

**KOCAELİ ÜNİVERSİTESİ \* FEN BİLİMLERİ ENSTİTÜSÜ**

**SÜREKLİ MIKNATISLI SENKRON MOTOR'UN BULANIK  
MANTIK İLE HIZ KONTROLÜ**

**YÜKSEK LİSANS**

**Elektrik Müh. Aykut TEKER**

**Anabilim Dalı: Elektrik Mühendisliği**

**Danışman: Prof. Dr. Feriha ERFAN KUYUMCU**

**KOCAELİ, 2008**

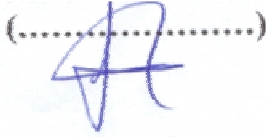
**KOCAELİ ÜNİVERSİTESİ \* FEN BİLİMLERİ ENSTİTÜSÜ**

**SÜREKLİ MIKNATISLI SENKRON MOTOR'UN BULANIK  
MANTIK İLE HIZ KONTROLÜ**

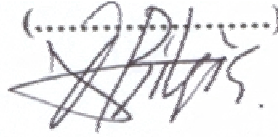
**YÜKSEK LİSANS TEZİ  
Elektrik Müh. Aykut TEKER**

**Tezin Enstitüye Verildiği Tarih : 02.01.2008  
Tezin Savunulduğu Tarih : 30.01.2008**

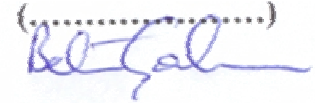
**Tez Danışmanı  
Prof. Dr. Feriha ERFAN KUYUMCU**

(.....)  


**Üye  
Prof.Dr. M. Oruç BİLGİÇ**

(.....)  


**Üye  
Prof.Dr. Bekir ÇAKIR**

(.....)  


**KOCAELİ, 2008**

## **ÖNSÖZ ve TEŞEKKÜR**

Ekte sunulan Yüksek Lisans tez çalışması, Mayıs–Ekim 2007 tarihleri arasında Danimarka Aalborg Üniversitesi'nde ve 2005–2008 tarihleri arasındaki süreçte ise Kocaeli Üniversitesi'nde, Bulanık ve PID olarak bilinen iki farklı kontrol yöntemiyle Sürekli Mıknatıslı Senkron Motor'un hız kontrolü konusunda hazırlanmıştır. Ayrıca olumlu ve olumsuz yönleri de gösterilerek bu kontrol yöntemleri karşılaştırılıp sonuçlarla ilgili yorumlara da detaylı bir şekilde yer verilmektedir. Tezin yapısı toplam olarak 5 bölümden oluşmaktadır.

Danimarka'da bulunduğum süreçte deneysel çalışmamı tamamlamamda bana yardımlarından dolayı Aalborg Üniversitesi Öğretim üyelerinden Sn. Doç.Dr.Ewen RITCHIE'e, verdiği güzel fikirlerinden ve desteklerinden dolayı danışmanım Sn. Prof.Dr.Feriha ERFAN KUYUMCU'ya teşekkür etmek istiyorum. Aalborg Üniversitesi'nde laboratuvar çalışmamda beni hiç yalnız bırakmayan ve yardım eden Aalborg Üniversitesi öğretim üyelerinden Sn. Yrd.Doç.Dr.Lu KAIYUAN'a ve doktora öğrencisi olan Gaizka ALMANDOZ'a teşekkürlerimi sunarım. Ayrıca hayatım boyunca beni destekleyen ve bugünlere getiren babam Hüseyin TEKER'e ve annem Ayfer TEKER'e de sonsuz minnet duygularımı sunarım.

## İÇİNDEKİLER

ÖNSÖZ ve TEŞEKKÜR.....	i
İÇİNDEKİLER .....	ii
ŞEKİLLER DİZİNİ.....	iv
TABLolar DİZİNİ .....	vi
SEMBOLLER.....	vii
ÖZET.....	ix
İNGİLİZCE ÖZET.....	x
1. GİRİŞ .....	1
1.1 Literatür ve Bu Konudaki İleriye Dönük Çalışmalar.....	1
1.2 Problemin Tanımı .....	4
1.3 Akış Şeması.....	5
1.4 Kontrol Stratejisi .....	7
1.5 Referans Akım Üreticinin Modellenmesi.....	8
1.6 Bulanık/PID Hız Denetleyicileri .....	9
1.7 Darbe Genişlik Modülasyonu (PWM) .....	10
1.8 Motor ve İnverter Modeli.....	11
2. SABİT MIKNATISLI SENKRON MOTORLAR (S.M.S.M.) .....	13
2.1 Sabit Mıknatıslı Makinaların Tarihsel Gelişimi.....	13
2.2 Sabit Mıknatıslı Senkron Makinaların Yapısı.....	14
2.3 Park Denklemleri .....	15
2.4 Sabit Mıknatıslı Senkron Makinanın Temel Özellikleri .....	16
2.5 Sabit Mıknatıslı Senkron Makina'nın Olumlu ve Olumsuz Özellikleri .....	17
3. PID VE BULANIK MANTIK TEORİLERİ .....	19
3.1 PID Kontrol.....	19
3.1.1 PID denetleyicisi ile tasarımı .....	20
3.1.2 Doğrusal sistemlerde birim darbe, rampa ve basamak cevapları.....	23
3.2 Bulanık Mantık (Fuzzy) İle Kontrol .....	25
3.2.1 Küme kavramları.....	26
3.2.2 Klasik mantık ile bulanık mantığın karşılaştırılması .....	28
3.2.3 Bulanık mantık kontrol yapısı.....	30
3.2.4 Üyelik fonksiyon çeşitleri .....	31
3.2.5 Bulanıklaştırma ve durulaştırma .....	33
3.2.6 Bulanık denetleyicinin yapısı.....	42
4. SİMÜLASYON VE DENEYSEL ÇALIŞMALAR .....	45
4.1 Matlab Programı Kullanılarak Sürekli Mıknatıslı Senkron Motor'un Matematik Modeli.....	45
4.1.1 Simülasyonda kullanılan bloklar.....	45
4.1.2 Sürekli mıknatıslı senkron motorun simülasyon modeli.....	51
4.2 PID ve Bulanık Denetleyiciler İçin Simülasyon Sonuçları.....	55
4.2.1 PID ve bulanık denetleyiciler ile S.M.S.M.'nin hız denetimi.....	55
4.2.2 Park dönüşüm denklemleri kullanılarak dq-eksenden abc-eksenine ters dönüşüm.....	59
4.2.3 PWM akım kontrol modeli .....	60

4.2.4	Motor parametreleri .....	62
4.2.5	PID blok seçimi.....	63
4.2.6	Bulanık blok .....	64
4.2.7	Ölçüm blokları ve simülasyon sonuçları.....	75
4.3	Deneysel ve Simülasyon Sonuçlarının Karşılaştırılması .....	81
4.3.1	Dspace programında oluşturulan kontrol ve ölçüm blokları.....	82
4.3.2	Deneysel çalışmada karşılaşılan sorunlar.....	86
4.3.3	Deneysel çalışma sonuçları .....	87
5.	SONUÇLAR ve ÖNERİLER.....	92
	KAYNAKLAR .....	96
	EKLER.....	99
	ÖZGEÇMİŞ .....	101

## ŞEKİLLER DİZİNİ

Şekil 1.1: Yüzeyine mıknatıs yerleştirilmiş S.M.S.M.'nin önden görünüşü .....	2
Şekil 1.2: S.M.S.M. ve F.D.A.M. için girişlerine uygulanan gerilim değişimleri .....	3
Şekil 1.3: Çalışmanın akış şeması .....	5
Şekil 1.4: S.M.S.M. sürücüsüne ait PID/Fuzzy kontrollerin blok şema ile gösterilmesi .....	8
Şekil 1.5: PD, PID ve PI kontrolleri .....	9
Şekil 1.6: (a) Anahtarlama, (b) PWM inverter .....	11
Şekil 2.1: (a), (b) dq ve abc stator eksenleri ve Park dönüşümü .....	15
Şekil 2.2: (a) d-eksen akı yolları ve (b) q-eksen akı yolları .....	17
Şekil 3.1: Geri beslemeli sistemin genel gösterimi .....	19
Şekil 3.2: PID denetleyicisinin birim basamak yanıtı .....	21
Şekil 3.3: PID blok genel gösterimi .....	22
Şekil 3.4: Ayırık darbe .....	23
Şekil 3.5: Birim basamak cevabı .....	24
Şekil 3.6: Üyelik fonksiyonlarının karşılaştırılması. (a) klasik kontrol, (b) bulanık kontrol .....	29
Şekil 3.7: Bulanık denetleyicisinin ana yapısı .....	30
Şekil 3.8: Bulanık mantık denetleyici olduğu ve olmadığı durumlardaki çıkış değişimi .....	31
Şekil 3.9: Klasik bulanık kontrol üyelik fonksiyon çeşitleri .....	32
Şekil 3.10: Üyelik fonksiyon ve onun tamamlayıcısı .....	32
Şekil 3.11: (a) Üyelik fonksiyonlarında kesişimin gösterimi, (b) Birleşimin gösterimi .....	32
Şekil 3.12: (a) Üçgen üyelik fonksiyonunun gösterimi, (b) sınırlarının tanımlanması .....	33
Şekil 3.13: Giriş için kullanılan veri üyelik işlevleri, (a) tek darbe girişi, (b) daha genel bulanık girişler .....	34
Şekil 3.14: Durulaştırma .....	36
Şekil 3.15: Klasik küme ile bulanık küme arasındaki fark .....	37
Şekil 3.16: Sentroid veya C-o-A durulaştırması .....	38
Şekil 3.17: (a), (b) C-o-A durulaştırma eğilimi .....	39
Şekil 3.18: Durulaştırma sonuçlarının karşılaştırılması .....	41
Şekil 3.19: Bulanık mantık denetleyici yapısı .....	42
Şekil 4.1: Gerilim kaynakları .....	45
Şekil 4.2: Gerilim sinyallerinin üretilmesi .....	46
Şekil 4.3: dq-abc dönüşüm bloğu .....	47
Şekil 4.4: Motor parametrelerin yazıldığı bloklar .....	48
Şekil 4.5: Elektriksel momente ( $T_e$ ) ait blok .....	48
Şekil 4.6: (a) $\lambda_d$ için blok, (b) $i_d$ için blok .....	49
Şekil 4.7: (a) $\lambda_q$ için blok, (b) $i_q$ için blok, (c) d ve q-eksenlerine ait bloklar .....	50
Şekil 4.8: S.M.S.M.'nin modeli .....	51
Şekil 4.9: (a) İlk 10 saniye için S.M.S.M.'nin hız değişimi, (b) İlk 1 saniye için S.M.S.M.'nin hız değişimi .....	52

