartının Özellikleri

Genel Özellikler

- 16 bit çözünürlüklü Analog/Dijital dönüştürücü
- Yüksek performanslı 100 Khz örnekleme oranı
- Ayrı ayrı programlanabilir kazançlar ile 16 farklı analog giriş
- Programlanabilir DMA kanalı
- Programlanabilir IRQ kesme seviyesi
- Gürültü azaltma için metalik korumalı Analog/Dijital modül
- Otomatik kanal tarama devresi
- Seçimli olarak 16-bit Dijital/ Analog çıkış modülü

Analog Giriş

- Kanallar: 16 diferansiyel
- Çözünürlük: 16 bit
- Dönüştürme süresi: 8.5 Mikro saniye
- Maksimum örnekleme oranı: 100 khz
- Yazılım için programlanabilir giriş aralığı (Volt)
Çift kutuplu: ± 10 , ± 5 , ± 2.5 , ± 1.25
Tek kutuplu: 0~10, 0~5, 0~2.5, 0~1.25
- Tetikleme modu: software, pacer ve harici tetikleme
- Veri transferi: yazılım, kesme veya DMA
- Doğruluk: %0.003
- Giriş Empedansı: $>10 \text{ M}\Omega$
- Giriş Gerilimi: $\pm 15 \text{ V}$

Dijital Giriş

- Kanal sayısı: 16
- Mantık düzeyi: TTL-uyumlu
Mantık düzeyi 0: 0.8 V maksimum
Mantık düzeyi 1: 2 V minimum
- Konektör: 20 pin yassı kablo

Dijital çıkış

- Kanal sayısı: 16
- Mantık düzeyi: TTL-uyumlu
Mantık düzeyi 0: 0.4 V maksimum
Mantık düzeyi 1: 2.4 V minimum
- Konektör: 20 pin yassı kablo

Yazılım ve Sürücüler

- Windows DLL Sürücüler: PCL- 816'nın Windows 95/98/NT dinamik link kütüphanesi Visual Basic, Visual C++, Delphi ve C++ Builder gibi araçları kullanan Microsoft Windows programları yazmanıza izin verir.
- ActiveX Control: Advantech ActiveDAQ görsel programlama dilleri için ActiveX kontrolü sağlar.
- Uygulama Paketleri: PCL-816 Labview ve VisiDAQ gibi hazır veri toplama paket programları ile desteklenir.