Şekil 4.10: Deneysel çalışmada kullanılan inverter ve PWM parçalarının gösterimi	53
Şekil 4.11: (a) Deneysel çalışmada kullanılan S.M.S.M., (b) motora bağlı yükün gösterimi.....	53
Şekil 4.12: (a) Deneysel çalışmada kullanılan deney düzeneğinin gösterimi, (b) makina laboratuvarının gösterimi .....	54
Şekil 4.13: PID ve Bulanık denetleyicilerle ile hız kontrolü .....	56
Şekil 4.14: PID denetleyicisine ait kontrol blokları .....	57
Şekil 4.15: Bulanık denetleyicisine ait kontrol blokları.....	58
Şekil 4.16: (a), (b) Referans akımlar için dq-abc eksen dönüşüm blokları.....	59
Şekil 4.17: (a), (b), (c) Matlab’da yer alan akım kontrol ve inverter blokları, (d) histeresis PWM, akım kontrolü ve anahtarlama mantığı, (e) PWM inverter.....	61
Şekil 4.18: (a), (b), (c) S.M.S.M. ve motor parametrelerine ait bloklar.....	62
Şekil 4.19: (a), (b) Kontrol için PID blok gösterimi .....	63
Şekil 4.20: Kontrol için bulanık blok .....	64
Şekil 4.21: Kurak tablosu için çıkışlar .....	65
Şekil 4.22: Bulanık control için kural editörü .....	70
Şekil 4.23: Bulanık için fis editör .....	71
Şekil 4.24: Giriş–1 (e(t)) için üyelik fonksiyon editörü.....	71
Şekil 4.25: Giriş–2 için (de(t)) üyelik fonksiyon editörü.....	72
Şekil 4.26: Çıkış (u(t)) için üyelik fonksiyon editörü .....	72
Şekil 4.27: Bulanık denetleyici araç çubuğunda yer alan üyelik fonksiyon çeşitleri.	73
Şekil 4.28: Bulanık mantık için kuralların çalışmasını gösteren editör .....	73
Şekil 4.29: Bulanık mantık teorisinde durulaştırma.....	74
Şekil 4.30: Bulanık denetleyicinin yüzeysel görünümü.....	75
Şekil 4.31: (a), (b) Ölçüm blokları .....	76
Şekil 4.32: $t=0.015$ saniye için için referans ve gerçek hızların ( $w_e$ , $w_r$ ) Bulanık kontrol ile PID kontrol için değişimleri .....	76
Şekil 4.33: Bulanık ve PID kontroller için moment ( $T_e$ ) değişimi.....	77
Şekil 4.34: PID ve Bulanık kontrollerde abc eksenlerine ait stator akımları ( $i_{abc}$ ) ....	78
Şekil 4.35: Bulanık ve PID kontrollerde d-ekseni halkalanma aksının ( $\lambda_d$ ) değişimi	79
Şekil 4.36: Bulanık ve PID kontrollerde q-ekseni halkalanma akısının ( $\lambda_q$ ) değişimi .....	80
Şekil 4.37: Bulanık ve PID için d-ekseni akımının ( $i_d$ ) değişimi.....	80
Şekil 4.38: Bulanık ve PID için q-ekseni akımının ( $i_q$ ) değişimi.....	81
Şekil 4.39: Deneysel çalışma için kontrol prensibi.....	82
Şekil 4.40: Dspace kontrol blokları.....	82
Şekil 4.41: Ölçüm ve kontrol blokları.....	83
Şekil 4.42: Hız ve akım kontrol blokları .....	84
Şekil 4.43: (a) Akım kontrolünde kullanılan PI denetleyici bloğu, (b) PI bloğun iç görünümü .....	85
Şekil 4.44: Dspace için deneysel çalışma düzeneği.....	87
Şekil 4.45: (a) 25 kural ile yapılmış simülasyon sonucu, (b) deneysel sonuç için hız çıkışı .....	88
Şekil 4.46: (a) Simülasyon için abc-eksenlerine ait stator akımlarının gösterilmesi, (b) deneysel olarak abc eksenlerine ait stator akımların gösterilmesi.....	89
Şekil 4.47: (a) Simülasyonda d-ekseni akımının gösterilmesi, (b) deneysel olarak elde edilen d-ekseni akımının gösterilmesi .....	90
Şekil 4.48: (a) Simülasyonda q-ekseni akımının gösterilmesi, (b) deneysel olarak elde edilen q-ekseni akımının gösterilmesi .....	91

## TABLolar DİZİNİ

Tablo 1.1: P, I ve D parametrelerin etkisi .....	10
Tablo 3.1: $K_P$ , $K_I$ ve $K_D$ katsayılarının etkileri .....	22
Tablo 3.2: Yükselme zamanı, darbe süresi, taşma ve oturma zamanlarını gösteren denklemler.....	25
Tablo 3.3: Birim darbe, birim basamak ve rampa cevaplarının transfer fonksiyonu şeklindeki gösterimleri .....	25
Tablo 3.4: Bulanık ilişki yöntemleri .....	35
Tablo 4.1: S.M.S.M. parametreleri .....	47
Tablo 4.2: Bulanık denetleyici için kural tablosu .....	67
Tablo 4.3: Bulanık mantık için Matlab içinde kullanılan kurallar .....	74
Tablo 5.1: Bulanık denetleyici için deneysel sonuçlar.....	93
Tablo 5.2: Bulanık ve PID denetleyiciler için simülasyon sonuçları.....	94



## SEMBOLLER

B	: Sürtünme katsayısı, (Nms/rad)
E	: endüklenen gerilim, (V)
e	: izleme hatası
h	: histeresis band genişliği
i	: akım, (A)
J	: atalet momenti ( $\text{kgm}^2$ )
K	: kazanç
L	: endüktans, (H)
M	: taşma miktarı
p	: kutup çifti sayısı
R	: faz başına düşen stator direnci, ( $\Omega$ )
T	: moment, (N.m)
t	: zaman, (s)
u	: sinyal
V	: gerilim, (V)
$\omega$	: açısal hız, (rad/s)
X	: üyelik fonksiyonu
Y	: çıkış işareti
y	: oturma (Steady state value) değeri
$\lambda$	: halkalama akısı, (Wb)

## Alt indisler

a	: a ekseni
b	: b ekseni
<i>b</i>	: band genişliği
c	: c ekseni
D	: türev
d	: d-ekseni
e	: elektriksel
emk	: elektro motor kuvvet
I	: integral
l	: yük
m	: manyetik (mıknatıs)
P	: pozitif
<i>P</i>	: oransal
q	: q-ekseni
r	: yükselme
ref	: referans
s	: yerleşme
U	: negatif
$\infty$	: sınır değer

## Üst İndisler

\* : önceden seçilen değer

## Kısaltmalar

S.M.S.M.	: Sürekli Mıknatıslı Senkron Motor
D.S.P.	: Sayısal İşaret İşleme
F.D.A.M. (B.L.D.C.)	: Fırçasız Doğru Akım Motoru
D.A.	: Doğru Akım
A.A.	: Alternatif Akım
PWM	: Darbe Genişlik Modulasyonu (Pulse Width Modulation)
PI	: Orantı-İntegral
PD	: Orantı-Türev
PID	: Orantı-İntegral-Türev
B.M. (B.L.)	: Bulanık Mantık (Fuzzy Control)
K.M. (C.L.)	: Klasik Mantık
B.M.D. (F.L.C.)	: Bulanık Mantık Denetleyici (Fuzzy Logic Control)
K.M.D. (C.L.C)	: Klasik Mantık Denetleyici (Clasical Logic Control)
E.M.K.	: Elektro Motor Kuvvet
Al-Ni-Co	: Alüminyum Nikel Kobalt
Ü.F. (M.F.)	: Üyelik Fonksiyonu (Membership Function)
PB	: Pozitif Büyük
PO	: Pozitif Orta
PK	: Pozitif Küçük
NK	: Negatif Küçük
NO	: Negatif Orta
NB	: Negatif Büyük
B.A.S. (F.I.S.)	: Bulanık Ara Sistemi (Fuzzy Interface System)
N.N.	: Yapay Sinir Ağları (Neural Network)
G.K. (G.C.)	: Genetik kontrol (Genetic Control)

# SÜREKLİ MIKNATISLI SENKRON MOTOR'UN BULANIK MANTIK İLE HIZ DENETİMİ

AYKUT TEKER

**Anahtar Kelimeler:** Bulanık, Kontrol, Sürekli Miknatıslı Senkron Motor, Modern Kontrol, Klasik Kontrol, S.M.S.M., PID.

**Özet:** Bu çalışmada Sürekli Miknatıslı Senkron Motor'un (S.M.S.M.) hız kontrolü için Bulanık Ara Sistem (F.I.S.) denetleyicisi kullanılarak hata ve hatadaki değişimin kullanılması anlatılmaktadır. S.M.S.M.'ler basit yapılarından dolayı çoğunlukla elektrik sürücülerinde kullanılırlar. Her ne kadar S.M.S.M. sürücü sistemleri motor dinamiklerinden ve yük karakteristiklerinden dolayı doğrusal olmayan karakteristiğe sahip olsalar da, endüstri uygulamalarında güvenilir bir hız kontrolüne ihtiyaç duyarlar. Bu tür problemlerin üstesinden gelmek için sürücü sistemlerinde Bulanık kontrol tasarlanmıştır. Simülasyon ve deneysel çalışmalar gösteriyor ki; bulanık denetleyicileri, S.M.S.M.'nin hız kontrolünde daha etkin ve güvenilirdir.

S.M.S.M.'ler yüksek verim ve güç faktörü, yüksek güç-ağırlık oranı ve yüksek moment-eylemsizlik oranı gibi üstünlüklerinden dolayı, endüstride, özellikle servo sistemlerde, sıklıkla kullanılmaktadır. Bu nedenle üzerlerinde en çok araştırma yapılan alanlardan biridir. Birçok endüstriyel değişken hız uygulamalarda hız ve moment kontrolüne ihtiyaç duyulmaktadır.

S.M.S.M.'lerin performansında makinanın parametre yükündeki değişimler çok etkili olmaktadır. Bu sorunun üstesinden gelmek için Bulanık Mantık Kontrol (B.M.K.), Nöro Ağlar (N.A.) ve Genetik Kontrol (G.K.) gibi birçok modern kontrol stratejileri hız denetiminde kullanılmaktadır.

Bu tezde Bulanık kontrol ile PID kontrol, simülasyon ve deneysel olarak karşılaştırılmaktadır. Bu şekilde her iki yöntemin de üstün ve üstün olmayan yönleri irdelenerek hangisinin ne şekilde seçilebileceği gösterilmektedir.

# THE SPEED CONTROL OF PERMANENT MAGNET SYNCHRONOUS MOTOR (P.M.S.M.) BY FUZZY

AYKUT TEKER

**Keywords:** Fuzzy Control, Permanent Magnet Synchronous Motor, Modern Control, Classical Control, P.M.S.M., PID.

**Abstract:** In this study, a Fuzzy Inference System (F.I.S.) controller using error and derivative of error inputs is proposed for the speed control of a Permanent Magnet Synchronous Motors (P.M.S.M.). The P.M.S.M. is often used in electrical drives because of their simple structures. However, the P.M.S.M. drive systems have a nonlinear characteristic arisen from motor dynamics and load characteristics. So they are required robust speed control in industry applications. To overcome this problem, an F.I.S. is designed and adapted to the drive system. Simulation and experimental results show that the F.I.S. controller is reliable and high effectiveness in the speed control of the P.M.S.M.

Recently, permanent magnet synchronous motors (P.M.S.M.) have been frequently used in industry, especially in servo systems, because of their superiorities as high efficiency and power factor, high power-weight ratio and torque-inertia ratio, etc. For these reasons, P.M.S.M. becomes one of the most important research areas. Industrial variable speed applications demand to controlled variable speed and torque.

However, the performances of the P.M.S.M. are very sensitive to parameter and load variations. To overcome with these problems several control strategies such as Fuzzy Logic Control (F.L.C.), Neural Network (N.N.) and Genetic Control (G.C.) have been proposed for speed control.

In this thesis, Fuzzy and PID controls are compared with simulation and experimentally. It is showed that which control method can be chosen at which status by using their advantages and disadvantages.

## 1. GİRİŞ

Bulanık mantık her gün kullandığımız ve davranışları yorumladığımız yapıya ulaşmamızı sağlayan matematiksel bir çözüm yöntemidir ve temelini doğru ve yanlış değerlerin belirlendiği Bulanık Küme Kuramı (Fuzzy Set Theory) oluşturmaktadır. Yine klasik mantık yönteminde olduğu gibi yapısında “1” ve “0” değerleri vardır [2,3]. Ancak Bulanık mantık yalnızca bu değerlerle yetinmeyip bunların ara değerlerini de kullanarak; örneğin uzaklıkla ilgili bir problemde mesafenin yalnızca yakın ya da uzak olduğunu belirtmekle kalmayıp ne kadar yakın ya da ne kadar uzak olduğunu da belirtir. Klasik mantık sistemindeki yetersizlik, belirsiz ya da karmaşık bir problemin çözümünde etkili olmamasıdır. Hatta bazen bu yöntemle çözümler imkânsız hale gelmektedir. Gerçek Dünya dilini kullanan Bulanık mantık, dilsel niteleyiciler (Linguistic Variables) yardımıyla biraz sıcak, ılık, uzun, çok uzun, soğuk gibi günlük hayatımızda kullandığımız kelimeler ile insan mantığına en yakın doğrulukta elektronik denetimi sağlayabilmektedir.

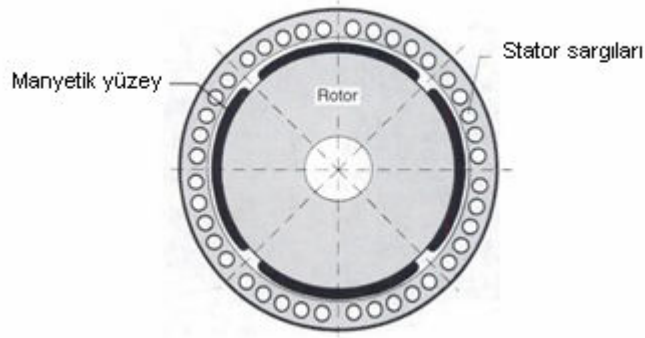
Bulanık mantık, elektrikli ev aletlerinden oto elektroniğine, iş makinalarından üretim mühendisliğine ve endüstride kullanılan kontrol teknolojilerinden otomasyona kadar aklımıza gelebilecek her yerde kendisine uygulama alanı bulabilir. Bu çalışmada, Sürekli Mıknatıslı Senkron Motor’un (S.M.S.M.) hız kontrolü için kullanılan Bulanık Mantık Denetleyicisi’nde (Fuzzy Logic Controller) hız için hata ve hatadaki değişim değişkenleri dikkate alınmaktadır. Gerekli olan referans akım üretilmekte olup, bu sayede makinaya ait hız değişimi kontrol edilebilmektedir [19].

### 1.1 Literatür ve Bu Konudaki İleriye Dönük Çalışmalar

Bilindiği üzere geçmişten günümüze kadar motor uygulamalarında birçok manyetik malzeme kullanılmıştır. 1970’li yılların sonlarına doğru manyetik malzeme olarak kobalt-bor kullanılırken düşük maliyetleri nedeniyle 1980’li yıllarda neodyum-demir-bor manyetik malzemeleri kullanılmaya başlanmıştır. Bu malzemeler kontrol

yöntemindeki yüksek seviyeli sürekli mıknatıslı motorlarındaki duruma benzerlik göstermektedir. Küçük motorlar için birkaç kilowatt'lık sürekli mıknatıslı motorlar, diğerlerine göre daha ekonomiktir [1].

Aynı zamanda güç ve momentteki artışlar ve mekanik olarak hiç fırçaya sahip olmamaları da bu motora ait üstün özellikler arasında sayılabilir. Makinalar içindeki S.M.S.M.'ler otomasyon dünyasında çoğunlukla robot uygulamalarında, yüksek hassasiyetli sürücülerde ve diğer sürücülerde kullanılmaktadır. S.M.S.M.'ler sürekli mıknatısın bulunduğu konuma (yerleştirilme şekline) göre isimlendirilmektedir. Aşağıdaki Şekil 1.1'de, klasik olarak mıknatısın yüzeye yerleştirilmiş motor örneği görülebilir.

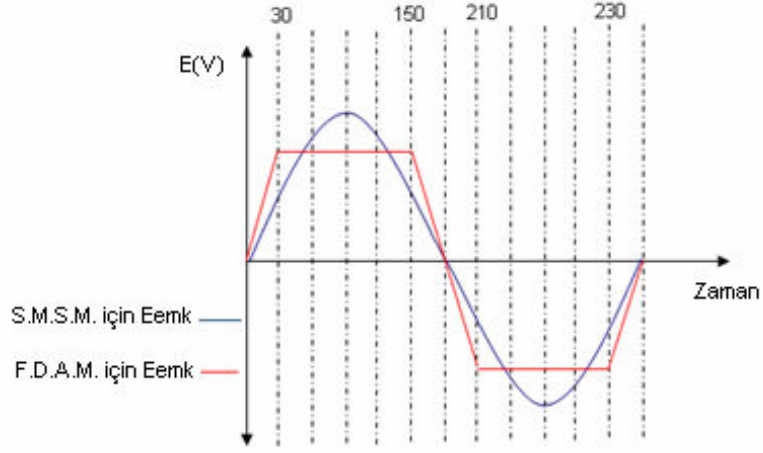


Şekil 1.1: Yüzeyine mıknatıs yerleştirilmiş S.M.S.M.'nin önden görünüşü

Uzun süreli yapılan araştırmalar sonucunda S.M.S.M.'ler ve Fırçasız Doğru Akım Motor'lar (F.D.A.M.), servo uygulamalarında diğer motorlarla yarışabilecek kadar iyi durumda olduğu anlaşılmaktadır [25]. Klasik Sürekli Mıknatıslı Senkron Motor'lar ile Fırçasız Doğru Akım Motor'lar çalışma mantıkları ve yapıları bakımından birbirleriyle benzerlik göstermektedir.

Şekil 1.2'de de gösterildiği gibi bu motorlar için tek fark girişlerine uygulanan gerilimin (e.m.k.) şeklidir. Daha açık ifade edilmek istenirse, S.M.S.M.'lerin girişine sinüs şeklinde gerilim uygulanırken Fırçasız Doğru Akım Motor'larda ise trapez (yamuk) şeklinde gerilim uygulanmaktadır. Bunun dışındaki tüm özellikleri birbirleriyle aynıdır [19,20].

- F.D.A.M.: Giriş uygulanan gerilim trapez  $E_{emf}$  şeklindedir.
- S.M.S.M.: Giriş uygulanan gerilim sinüs  $E_{emf}$  şeklindedir.



Şekil 1.2: S.M.S.M. ve F.D.A.M. için girişlerine uygulanan gerilim değişimleri

Bilindiği üzere senkron motorların denklemlerinde abc-eksenlerden dq-eksenlere dönüşüm yapılmaktadır. Bu sayede hesaplamalar ve kontrol yöntemi daha kolay olmaktadır. Ancak fırçasız doğru akım motorlarında elektro motor kuvvet (e.m.k.) sinüs şeklinde olmadığından bu dönüşümü yapmak pek faydalı olmaz. Sürekli Miknatıslı Senkron Motor ve Fırçasız Doğru Akım Motor'ların doğru akımlı motorlara göre üstün yanları [1]:

- Gürültüsüz çalışmaları,
- Uzun ömürlü olmaları,
- Kıvılcımsız olmaları,
- Yüksek hızlara erişebilmeleri,
- Yüksek güç hassasiyetleri ve daha düşük boyutta olmaları ile
- Daha iyi ısı iletimine sahip olmaları şeklinde sıralanabilir.

S.M.S.M. ve F.D.A.M.'lerin asenkron motorlara göre üstünlükleri aşağıdaki gibi sıralanabilir:

- Yüksek etkiye sahip olmaları,

- Yüksek güç faktörüne sahip olmaları,
- 10 kW'tan düşük uygulamalar için yüksek güç hassasiyetlerinin olması, küçük boyutlu olmaları,
- İyi bir ısı iletimine sahip olmalarıdır.

Bu özelliklerden de anlaşılacağı üzere, bu makinalar düşük güçlü uygulamalar için asenkron motorlara göre daha iyidir.

## 1.2 Problemin Tanımı

Sürekli Mıknatıslı Senkron Motor'lar uzun ömürleri nedeniyle uygulamalarda en çok tercih edilen motorlar arasında yer almaktadır. Bu motorlar, güç elektroniğinde ve kontrol teknolojisinde yüksek performansları nedeniyle kontrol sistemlerinin büyük bir kısmında kullanılabilir. S.M.S.M.'nin yük ve parametre değişkenlerinin çok hassas olmasından dolayı bu sorunun üstesinden gelmek için Bulanık Mantık (Fuzzy Logic), Yapay Sinir Ağlar (Neural Network) ve Genetik Kontrol (Genetic Control) gibi yeni kontrol yöntemleri geliştirilmiştir [2,25]. Basit yapıları ve etkileri nedeniyle bu motorlar sürücü sistemlerinde de kullanılmaktadır.

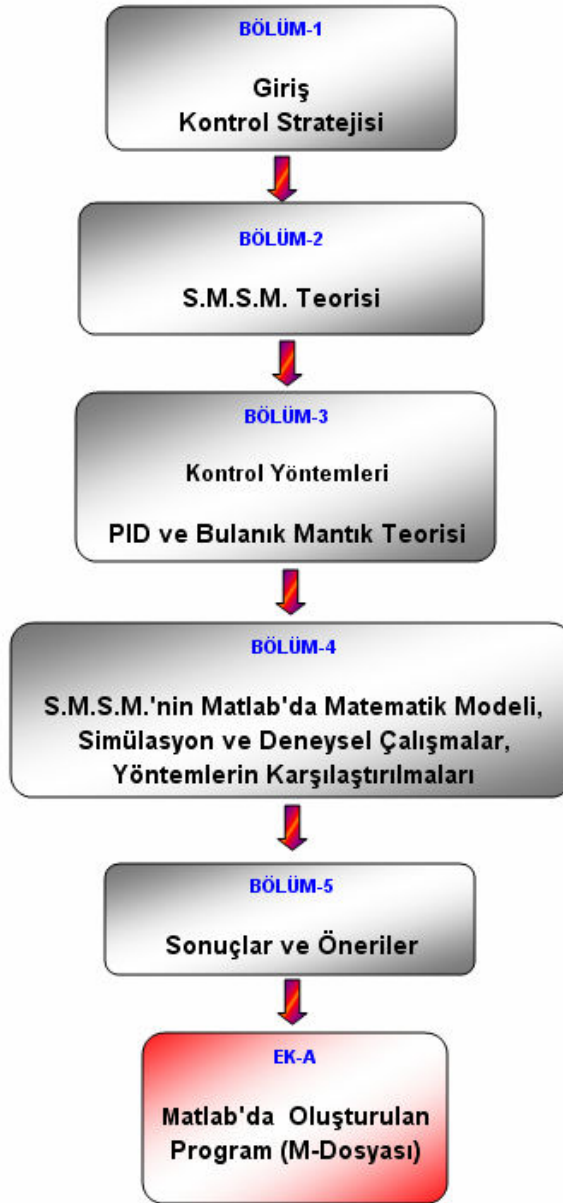
S.M.S.M. sürücü sistemlerin motor dinamiklerinin ve yük karakteristiklerinin doğrusal olmamasından dolayı endüstri uygulamalarında güçlü hız kontrolüne ihtiyaç duymaktadırlar ve bu sorunun üstesinden gelmek için Bulanık mantık yönteminden yararlanılmaktadır. Bulanık mantık kontrol yapısı hassasiyet gerektiren uygulamalarda kullanılmaktadır. Örneğin otomasyon ve kontrol mühendisliğinin olduğu yerler bu kontrol için uygun olmaktadır.