EK-2

Veri Toplama Kartından Akım Verilerini Okumak için Yazılan Delphi Programı

```
unit AIUnit;
interface
uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  OleCtrls, StdCtrls, DAQAILib_TLB, Buttons, ExtCtrls, Variants, TeeProcs,
  TeEngine, Chart, Series, Grids;
type
  TForm1 = class(TForm)
    GroupBox1: TGroupBox;
    Label1: TLabel;
    Label2: TLabel;
    GroupBox8: TGroupBox;
    cmdAcquireStart: TButton;
    cmdAcquireStop: TButton;
    cmdStatus: TButton;
    txtStatus: TEdit;
    txtDeviceNum: TEdit;
    txtDeviceName: TEdit;
    cmdSelectDevice: TButton;
    GroupBox10: TGroupBox;
    Label5: TLabel;
    cmbStartChannel: TComboBox;
    Label6: TLabel;
    cmbNumChan: TComboBox;
    Label3: TLabel;
    cmbInputRange: TComboBox;
    chkOverAllGain: TCheckBox;
    cmdGainList: TButton;
    Label4: TLabel;
    cmbDataType: TComboBox;
    Label12: TLabel;
    cmbTransferMode: TComboBox;
    Label13: TLabel;
    cmbTrigSource: TComboBox;
    Label14: TLabel;
    cmbClockSource: TComboBox;
    Label15: TLabel;
    txtSampleRate: TEdit;
    Label8: TLabel;
    txtNumOfSample: TEdit;
    GroupBox6: TGroupBox;
    Label9: TLabel;
    chkFIFOEnabled: TCheckBox;
    txtFIFOSize: TEdit;
    chkEventEnabled: TCheckBox;
    chkCyclicMode: TCheckBox;
    GroupBox2: TGroupBox;
    Label11: TLabel;
    txtErrorCode: TEdit;
    txtErrorMessage: TEdit;
    Label10: TLabel;
    DAQAI1: TDAQAI;
    ErrorTimer: TTimer;
    Label7: TLabel;
    Label28: TLabel;
    sure: TEdit;
    Label16: TLabel;
    Label29: TLabel;
```

```

Button5: TButton;
lstfaza: TListBox;
lstfazb: TListBox;
lstfazc: TListBox;
Label23: TLabel;
Label22: TLabel;
Label24: TLabel;
Memo1: TMemo;
Label31: TLabel;
Label25: TLabel;
Label26: TLabel;
Label27: TLabel;
GroupBox3: TGroupBox;
Edit3: TEdit;
Button2: TButton;
Label21: TLabel;
knlsys: TEdit;
Label18: TLabel;
Label19: TLabel;
Label20: TLabel;
Edit1: TEdit;
Edit2: TEdit;
Edit4: TEdit;
cmdExit: TBitBtn;
procedure chkFIFOEnabledClick(Sender: TObject);
procedure chkOverAllGainClick(Sender: TObject);
procedure cmbDataTypeClick(Sender: TObject);
procedure cmbTransferModeClick(Sender: TObject);
procedure cmdAcquireStartClick(Sender: TObject);
procedure cmdAcquireStopClick(Sender: TObject);
procedure cmdGainListClick(Sender: TObject);
procedure cmdSelectDeviceClick(Sender: TObject);
procedure cmdStatusClick(Sender: TObject);
procedure DAQAllEventRaw(Sender: TObject; DataCount: Integer;
    Data: OleVariant);
procedure DAQAllEventReal(Sender: TObject; DataCount: Integer;
    Data: OleVariant);
procedure DAQAllTerminated(Sender: TObject);
procedure FormCreate(Sender: TObject);
procedure txtDeviceNumChange(Sender: TObject);
procedure txtNumOfSampleChange(Sender: TObject);
procedure txtSampleRateChange(Sender: TObject);
procedure chkCyclicModeClick(Sender: TObject);
procedure chkEventEnabledClick(Sender: TObject);
procedure cmbClockSourceChange(Sender: TObject);
procedure cmbTrigSourceChange(Sender: TObject);
procedure cmbStartChannelChange(Sender: TObject);
procedure cmbNumChanChange(Sender: TObject);
procedure ErrorTimerTimer(Sender: TObject);
procedure FormClose(Sender: TObject; var Action: TCloseAction);
procedure Button2Click(Sender: TObject);
procedure Button5Click(Sender: TObject);
private
    { Private declarations }
public
    { Public declarations }
end;
var
    Form1: TForm1;
bRet : Boolean;
f:textfile;
dizi:array of string;
fa,fb,fc:Textfile;
implementation
uses GainUnit;
{$R * DFM}
procedure TForm1.chkFIFOEnabledClick(Sender: TObject);

```

```

begin
  if chkFIFOEnabled.State = cbChecked then
    begin
      txtFIFOSize.Enabled := True;
      DAQAI1.FIFOEnabled := True;
      txtFIFOSize.Text := IntToStr(DAQAI1.FIFOSize);
    end
  else
    begin
      txtFIFOSize.Enabled := False;
      DAQAI1.FIFOEnabled := False;
      txtFIFOSize.Text := '0';
    end;
  end;
end;
procedure TForm1.chkOverAllGainClick(Sender: TObject);
begin
  if chkOverAllGain.State = cbChecked then
    begin
      cmbInputRange.Enabled := True;
      cmdGainList.Enabled := False;
      DAQAI1.InputRangeMode := adOverallRange;
    end
  else
    begin
      cmbInputRange.Enabled := False;
      cmdGainList.Enabled := True;
      DAQAI1.InputRangeMode := adDifferentRange;
      {Set gain code list}
    end;
  end;
end;
procedure TForm1.cmbDataTypeClick(Sender: TObject);
begin
  DAQAI1.DataType := cmbDataType.ItemIndex;
end;
procedure TForm1.cmbTransferModeClick(Sender: TObject);
begin
  DAQAI1.TransferMode := cmbTransferMode.ItemIndex;
end;
procedure TForm1.cmdAcquireStartClick(Sender: TObject);
var
  i : integer;
begin
  lstfaza.Clear;
  lstfazb.Clear;
  lstfazc.Clear;
  if DAQAI1.OpenDevice then
    begin
      ShowMessage(DAQAI1.ErrorMessage);
      Exit;
    end;
  if DAQAI1.InputRangeMode = adDifferentRange then
    begin
      gGainCodeList := VarArrayCreate([0, gNumOfChan - 1], varSmallint);
      for i := 0 to gNumOfChan - 1 do
        gGainCodeList[i] := gGainCodeSave[i];
      DAQAI1.InputRangeList := gGainCodeList;
    end
  else
    begin
      DAQAI1.OverallInputRange := cmbInputRange.ItemIndex;
      if DAQAI1.OverallInputRange < 0 then
        DAQAI1.OverallInputRange := 0;
      end;
      {Veri alımına başlanması}
      bRet := DAQAI1.AcquireStart;
      edit1.Text:=timetostr(time);
      if bRet then