Bu çalışmada da anlatılan Bulanık Mantık Kontrol yöntemi ilk olarak Zadeh [3] tarafından geliştirilmiştir. Bu kontrol yöntemi karışık matematiksel denklemlerden uzak tamamen dilsel ifadelerle yapılmaktadır. Bu yönüyle de diğer kontrol yöntemlerinden ayrılmaktadır. Genel olarak ifade edilmek istenirse, bu kontrol yönteminde değişkenler sayılarla değil sadece; pozitif büyük, sıfır ya da negatif küçük gibi günlük yaşamda kullanılan sözel değişkenlerle tanımlanmaktadır.



### 1.3 Akış Şeması

Aşağıdaki Şekil 1.3'te, yüksek lisans tez çalışmasının genel bir akış şeması gösterilmektedir. Tez, toplam olarak 5 bölümden ve 1 ekten oluşmaktadır. Bölüm 4'de, laboratuvarında yer alan motorun genel özelliklerini gösteren bilgi belgelerine yer verilmektedir. Ek kısmında ise, simülasyon ve deneysel çalışmalarda çizimler için yazılan Matlab programı yer almaktadır.



Şekil 1.3: Çalışmanın akış şeması

Tezin birinci bölümünde, S.M.S.M.'nin geçmişteki ve gelecekteki durumu ifade edilip Bulanık mantık yönteminin bu motorda kullanılmasının sebeplerinden genel hatlarıyla bahsedilmektedir. Bulanık Mantık ve PID denetleyicileri için kontrol stratejilerinden bahsedilip daha sonra da inverter, PWM akım denetleyicisi, referans akım üretici, makina ve bulanık mantık/PID kontrol blokları gibi tüm sistem blokları tek tek açıklanmaktadır.

Tezin ikinci bölümünde ise, manyetik malzemelerin etkileri, kullanılan motor için kontrol yöntemleri ve bu motorun diğer motorlara göre üstün yönleri ile üstün olmayan yönleri anlatılmaktadır.

Tezin üçüncü bölümünde, PID kontrol yöntemi ayrıntılı bir şekilde anlatılıp S.M.S.M.'ye etkileri ifade edilmektedir. Ayrıca bu kontrol yöntemin üstün ve üstün olmayan yönleri de bu bölümde yer almaktadır. PID kısmındakine benzer şekilde, Bulanık mantık yöntemi hakkında bilgiler verilip kontrol amacıyla bu yöntemin nasıl kullanılabileceği açıklanmaktadır. Buna ek olarak da bu yöntemin üstün yanları, kullanım alanları ve olumsuz yönleri sunulup Bulanık mantık kontrol içinde yer alan kural tablosu, üyelik fonksiyonu gibi değişkenler de tanımlanmaktadır.

Tezin dördüncü bölümünde ise, simülasyonda ve deneysel çalışmada kullanılan Sürekli Mıknatıslı Senkron Motor'un matematik modeli Matlab programında oluşturularak parametreleri ile verilmektedir. Kontrol yöntemlerinden PID ve Bulanık denetleyicileri yardımıyla S.M.S.M.'nin hız kontrolüne ait simülasyon gösterilmektedir ve simülasyon sonuçları ile bu iki yöntem karşılaştırılmaktadır. Ayrıca bu makineye ait deneysel hız kontrolü oluşturulmaktadır. Kısacası bu bölümde, simülasyon çalışmasındakine benzer olarak laboratuvar ortamında da uygulanıp deneysel çalışmalarının sonuçları verilmektedir.

En son olarak tezin beşinci bölümde ise, simülasyon ve deneysel sonuçların karşılaştırılması yapılmaktadır. Yapılan simülasyon ve deneysel sonuçlardan hangi yöntemin daha iyi olduğuna karar verilip ileride bu konuyla ilgili ne tür çalışmalar yapılabileceği gibi bilgiler verilmektedir.

## 1.4 Kontrol Stratejisi

S.M.S.M.'ye ait kontrol stratejisinde temel prensip, rotor hızı olan  $w_r$ 'nin kullanıcı tarafından önceden belirlenmiş olan referans hız ile karşılaştırılması sonucu oluşan hata ve hatadaki değişimin incelenmesine dayanmaktadır [1,4]. Düşük hızlardaki çalışmalarda kaçak akıların etkisi pek istenmeyen durumdur. Bu nedenle d-ekseni parametresi olan  $i_d^*$  referans akımı sıfır alınmaktadır ve d-ekseni ait akım denkleminde bakıldığında da görülebileceği üzere manyetik akı nedeniyle d-eksenindeki akıdan manyetik akı etkisi çıkarıldığında akım değeri çok küçük olmaktadır (Denklemler 4.8). Buna dayanarak bu akım değeri sıfır alınabilir [2].

Düşük hız uygulamalarında akı azalmasının önemsizliği nedeniyle aşağıdaki eşitlikler yazılabilmektedir [27]:

$$\lambda_d = \lambda_{af} \quad (1.1)$$

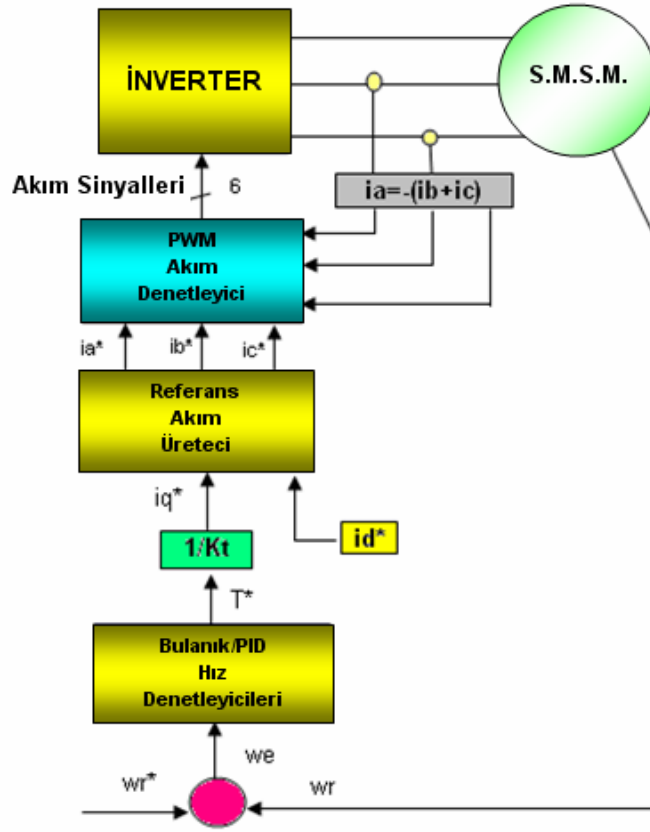
$$T_e = \frac{3p\lambda_{af}i_q}{2} \quad (1.2)$$

Burada doğrudan olarak moment denklemi yazılabilmektedir. Moment eşitliğinde yer alan  $K_t$  sabiti modellemede blok olarak gösterilebilmektedir.

$$T_e = K_t i_q \quad (1.3)$$

$$K_t = \frac{3p\lambda_{af}}{2} \quad (1.4)$$

Ayrıca, dq-eksenlerine ait  $i_d^*$  ve  $i_q^*$  referans akımları kullanılarak, referans akım üreteç kısmı sayesinde  $i_a^*$ ,  $i_b^*$  ve  $i_c^*$  referans akımları üretilmektedir. Referans akımları sinüsoidal dalga formundadır. PWM akım kontrolü bloğunda ise, bu referans akımları ile makineye ait gerçek akımlar karşılaştırılarak inverterdeki (evirici) anahtarlanmaların doğru sırayla yapılması sağlanmaktadır.



Şekil 1.4: S.M.S.M. sürücüsüne ait PID/Fuzzy kontrollerin blok şema ile gösterilmesi

## 1.5 Referans Akım Üretecinin Modellenmesi

Referans akım üreticinin girişinde q-eksenine ait olan referans akımı yer almaktadır. Burada referans momenti ( $T_{ref}$ ) ve  $K_t$  sabitinden yararlanılmaktadır [5].

$$i_{qref}^* = \frac{T_{ref}}{K_t} \quad (1.5)$$

Motorun düşük hızlardaki uygulamalarında d-ekseni akımı ihmal edildiğinden dolayı a,b ve c-eksenleri akımlarına ait aşağıdaki denklemlerde d-ekseni referans akımı yerine sıfır yazılabilmektedir:

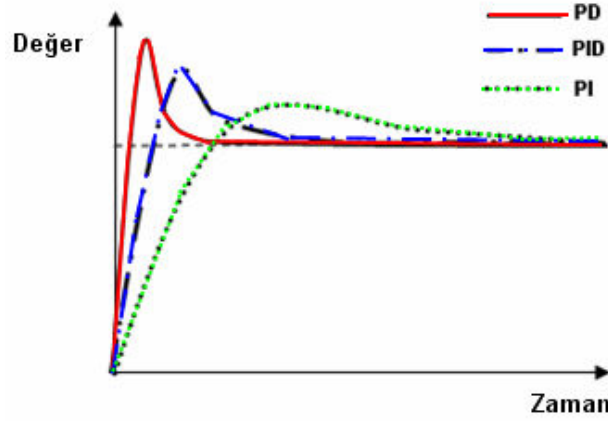
$$i_a^* = i_q^* \cos(\theta_r) + i_d^* \sin(\theta_r) \quad (1.6)$$

$$i_b^* = i_q^* \cos(\theta_r - 120) + i_d^* \sin(\theta_r - 120) \quad (1.7)$$

$$i_c^* = i_q^* \cos(\theta_r + 120) + i_d^* \sin(\theta_r + 120) \quad (1.8)$$

## 1.6 Bulanık/PID Hız Denetleyicileri

Referans hız seçildikten sonra Bulanık ya da PID denetleyicileri kullanılarak motor bu hıza sabitlenebilir. Bu ayarlama PID katsayılarının ( $K_p$ ,  $K_i$ ,  $K_d$ ) değiştirilmesi ile yapılır. Burada tüm katsayıların kullanılması her zaman gerekmebilir. Az sayıda kullanılan katsayılarla istenilen sonuca ulaşılması yeterlidir. Tablo.1.1’de bu katsayıların etkileri görülebilir. Şekil.1.6’da ise PI, PD ve PID denetleyicilerinin çıkışı nasıl etkilediği gösterilmektedir. Uygulayıcı, sistemde nasıl bir gereksinim duyuyorsa buna göre kolaylıkla seçimini yapabilir.



Şekil 1.5: PD, PID ve PI kontrolleri

Şekil 1.5’ten de anlaşılacağı üzere oransal, türev ya da integral ifadelerin sisteme sağladığı yararlar kadar bazı kötü yanları da bulunmaktadır. En göze çarpan, birinde yükselme zamanı azalırken aşma (taşma) miktarı artmaktadır, benzer şekilde aşma miktarı az olan bir sinyalin de yükselme zamanı artmaktadır. Bu nedenle parametrelerin seçiminde istenilen hassasiyet miktarı çok büyük önem taşımaktadır. Tüm parametrelerin kullanılması her zaman gerekmebilir. Eğer çok hassas bir sistemde (kâğıt sarma makinası, tıp sektörü, vb.) bu kontrol yöntemi kullanılacak ise, küçük değişimler önem taşır. Bunun tersine normal bir sistemde çok fazla hassasiyet istenmeyebilir. Bu durum eğer istenilen sonuçlar sadece bir ya da iki parametrenin kullanılması ile sağlanıyorsa üçüncü parametreyi kullanmak gerekmebilir. Yapılan bu çalışmada bir motor kontrolü yapılacağından akım değerleri büyük önem

taşımaktadır. Akım değerinin çok fazla artması istenen bir durum olmaz. Başka bir deyişle, sistemin yükselme zamanını azaltacak derken kısa süreli yüksek genliklere sahip akım değerlerinin üretilmesine neden olunabilir.

Tablo 1.1: P, I ve D parametrelerin etkisi

	Yükselme Zamanı	Taşma	Oturma Zamanı	Oturma Hatası	Kararlılık
Artan $K_P$	Azalı	Artar	Az Artar	Azalı	Kötüleşir
Artan $K_I$	Az Azalı	Artar	Artar	Aşırı Azalma	Kötüleşir
Artan $K_D$	Az Azalı	Azalı	Azalı	Az Değişir	İyileşir

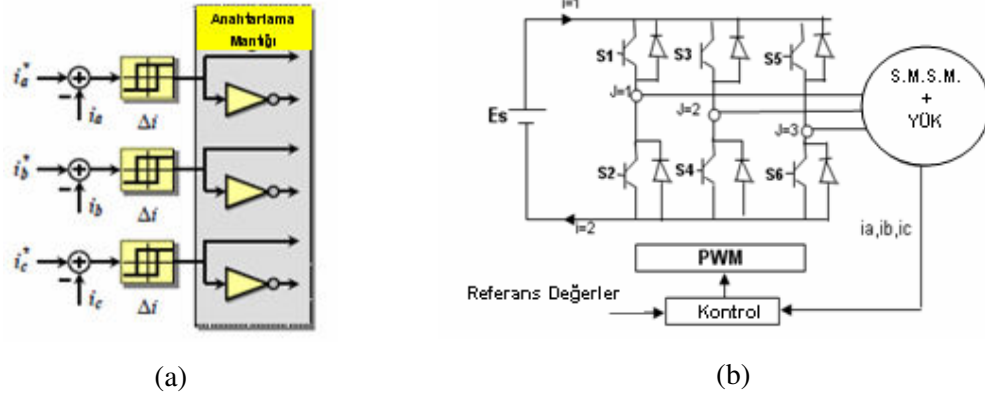
Tablo 1.1'den de anlaşılacağı üzere,  $K_I$  ve  $K_D$  katsayıları sabit iken sadece  $K_P$  katsayısı arttırılırsa yükselme zamanı düşer, aşma miktarı artar, oturma zamanı kısmen artarken değişmezlik biraz kötüleşir [9]. Başka bir deyişle, PID parametrelerinin değerleri değiştirildiğinde sistemin cevap sinyali de buna paralel olarak değişmektedir. Uygulayıcılar, kullandıkları sisteme en uygun şekilde parametrelerin seçilmesini sağlayabilirler. Burada  $K_P$ ,  $K_I$ ,  $K_D$  parametrelerin hepsinin de kullanılacağı anlamına gelmemektedir [28].

Eğer Bulanık denetleyici kullanılırsa kural tablosu oluşturmak gerekmektedir. Oluşturulan kural tablosuna göre motorun hızı istenilen değerde sabitlenebilir, başka bir deyişle kontrolü sağlanabilir. Kural tablosu oluşturulurken pozitif büyük, pozitif küçük, negatif büyük, negatif küçük ya da sıfır gibi sadece mantıksal ifadelerden yararlanılır [3]. Hızdaki hata ve hatanın değişimi incelenerek hızın kontrolü sağlanabilir.

### 1.7 Darbe Genişlik Modülasyonu (PWM)

PWM akım denetleyicisinin yapısında inverter (evirici) yer almaktadır. Gelen sinyallere göre anahtar açılıp kapatılarak akımlar dolayısıyla da gerilimler üretilmiş olur.

- Eğer  $i_a < (i_a^* - hb)$  anahtar 1 AÇIK ve anahtar 4 KAPALI
- Eğer  $i_a > (i_a^* + hb)$  anahtar 1 KAPALI ve anahtar 4 AÇIK
- Eğer  $i_b < (i_b^* - hb)$  anahtar 3 AÇIK ve anahtar 6 KAPALI
- Eğer  $i_b > (i_b^* + hb)$  anahtar 3 KAPALI ve anahtar 6 AÇIK
- Eğer  $i_c < (i_c^* - hb)$  anahtar 5 AÇIK ve anahtar 2 KAPALI
- Eğer  $i_c > (i_c^* + hb)$  anahtar 5 KAPALI ve anahtar 2 AÇIK



Şekil 1.6: (a) Anahtarlama, (b) PWM inverter

## 1.8 Motor ve İnverter Modeli

S.M.S.M.'nin modellenmesinde aşağıdaki hususlar göz önünde tutulmaktadır.

- Doymanın sabit olduğunun kabul edilmesi,
- E.M.K.'nın sinüsoidal etkisi,
- Girdap akımlarının ve histeresis kayıplarının ihmal edilmesi,
- Alan akım dinamiklerinin olmaması,
- Rotorun kafesiz olmasıdır.

Bu koşullar altında S.M.S.M.'ye ait denklemler aşağıdaki gibi yazılabilir [1].

$$V_q = R \cdot i_q + \frac{d\lambda_q}{dt} + \omega_r \cdot \lambda_d \quad (1.9)$$

$$V_d = R.i_d + \frac{d\lambda_d}{dt} - \omega_r.\lambda_q \quad (1.10)$$

$$\lambda_q = L_q.i_q \quad (1.11)$$

$$\lambda_d = L_d.i_d + \lambda_{af} \quad (1.12)$$

Burada  $V_d$  ve  $V_q$  ifadeleri dq-eksenine ait gerilim ifadeleridir,  $i_q$  ve  $i_d$  ise yine bu eksene ait akım ifadelerini göstermektedir.  $L_d$  ve  $L_q$ ; dq-ekseni endüktanslarını,  $\lambda_d$  ve  $\lambda_q$ ; dq-ekseni akı etkisini,  $R$ ; faz başına düşen stator sargısı direncini,  $\omega_r$ ; rotor hızını ve  $\lambda_{af}$  ise, stator akı etkisini göstermektedir.

Elektromanyetik momentin hesabı ise aşağıdaki gibidir:

$$T_e = \frac{3p[\lambda_{af}i_q + (L_d - L_q)i_d i_q]}{2} \quad (1.13)$$

Burada  $p$  ise, kutup çifti sayısını göstermektedir. Genel anlamda moment ifadesi şöyledir:

$$T_e = T_l + B\omega_r + J \frac{d\omega_r}{dt} \quad (1.14)$$

Burada  $T_e$ ; motorda endüklenen momentini,  $T_l$ ; yük momentini,  $B$ ; akı yoğunluğunu ve  $J$  ise atalet momentini temsil etmektedir. Simülasyon dinamiğinde S.M.S.M. için aşağıdaki durum denklemleri yazılabilir.

$$\frac{di_d}{dt} = (V_d - Ri_d + \omega_r L_q i_q) / L_d \quad (1.15)$$

$$\frac{di_q}{dt} = (V_q - Ri_q - \omega_r L_d i_d - \omega_r \lambda_{af}) / L_q \quad (1.16)$$

$$\frac{d\omega_r}{dt} = (T_e - T_l - B\omega_r) / J \quad (1.17)$$



## **2. SABİT MIKNATISLI SENKRON MOTORLAR (S.M.S.M.)**

Bu bölümde, Sabit Mıknatıslı Senkron Motor'un tarihsel gelişimi ve makinanın genel yapısı hakkında kısaca bilgiler verilmektedir. Bilindiği üzere sabit mıknatıslar ile uyarılmış bir elektrik makinasının gerçekleştirilebilirliği, mıknatıs malzemelerindeki gelişmeler, güç elektroniği ve denetimindeki gelişmeler ile artmaktadır. Bu nedenle makinaya ait hız denetimini daha başarılı bir şekilde yapabilmek için muhakkak bu kontrol yapılarının da bilinmesi gerekmektedir.

### **2.1 Sabit Mıknatıslı Makinaların Tarihsel Gelişimi**

Yaklaşık olarak 1930'lu yıllarda Al-Ni-Co alaşımlı daimi mıknatısların bulunması sayesinde, daimi mıknatıslarla uyarılmış büyük makinaların yapımı kolaylaşmıştır. Sabit mıknatıslı malzemelerin bulunması ve bu malzemelerin kullanılması ile uyarılan yeni makinaların geliştirilmesi için çalışmalar uzun yıllardır devam etmektedir. Bu tip makinaların günümüze kadar olan gelişimi daimi mıknatıslı malzemelerin gelişmesine paralel olmaktadır. 1950'li yıllarda Baryum, Stronsiyum ve Kurşun'un demir oksitle oluşturduğu ferrit malzemelerin bulunması ve geliştirilmesi bu mıknatıs malzemelerin elektrik makinalarında kullanılmasına yol açmıştır [8].

İlk olarak 1953 yılında Al-Ni-Co mıknatıslarla uyarılmış senkron makinası F.W.Merril tarafından tasarlanmıştır. Asenkron motorun kafes içindeki rotoruna açılan oluklara Al-Ni-Co mıknatıslardan yapılmış çubuklar yerleştirilmiştir.

J.F.Douglas ise Merril'in geliştirmiş olduğu senkron makinanın akım eğrilerini incelemiştir. Bu inceleme sonucu mıknatıs üretimiyle uygun çalışma karakteristiklerine sahip motorlar üretilip kullanılabileceğini ileri sürmüştür.