```

```

begin
    ShowMessage(DAQAI1.ErrorMessage);
    Exit;
end;
GroupBox10.Enabled := False;
cmdAcquireStart.Enabled := False;
cmdAcquireStop.Enabled := True;
cmdExit.Enabled := False;
cmdSelectDevice.Enabled := False;
ErrorTimer.Enabled := True;
end;
procedure TForm1.cmdAcquireStopClick(Sender: TObject);
var
    i:integer;
    ororn:single;
    vsay:longint;
    ks:integer;
    a: array of single;
begin
    GroupBox10.Enabled := True;
    cmdAcquireStart.Enabled := True;
    cmdAcquireStop.Enabled := False;
    cmdExit.Enabled := True;
    cmdSelectDevice.Enabled := True;
    ErrorTimer.Enabled := False;

    {Veri alımının durdurulması}
    bRet := DAQAI1.AcquireStop;
    if bRet then
        begin
            ShowMessage(DAQAI1.ErrorMessage);
            Exit;
        end;
    {Devrenin kapatılması}
    bRet := DAQAI1.CloseDevice;
    if bRet then
        begin
            ShowMessage(DAQAI1.ErrorMessage);
            Exit;
        end;
    setlength(a,vsay-1);
    ororn:=0.001;
    ks:=strtoint(knlsys.Text);
end;
procedure TForm1.cmdGainListClick(Sender: TObject);
begin
    frmGainList.Show;
end;
procedure TForm1.cmdSelectDeviceClick(Sender: TObject);
var
    i : Integer;
    j : Integer;
    strGain : WideString;
begin
    DAQAI1.SelectDevice;
    txtDeviceNum.Text := IntToStr(DAQAI1.DeviceNumber);
    txtDeviceName.Text := DAQAI1.DeviceName;
    cmbInputRange.Clear;
    cmbStartChannel.Clear;
    cmbNumChan.Clear;
    gNumOfInputRange := 0;
    gNumOfChan := 0;
    {Devrenin açılması}
    if DAQAI1.OpenDevice then
        begin
            ShowMessage(DAQAI1.ErrorMessage);

```

```

    Exit;
end;

{Giriş aralığı listesinin alınması}
SetLength(strGain, 30);
bRet := DAQAI1.GetFirstInputRange(strGain);
i := 0;
while bRet = False do
begin
    gInputRangeList[i] := strGain;
    cmbInputRange.Items.Add(gInputRangeList[i]);
    i := i + 1;
    bRet := DAQAI1.GetNextInputRange(strGain);
end;
If i <> 0 then
begin
    cmbInputRange.ItemIndex := 0;
    gNumOfInputRange := i;
end;

{Maksimum kanal sayısının alınması}
if DAQAI1.MaxDifferentialChannel > DAQAI1.MaxSingleEndedChannel then
    i := DAQAI1.MaxDifferentialChannel
else
    i := DAQAI1.MaxSingleEndedChannel;
if i = 0 then
begin
    ShowMessage('Function Not Supported');
    DAQAI1.CloseDevice;
    Exit;
end;
gNumOfChan := i;

for j := 0 to i - 1 do
begin
    cmbStartChannel.Items.Add (IntToStr(j));
    cmbNumChan.Items.Add (IntToStr(j + 1));
    gGainCodeSave[j] := 0;
end;
cmbStartChannel.ItemIndex := 0;
cmbNumChan.ItemIndex := 0;
DAQAI1.StartChannel := 0;
DAQAI1.NumberOfChannels := 1;
DAQAI1.CloseDevice;
end;
procedure TForm1.cmdStatusClick(Sender: TObject);
begin
    txtStatus.Text := IntToStr(DAQAI1.AcquireStatus);
end;
procedure TForm1.DAQAI1EventReal(Sender: TObject; DataCount: Integer; Data: OleVariant);
// const b:integer=1;
var
    i,t1,t2,t,a : LongInt;
    j,k : LongInt;
    sa:string[15];
begin
    j := DataCount;
    setlength(dizi,j);
    edit2.Text:=timetostr(time);
    t1:=strtoint(edit1.Text[7]+edit1.Text[8]);
    t2:=strtoint(edit2.Text[7]+edit2.Text[8]);
    if t2<t1 then
        t2:=t2+60;
    t:=t2-t1;
    if t=trtoint(sure.Text) then
    begin
        label18.caption:='Geçen Süre:'+inttostr(t2-t1)+' '+'saniye';
    end;
end;