K.Falk, iyi özelliklere sahip Al-Ni-Co mıknatıs malzemeler üzerinde çalışmalar yapmıştır. Ayrıca kaçak akı oluklarının motor davranışına olan etkilerini incelemiştir. W.Volkrodt 1960 yılında ferrit mıknatıs malzemeleri kullanarak senkron motor tasarımını gerçekleştirmiştir. Bu tasarımda da asenkron motor esas alınmıştır. Sabit mıknatıs malzemeler rotorun baş kısmına yerleştirilmiştir. Yapılan çalışmalarda manyetik devreyi motorun reaktanslarını stator alanının mıknatıslar üzerindeki etkilerini incelemiştir. Bununla birlikte mıknatıs malzemelerin özelliklerinin ve dolayısıyla çalışma noktalarının değişmemesi için sınır koşulları da verilmektedir [18–22].

## **2.2 Sabit Mıknatıslı Senkron Makinaların Yapısı**

Bilindiği üzere klasik yapıdaki senkron makinaların rotorunda doğru akım (D.A.) ile beslenen uyarma sargıları bulunmaktadır. Bu sargılar sayesinde motor olarak çalışan makinada stator sargılarının yarattığı alternatif alan etkisi ile amortisör sargılarından geçen akım rotora  $F$  kuvveti uygulayarak motorun asenkron çalışmasını sağlar. Senkron motorda moment üretilebilmesi yani senkron çalışabilmesi için frekanstan bağımsız D.A. uyarma gereklidir. Yükten bağımsız sabit hızda çalışması istenen uygulamalarda uyarma sargısı D.A. ile beslenen klasik yapıdaki senkron motorlar kullanılmaktadır. Kompresör, konveyörler gibi değişken hız tahriği gerektiren uygulamalarında ise, senkron motorlar frekans kaynaklı evirici ile kontrol edilmektedir.

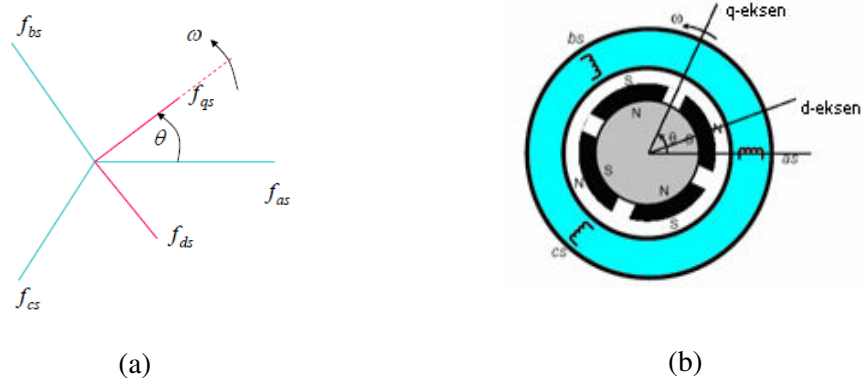
Senkron motorların uyarma şekline bağlı olarak farklı tipleri bulunmaktadır. Bunlar D.A. uyarma sargıları, daimi mıknatıslar ile olabildiği gibi relüktans tipinde de olabilmektedir. Sabit mıknatıslar kullanılarak tasarlanan yeni kuşak makinalar malzeme ve kontrol yöntemlerinin geliştirilmesi ile daha cazip hale gelmektedir. Uyarma akımından dolayı meydana gelen uyarma kayıpları D.A. uyarma sargıları yerine daimi mıknatıs malzemeler kullanılarak azaltılmaktadır.

Sabit mıknatıslı senkron motorun statoru senkron motorun statoru ile aynıdır. Rotoruna ise, sargı yerine değişik tasarım tiplerine göre daimi mıknatıs malzeme kullanılmaktadır. Tasarım özelliklerine göre mıknatıs malzeme, rotorda yüzeye

monte edildiği gibi rotorun içine farklı yapılarda da yerleştirilmektedir. Mıknatıs malzeme bir kez mıknatıslandıktan sonra belirli bir hava aralığında manyetik enerjinin depo edilmesini sağlamaktadır.

### 2.3 Park Denklemleri

Bir makinanın kontrolünde ya da modellenmesinde yapılması gereken ilk adım, eksen dönüşümüdür. Bilindiği üzere abc-eksenlerinde, hesaplama yapılması zor olmakla beraber kontrol işlemi de uygulanamamaktadır. Bu nedenle, 3 tane olan eksenden 2 tane eksene geçiş yapmak gerekir. Bu çalışmada kullanılan motor için de abc-eksenleriyle beraber dq-eksenleri de kullanılmıştır. Bu eksen dönüşümleri ise basit bir eksen dönüşüm denklemiyle yapılabilmektedir. Akının izlediği yola bağlı olarak eksenler d-ekseni ya da q-ekseni olarak isimlendirilir.



Şekil 2.1: (a), (b) dq ve abc stator eksenleri ve Park dönüşümü

Yukarıdaki şekilden de görülebileceği üzere d ve q-eksenleri arasında belli bir açısal fark vardır. Bunlardan d-ekseni vektörü, sargıları dik olarak keserken q-ekseni vektörü ise kutupların arasında geçmektedir. İlk kez eksen dönüşümü Robert H. Park tarafından bulunup “Park Dönüşümü” olarak adlandırılmıştır [4,15].

Dönüşüm denklemleri:

$$f_{qd0s} = K_s \cdot f_{abcs} \quad (2.1)$$

$$(f_{qd0s})^T = [f_{qs} \quad f_{ds} \quad f_{0s}] \quad (2.2)$$

$$(f_{abcs})^T = [f_{as} \quad f_{bs} \quad f_{cs}] \quad (2.3)$$

$$(K_s)^{-1} = \begin{bmatrix} \cos \theta & \sin \theta & 1 \\ \cos\left(\theta - \frac{2\pi}{3}\right) & \sin\left(\theta - \frac{2\pi}{3}\right) & 1 \\ \cos\left(\theta + \frac{2\pi}{3}\right) & \sin\left(\theta + \frac{2\pi}{3}\right) & 1 \end{bmatrix} \quad (2.4)$$

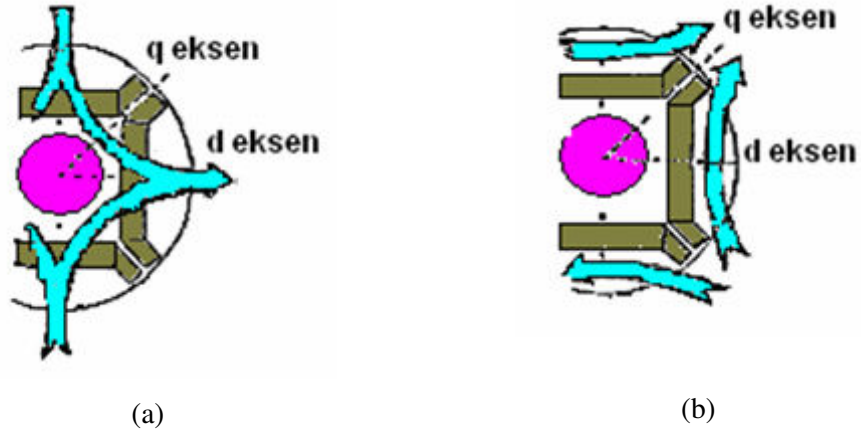
$$\begin{bmatrix} f_a \\ f_b \\ f_c \end{bmatrix} = \begin{bmatrix} \cos \theta & \sin \theta & 1 \\ \cos\left(\theta - \frac{2\pi}{3}\right) & \sin\left(\theta - \frac{2\pi}{3}\right) & 1 \\ \cos\left(\theta + \frac{2\pi}{3}\right) & \sin\left(\theta + \frac{2\pi}{3}\right) & 1 \end{bmatrix} \cdot \begin{bmatrix} f_q \\ f_d \\ f_0 \end{bmatrix} \quad (2.5)$$

$$\begin{bmatrix} f_q \\ f_d \\ f_0 \end{bmatrix} = \frac{2}{3} \cdot \begin{bmatrix} \cos \theta & \cos\left(\theta - \frac{2\pi}{3}\right) & \cos\left(\theta + \frac{2\pi}{3}\right) \\ \sin \theta & \sin\left(\theta - \frac{2\pi}{3}\right) & \sin\left(\theta + \frac{2\pi}{3}\right) \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix} \cdot \begin{bmatrix} f_a \\ f_b \\ f_c \end{bmatrix} \quad (2.6)$$

#### 2.4 Sabit Mıknatıslı Senkron Makinanın Temel Özellikleri

Sabit mıknatıslı senkron motorlar; mıknatıslar rotor içine gömülerek yapılan motorlar ve mıknatıslar rotor yüzeyine yerleştirilerek yapılan motorlar olarak iki tiptedir. Mıknatıslar rotor içine yerleştirilen makinada hava aralığı sabittir. Bu makinalarda değişik tasarımlarda akı yoğunlaştırma yöntemi kullanılır [29].

Bir kutup altındaki akı mıknatıs yüzeyindeki akıdan daha büyük olacak şekilde makina tipleri yapılabilir. Burada temel özelliklerden bahsedilirken radyal mıknatıslanmalı gömülü mıknatıs tasarımı ele alınmaktadır [1,12].



Şekil 2.2: (a) d-eksen akı yolları ve (b) q-eksen akı yolları

Radyal mıknatıslanmalı gömülü mıknatıs tasarımı söz konusu olan 4 kutuplu bir senkron makina ele alınarak akı yolları açıklanabilmektedir. Burada d-ekseni rotor eksenini boyunca olup q-ekseni d-ekseniyle  $45^\circ$  açı yapacak şekildedir. Şekil 2.2(a)'da gösterildiği gibi d-ekseni iki mıknatıs ve iki hava aralığından geçerek q-ekseni ise sadece iki hava aralığından geçerek devresini tamamlamaktadır [10].

## 2.5 Sabit Mıknatıslı Senkron Makina'nın Olumlu ve Olumsuz Özellikleri

Son yıllarda elektrikle tahrik motorlarında çok önemli gelişmeler olmuş ve değişik tasarımlarda motorlar yapılmıştır. Bunların arasında mıknatıslı motorlar da vardır. Geliştirilen mıknatıslı motorların bazıları; Fırçasız Doğru Akım Motor, Sabit (sürekli) Mıknatıslı Senkron Motor, Sabit Mıknatıslı Adım Motor, Sabit Mıknatıslı Lineer Motor şeklinde sıralayabiliriz. Bu motorların yapısı, çalışma ilkesi ve karakteristikleri birbirinden farklıdır.

Sabit mıknatıslı senkron makineler çalışma prensibi ve tasarımı bakımından diğer elektrikli makinalarına göre üstünlükleri bulunmaktadır. S.M.S.M.'ye ait üstün ve üstün olmayan yanları aşağıda sıralanmaktadır [1,18-22].

S.M.S.M'nin olumlu özellikleri aşağıda verilmektedir:

- 1) S.M.S.M.'nin rotorunda fırça ve bilezik yoktur.

- 2) Yüksek hızlı sistemlerde tercih edilirler.
- 3) Şebekeden doğrudan yol alabilen sabit hızlı ya da V/f kontrollü evirici ile beslenen tahrik sistemlerinde kullanılırlar.
- 4) Verimleri yüksektir, çünkü rotorunda sargı yerine mıknatıs bulunmaktadır. Dolayısıyla uyarma kayıpları söz konusu değildir. Rotorunda daimi mıknatıs bulunduğundan uyarma akımına gerek yoktur. Daimi mıknatıs bir kez uyarıldıktan sonra enerji depo edebilen bir malzemedir.
- 5) S.M.S.M.'nin soğutması, rotorunda sargının olmamasından dolayı kolay sağlanabilmektedir.
- 6) Aynı güçteki diğer makinalara oranla hacmi ve ağırlığı azdır.

S.M.S.M.'nin olumsuz özellikleri aşağıda verilmektedir:

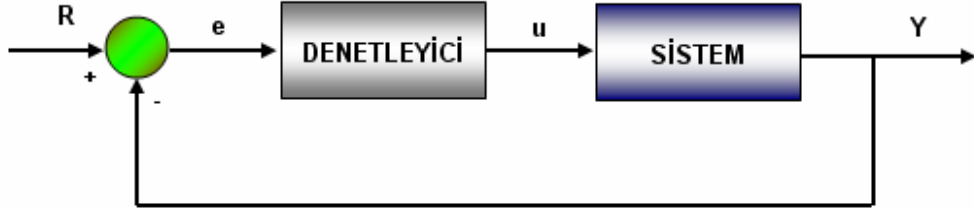
- 1) Şebeke frekansına bağlı olarak sabit bir hızda çalışabilirler. Kontrol edilmeksizin hızın değişimi söz konusu değildir. Uygulama alanları da sabit hızın gerektiği yerlerdir.
- 2) Generatör çalışma durumunda gerilim ayarı, uyarma akımı olmadığından dolayı yapılamaz.
- 3) Kullanılan mıknatıslı malzemelerin, uygun seçilmemesi durumunda zamanla mıknatıslıyetliğini kaybetmeleri makina için olumsuz bir etkidir.
- 4) Bu makinaların güçleri arttıkça boyutlarının da artacağı düşünülürse kütlelerinin de artması maliyet bakımından sakınca yaratmaktadır.

### 3. PID VE BULANIK MANTIK TEORİLERİ

Bu bölümde, çalışmada kullanılan PID ve Bulanık Mantık denetleyicileri hakkında bilgiler, denetleyici tasarımında izlenen yollar ve bu iki kontrol yöntemin karşılaştırılması yer almaktadır.

#### 3.1 PID Kontrol

Bu kısımda, oransal “P”, integral “I” ve türevsel “D” denetleyicilerin karakteristiklerini ve istenilen tepkiyi bulmakta nasıl kullanıldıkları incelenmektedir. Kontrol sistemlerinde genel olarak aşağıdaki gibi geri beslemeli sistemler tercih edilmektedir [9,28].



Şekil 3.1: Geri beslemeli sistemin genel gösterimi

- Sistem: Kontrol edilecek sistemdir.
- Denetleyici: Sistem için gerekli olan uyarıyı sağlar ve sistem davranışını denetlemek için tasarlanmıştır.

Bilindiği üzere, tüm kontrol örneklerinde denetleyici sabit K kazançlı basit bir kuvvetlendiriciden oluşmaktadır. Bu tür bir kontrol işleminde, kontrol işareti denetleyici çıkışına sabit bir oranla aktarıldığından “Oransal Kontrol” olarak adlandırılır. Kontrol sistemlerinde oransal kontrole ilave olarak, giriş işaretinin türevinin ya da integralinden de yararlanılabileceği düşünülebilir. Buna göre, içinde

toplayıcı (toplama veya çıkarma), kuvvetlendirici, zayıflatıcı, türev ve integral alıcı elemanlar bulunan denetleyiciler göz önünde bulundurulabilir. Tasarımcının görevi bu elemanlardan hangilerinin, ne oranda ve ne şekilde bağlanarak kullanılması gerektiğini belirlemektir. Şekil 3.1’de geri beslemeli sistem kullanılmış PID denetleyicinin nasıl çalıştığı incelenmektedir. Kullanılan “e” değişkeni, izleme hatasını ifade etmekte olup istenilen giriş değeri “R” harfi ile ve gerçek çıkış değeri ise “Y” harfiyle gösterilmektedir. İfade edilen “e” hata sinyali PID denetleyiciye gönderilerek denetleyici tarafından bu hata sinyalinin hem türevi hem de integrali hesaplanmaktadır [11,28].

$$u = K_p e + K_I \int e dt + K_D \frac{de}{dt} \quad (3.1)$$

Bu (u) sinyali denetlenen sisteme gönderilerek yeni bir çıkış (y) elde edilmiş olunur. Bu (y) çıkış sinyali algılayıcıya geri gönderilerek yeni bir hata sinyali (e) bulunur. Denetleyici yeni hata sinyaline aynı işlemleri uygular ve işlem böyle devam etmektedir.

### 3.1.1 PID denetleyicisi ile tasarımı

PD denetleyicisinin sisteme zayıflama getirdiği ancak sistemin kararlı hal davranışını etkilemediği bilinmektedir. PI denetleyicisi ise, sınırlı kararlılığı ve aynı zamanda kararlı hal hatalarını düzeltmesine karşılık yükselme zamanını arttırmaktadır. PI ve PD denetleyicilerinin iyi yönlerinden yararlanılarak hepsinin bir arada kullanıldığı PID denetleyicisi tercih edilmektedir. PID denetleyicisinin tasarımı için aşağıdaki yol izlenilebilir;

1. PID denetleyicisi seri bağlı bir PI ve PD kısımlarından oluşur. PID denetleyicisinin transfer fonksiyonu;

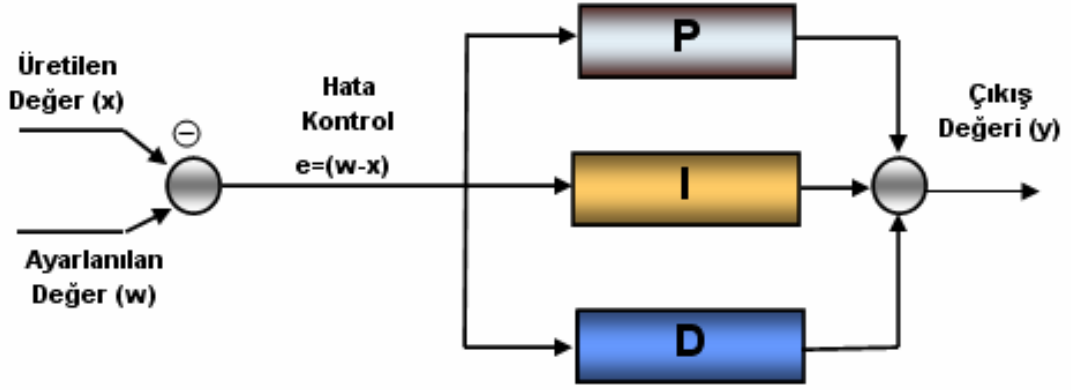
$$G_c(s) = K_p + K_D \cdot s + \frac{K_I}{s} = (1 + K_{D1}s)(K_{p2} + \frac{K_{I2}}{s}) \quad (3.2)$$





İdeal bir PID denetleyicisi için kontrol denklemi aşağıdaki gibi gösterilmektedir:

$$\Delta y = K_p \left( e + \frac{1}{T_i} \int e dt + T_d \frac{de}{dt} \right) \quad (3.6)$$



Şekil 3.3: PID blok genel gösterimi

Tablo 3.1:  $K_p$ ,  $K_I$  ve  $K_D$  katsayılarının etkileri

ETKİ	YÜKSELME ZAMANI	TAŞMA	OTURMA ZAMANI
$K_p$	↘	↗	~
$K_I$	↘	↗	↗
$K_D$	~	↘	↘

PID denetleyicisinde her katsayının ( $K_p$ ,  $K_I$  ve  $K_D$ ) sisteme etkisi, başka bir deyişle yükselme zamanına, taşma miktarına veya oturma zamanına etkileri Tablo 3.1’de gösterilmektedir. Oransal denetleyicilerin ( $K_p$ ), yükselme zamanını azaltma etkisi vardır ama asla tamamen yok etmez.

İntegral denetleyicinin ( $K_I$ ) kararlı hal hatasının çıkarılmasında etkisi vardır, ancak bu da geçici tepkinin daha kötü olmasına sebep olabilmektedir. Türevsel denetleyicinin ( $K_D$ ) sistemin kararlılığının artmasında aşmayı azaltarak etkili olup geçici tepkiyi düzeltmektedir.

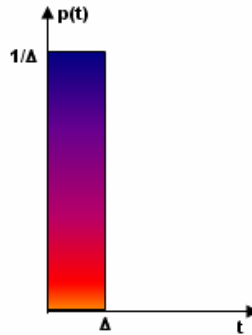
Bu düzeltmeler tam olarak geçerli olmamaktadır. Çünkü  $K_P$ ,  $K_I$  ve  $K_D$  birbirlerine bağımlıdır. Yani değişkenlerden birinin değişimi, diğer ikisinin etkisini değiştirebilir. Bu yüzden gösterilen Tablo 3.1,  $K_I$ ,  $K_P$  ve  $K_D$  değerlerinin belirlenmesinde sadece bir referans olmaktadır [28]. PID denetleyici tasarımında istenilen tepkiyi elde etmek için aşağıdaki adımlar izlenilmektedir:

1. Açık döngü tepkisi bulunur ve ihtiyaçlar belirlenir.
2. Yükselme zamanını düzeltmek için oransal denetleyici eklenir.
3. Aşmayı düzeltmek için türevsel denetleyici eklenir.
4. Kararlı hal hatasını yok etmek için integral denetleyici eklenir.
5. İstenilen tepki elde edilene kadar  $K_P$ ,  $K_I$  ve  $K_D$  ayarlanır.

Hangi denetleyicinin hangi karakteristiği kontrol ettiğini Tablo 3.1'den yararlanılarak bulabilir. Denetleyici tasarımında mümkün olduğu kadar basit tasarıma gidilmelidir. Eğer PI denetleyici ile istenilen tepki sağlanıyorsa, sisteme türevsel denetleyici eklenip sistem karmaşık hale getirilmemelidir.

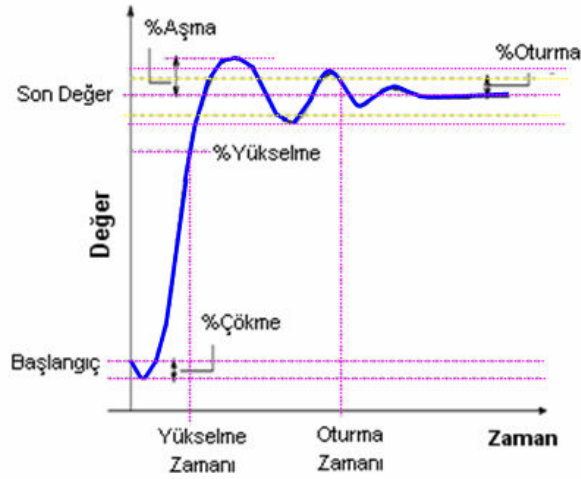
### 3.1.2 Doğrusal sistemlerde birim darbe, rampa ve basamak cevapları

Herhangi bir doğrusal sistemde birim darbe cevabı Şekil 3.4'de gösterildiği gibidir. Burada şeklin alanı '1' olduğu için "birim darbe" ismini almaktadır. Transfer fonksiyonunda hesaplamalar için Laplace dönüşüm denklemlerinden yararlanılmaktadır [25].