```

```

edit4.Text:=timetostr(time);
cmdacquirestop.Enabled:=false;
bRet := DAQAI1.AcquireStop;

edit2.Text:=timetostr(time);
if bRet then
begin
  ShowMessage(DAQAI1.ErrorMessage);
  Exit;
end;
{Devrenin kapatılması}
bRet := DAQAI1.CloseDevice;
if bRet then
begin
  ShowMessage(DAQAI1.ErrorMessage);
  Exit;
end;
GroupBox10.Enabled := True;
cmdAcquireStart.Enabled := True;
cmdAcquireStop.Enabled := False;
cmdExit.Enabled := True;
cmdSelectDevice.Enabled := True;
end
else
begin
  for i:=0 to j-1 do
  begin
    dizi[i]:=formatfloat('0.00000000',data[i]);
    sa:=dizi[i];
    a:=pos(',',dizi[i]);
    sa[a]:='.';
    dizi[i]:=sa;
  end;
  if strtoint(knlsys.Text)=3 then
  begin
    i:=0; k:=1; t:=2;
    while (i<=j-3) do
    begin
      lstfaza.Items.Add(dizi[i]);
      lstfazb.Items.Add(dizi[k]);
      lstfazc.Items.Add(dizi[t]);
      i:=i+3; k:=k+3; t:=t+3;
    end;
  end
  else if strtoint(knlsys.Text)=1 then
  begin
    for i:=0 to j-1 do
      lstfaza.Items.Add(dizi[i]);
    end;
  end;
end;
end;
procedure TForm1.DAQAI1Terminated(Sender: TObject);
var
  UserData : OleVariant;
  i : Integer;
begin
  ErrorTimer.Enabled := False;
  GroupBox10.Enabled := True;
  cmdAcquireStart.Enabled := True;
  cmdAcquireStop.Enabled := False;
  cmdExit.Enabled := True;
  cmdSelectDevice.Enabled := True;
  {Veri alımının durdurulması}
  DAQAI1.AcquireStop;
  {Close device}
  DAQAI1.CloseDevice;
end;

```

```

procedure TForm1.FormCreate(Sender: TObject);
begin
knlsys.Enabled:=false;
{Varsayılan devre ayarlarının seçimi }
//FIFO AYARLARI
chkFIFOEnabled.Checked:=true;
txtFIFOSize.Enabled := True;
DAQAI1.FIFOEnabled := True;
txtFIFOSize.Text := IntToStr(DAQAI1.FIFOSize);
// EVENT ACTIVATE AYARLARI
DAQAI1.EventEnabled := True;
chkEventEnabled.Checked:=true;
//Periyodiklik ayarları
chkCyclicMode.Checked:=true;
DAQAI1.CyclicMode := True;
cmdSelectDevice.Click();
{Başlangıç değerlerinin ayarlanması}
txtDeviceNum.Text := IntToStr(DAQAI1.DeviceNumber);
txtDeviceName.Text := DAQAI1.DeviceName;
cmbDataType.ItemIndex := DAQAI1.DataType;
cmbStartChannel.ItemIndex := DAQAI1.StartChannel;
cmbNumChan.ItemIndex := DAQAI1.NumberOfChannels - 1;
cmbTransferMode.ItemIndex := DAQAI1.TransferMode;
cmbClockSource.ItemIndex := DAQAI1.ClockSource;
if DAQAI1.ExtTrigger then
    cmbTrigSource.ItemIndex := 1
else
    cmbTrigSource.ItemIndex := 0;
DAQAI1.SampleRate := 3000;
DAQAI1.NumberOfSamples := 9000;
txtsamplerate.Text:=inttostr(3000);
txtnumofsamples.Text:=inttostr(9000);
end;
procedure TForm1.txtDeviceNumChange(Sender: TObject);
begin
    DAQAI1.DeviceNumber := StrToInt(txtDeviceNum.Text);
    txtDeviceName.Text := DAQAI1.DeviceName;
end;
procedure TForm1.txtNumOfSampleChange(Sender: TObject);
begin
    DAQAI1.NumberOfSamples := StrToInt(txtNumOfSample.Text);
end;
procedure TForm1.txtSampleRateChange(Sender: TObject);
begin
    DAQAI1.SampleRate := StrToInt(txtSampleRate.Text);
    txtnumofsamples.Text:=inttostr(StrToInt(txtSampleRate.Text)*3);
    DAQAI1.NumberOfSamples := StrToInt(txtNumOfSample.Text);
end;
procedure TForm1.chkCyclicModeClick(Sender: TObject);
begin
    if chkCyclicMode.State = cbChecked then
        DAQAI1.CyclicMode := True
    else
        DAQAI1.CyclicMode := False;
end;
procedure TForm1.chkEventEnabledClick(Sender: TObject);
begin
    if chkEventEnabled.State = cbChecked then
        DAQAI1.EventEnabled := True
    else
        DAQAI1.EventEnabled := False;
end;
procedure TForm1.cmbClockSourceChange(Sender: TObject);
begin
    DAQAI1.ClockSource := cmbClockSource.ItemIndex;
end;
procedure TForm1.cmbTrigSourceChange(Sender: TObject);

```



```

begin
    if cmbTrigSource.ItemIndex = 1 then
        DAQAI1.ExtTrigger := True;
    end;
procedure TForm1.cmbStartChannelChange(Sender: TObject);
var
    i, num : integer;
begin
    cmbNumChan.Clear;
    num := gNumOfChan - cmbStartChannel.ItemIndex;
    for i := 1 to num do
        cmbNumChan.Items.Add (IntToStr(i));
    cmbNumChan.ItemIndex := 0;
    DAQAI1.NumberOfChannels := 1;
    DAQAI1.StartChannel := cmbStartChannel.ItemIndex;
end;
procedure TForm1.cmbNumChanChange(Sender: TObject);
begin
    DAQAI1.NumberOfChannels := cmbNumChan.ItemIndex + 1;
    knlsys.Text:=inttostr((cmbnumchan.Itemindex)+1);
    edit1.Enabled:=false;
    edit4.Enabled:=false;
    txtnumofsample.Text:=inttostr(StrToInt(txtSampleRate.Text)*3);
    DAQAI1.NumberOfSamples := StrToInt(txtNumOfSample.Text);
end;
procedure TForm1.ErrorTimerTimer(Sender: TObject);
begin
    txtErrorCode.Text := IntToStr(DAQAI1.ErrorCode);
    txtErrorMessage.Text := DAQAI1.ErrorMessage;
end;
procedure TForm1.FormClose(Sender: TObject; var Action: TCloseAction);
begin
    ErrorTimer.Enabled := False;
    DAQAI1.AcquireStop;
    DAQAI1.CloseDevice;
end;
procedure TForm1.Button2Click(Sender: TObject);
begin
    label25.Caption:="";
    label26.Caption:="";
    label27.Caption:="";
    if strtoint(knlsys.Text)=1 then
        begin
            lstfaza.Items.SaveToFile('C:\MATLAB6p5\work\ucfazakimi\' + edit3.text + 'a.dat');
            label25.Caption:='A fazı Veri Sayısı :'+inttostr(lstfaza.Count);
            memo1.lines.savetofile('C:\MATLAB6p5\work\ucfazakimi\' + edit3.text + 'acıklama.txt');
        end
    else if strtoint(knlsys.Text)=3 then
        begin
            label25.Caption:='A fazı Veri Sayısı :'+inttostr(lstfaza.Count);
            label26.Caption:='B fazı veri sayısı :'+inttostr(lstfazb.Count);
            label27.Caption:='C fazı veri sayısı :'+inttostr(lstfazc.Count);
            lstfaza.Items.SaveToFile('C:\MATLAB6p5\work\ucfazakimi\' + edit3.text + 'a.dat');
            lstfazb.Items.SaveToFile('C:\MATLAB6p5\work\ucfazakimi\' + edit3.text + 'b.dat');
            lstfazc.Items.SaveToFile('C:\MATLAB6p5\work\ucfazakimi\' + edit3.text + 'c.dat');
            memo1.lines.savetofile('C:\MATLAB6p5\work\ucfazakimi\' + edit3.text + 'acıklama.txt');
        end;
    end;
end;
procedure TForm1.Button5Click(Sender: TObject);
begin
    label25.Caption:="";
    label26.Caption:="";
    label27.Caption:="";
    label25.Caption:='A fazı veri sayısı :'+inttostr(lstfaza.Count);
    label26.Caption:='B fazı veri sayısı :'+inttostr(lstfazb.Count);
    label27.Caption:='C fazı veri sayısı :'+inttostr(lstfazc.Count);
end; end.

```

Akım Sensorlarının Özellikleri

- Birincil akım: 5–100 A
- İkincil akım: 25 veya 50 mA
- Besleme gerilimi: +12... +15 V DC
- Sensorlar yatay veya dikey olarak bağlanabilirler.
- Birincil bağlantı sensorda tümleşik bulunmaktadır.
- Sensor ikincil devreye bir konektör veya kablo ile bağlanabilir
- Yüksek aşırı yüklenme kapasitesi
- Yüksek güvenilirlik
- Yüksek doğruluk