Şekil 3.4: Ayrık darbe

- Oturma değeri (Steady state value),  $y_{\infty}$ : Basamak cevabının en son değeridir. Bu değer yaklaşık olarak sabit olduğundan bu şekilde tanımlanmaktadır.
- Yükselme zamanı (Rising time),  $t_r$ : Sisteme ait birim basamak cevabında başlangıç durumundan dengeli durumdaki ilk değere geçinceye kadar (ilk salınımı yapıncaya kadar) geçen süreye verilen isimdir. Burada ilk zaman ifadesi  $k_r y_{\infty}$  şeklinde gösterilir.  $k_r$  değeri sabit bir değer olup yaklaşık olarak 0.9 or 1 aralığında seçilmektedir.
- Çökme (Undershoot),  $M_u$ : Taşma olayına benzemektedir. Taşmadan en büyük farkı ise, negatif yönde olan değişimi göstermesidir. Başlangıçta bazen çıkış ifadesi yükselmeden sıfırın altına doğru azalma yapabilmektedir.



Şekil 3.5: Birim basamak cevabı

- Oturma zamanı (Settling time),  $t_s$ : Yükselme zamanından sonra sistem belli bir süreden sonra kararlı hale gelmektedir. Bu sırada geçen süre oturma zamanı olarak isimlendirilmektedir. Belli bir toleransa ( $\pm\delta$ ) sahiptir ve genellikle bu değer, referans değerinin yüzdesel olarak %2'den %5'e kadar değişebilmektedir.
- Taşma (Overshoot),  $M_p$ : Referans değerinden olan uzaklığı ifade etmektedir. Başka bir deyişle, salınımı ifade etmektedir. Genellikle de yüzdesel olarak gösterilmektedir.

Tablo 3.2: Yükselme zamanı, darbe süresi, taşma ve oturma zamanlarını gösteren denklemler

Yükselme zamanı:	$t_r \cong \frac{1.8}{\omega_n}$
Darbe süresi:	$t_p \cong \frac{\pi}{\omega_d}$
Taşma:	$M_p = e^{-\pi\xi\sqrt{1-\xi^2}}$
Oturma zamanı (%):	$t_s \cong \frac{4.6}{\xi\omega_n}$

Tablo 3.3: Birim darbe, birim basamak ve rampa cevaplarının transfer fonksiyonu şeklindeki gösterimleri

Transfer Fonksiyon	Birim Darbe	Birim Basamak	Birim Rampa
$G(s) = \frac{C(s)}{R(s)}$	$R(s) = 1$	$R(s) = \frac{1}{s}$	$R(s) = \frac{1}{s^2}$

MATLAB’da tek girişli–tek çıkışlı (single input–single output) doğrusal sistemlerin tasarlanması için “sisotool” adında yardımcı bir program kullanılmaktadır. MATLAB ortamında tanımlanan herhangi bir transfer fonksiyonu “>>sisotool” komutuyla, çalıştırılan programın içerisine aktararak ana menüden “Analysis>Response to Step Command” seçilerek sistemin birim basamak cevabı incelenebilir. Programdaki kök yer eğrisine kutup veya sıfırlar eklenerek veya kazanç değiştirilerek istenen çıkış değişimi elde edilebilmektedir [28].

### 3.2 Bulanık Mantık (Fuzzy) İle Kontrol

Geliştirilmesi imkânsız gözükten matematik modellere dayalı sistemler, 1960 yılı ortalarında araştırmacıların önüne problem olarak, artan bir hızda gelmeye başlamıştır. 1965 yılında Kaliforniya Üniversitesi’nden Prof. Lotfi A. Zadeh [3,17]

ilk defa bulanık küme kuramının temel taşı olan “ Yumuşak ” yaklaşım ile sistem tanıma ve tasarımını gerçekleştirmiştir. 1966’da Bulanık Mantık, Bell laboratuvarlarında Dr. Peter Marinos tarafından oluşturulmuştur. 1972 yılında Londra Üniversitesi’nden Prof. E. H. Mamdani bulanık mantık temelli uzman sistemle bir buhar türbinin hızının ve performansının çok başarılı bir şekilde denetlenebileceğini göstermiştir.

Bulanık mantık kuramının ilk önemli endüstri içindeki uygulaması, 1980 yılında Danimarka’daki bir çimento fabrikasında gerçekleştirilerek değirmen içinde çok hassas bir denge ile oranlanması gereken sıcaklık ve oksijen ayarı en uygun biçimde yapılmıştır. Bundan sonra bir başka dikkate değer uygulama ise, Hitachi firması tarafından 1987 yılında Sendai Metro’sunda (Japonya) gerçekleştirilmiştir. Bu sayede trenin istenen konumda durması üç kat daha geliştirilerek kullanılan enerji %10 azaltılmıştır. Bunun üzerine Hitachi firmasının kurduğu benzeri bir sistemin, Tokyo Metro’suna da kurulması için istek gelmiştir.

Bu kadar başarılı uygulamanın ardından Bulanık mantığa olan ilgi artmış, uluslararası bir çalışma ortamı oluşturabilmek amacıyla 1989 yılında aralarında SGS Thomson, Omron, Hitachi, NCR, IBM, Toshiba ve Matsuhita gibi Dünya devlerinin de aralarında bulunduğu 51 firma tarafından LIFE ( Laboratory for Interchange Fuzzy Engineering ) laboratuvarları kurulmuştur. Son yıllarda giderek artan yapay zekâ ile ilgili çalışmalarda, fuzzy sınıflandırıcıların kullanımı ile Bulanık Kontrol’e yeni bir bakış açısı getirmişlerdir.

### **3.2.1 Küme kavramları**

Bu konunun açıklanmasındaki amaç, Fuzzy küme yaklaşımının klasik küme teorisiyle olan ilişkisini incelemektir. Bir evren (u) içinde bulunan elemanlar topluluğuna “küme” denir. Bu küme A olarak isimlendirilebilir.

Klasik kümeler teorisindeki bazı tanımlamaları şöyle sıralayabiliriz;

- Kesişim: İki kümenin aynı elemanlarının oluşturduğu kümeye kesişim denir.

- Bileşim: İki kümenin bütün elemanlarının oluşturduğu kümeye bileşim denir.
- Alt Küme: Bir A kümesi B kümesinin bütün elemanlarına, ayrıca başka elemanlara ve başka elemanlara sahip B, A'nın alt kümesidir.
- Tamlayıcı: Bir A kümesinin içinde olmayan bütün elemanları içeren kümeye A'nın tamlayıcı denir.

Mantık üzerine yapılan çalışmalar için bir kümenin elemanlarını doğru olarak vermek çok önemlidir. Örneğin “Ahmet yaşlıdır” önermesine göre Ahmet, “Yaşlılar Kümesi” olarak tanımlanan kümenin elemanı olmaktadır. Genel bir ifadeyle A isimli bir kümenin elemanı olmak için bu kümenin karakteristik fonksiyonuna sahip olmak gereklidir.

$$\text{Karakter (x)} = \begin{cases} 0 & \text{eğer x, A kümesi elemanı değil ise,} \\ 1 & \text{eğer x, A kümesi elemanı ise,} \end{cases}$$

şeklinde ifade edilir.

Klasik küme teorisinde bir eleman bir kümeye ait olma derecesi “1” ise aittir, “0” ise ait değildir. Bu yüzden 0 ve 1 ile temsil edilen klasik (boolean) mantık, bu düşünce işlemini yeterli bir şekilde ifade edememektedir. İnsan mantığı, açık, kapalı, sıcak, soğuk (0 ve 1) gibi değişkenlerden oluşan kesin ifadelerin yanı sıra, az açık, az kapalı, serin, ılık gibi ara değerleri de göz önüne almaktadır. Ayrıca Zadeh insanların denetim alanında, mevcut makinalardan daha iyi olduğunu ve kesin olmayan dilsel bilgilere bağlı olarak etkili kararlar alabildiklerini savunmuştur. Klasik denetim uygulamalarında karşılaşılan zorluklar nedeniyle, bulanık mantık denetimi alternatif yöntem olarak çok hızlı gelişmiş ve modern denetim alanında geniş uygulama alanı bulmuştur [3].

Bulanık mantığın genel özellikleri;

- Kesin değerlere dayanan düşünme yerine, yaklaşık düşünme kullanılır.

- Her şey  $[0,1]$  aralığında belirli bir derece ile gösterilir.
- Bilgi; büyük, küçük, çok az gibi dilsel ifadeler şeklindedir.
- Bulanık çıkarım işlemi dilsel ifadeler arasında tanımlanan kurallar ile yapılır.
- Her mantıksal sistem bulanık olarak ifade edilebilir.
- Bulanık mantık, matematiksel modeli çok zor elde edilen sistemler için çok uygundur.
- Belirsiz, doğru olmayan, iyi tanımlanmamış, zamanla değişen ve karmaşık sistemlere uygulanabilir. Bu gibi durumlarda uzman kişinin bilgi ve deneyimlerinden yararlanılma yoluna gidilir.

### 3.2.2 Klasik mantık ile bulanık mantığın karşılaştırılması

Bulanık mantık ile klasik mantık arasındaki temel fark, bilinen anlamda matematiğin sadece aşırı uç değerlerine izin vermesidir. Bilinen matematiksel yöntemlerle karmaşık sistemleri modellemek ve kontrol etmek verilerin tam olmamasından dolayı zor olmaktadır. Bulanık mantık, kişiyi bu zorunluluktan kurtarır ve daha niteliksel bir tanımlama olanağı sağlamaktadır. Bir kişi için 38,5 yaşında demek yerine, sadece orta yaşlı demek birçok uygulama için yeterli bir veridir. Böylece dikkate değer ölçüde bir bilgi yüklemesi söz konusu olacak ve matematiksel bir tanımlama yerine daha kolay anlaşılabilen niteliksel bir tanımlama yapılabilecektir. Bulanık mantıkta ise, fuzzy kümeleri kadar önemli bir diğer kavram da dilsel değişken kavramıdır. Dilsel değişken “sıcak” veya “soğuk” gibi ifadelerle tanımlanabilen değişkenlerdir. Bir dilsel değişkenin değerleri fuzzy kümeleri ile ifade edilebilmektedir. Örneğin oda sıcaklığı dilsel değişken için “sıcak”, “soğuk” ve “çok sıcak” ifadelerini alabilir. Bu üç ifadenin her biri ayrı ayrı fuzzy kümeleri ile modellenir [9,23,28].

Bulanık mantığın uygulama alanları çok geniştir. Sağladığı en büyük fayda ise, “insana özgü tecrübe ile öğrenme” olayının kolayca modellenebilmesi ve belirsiz kavramların bile matematiksel olarak ifade edilebilmesine olanak tanınmasıdır. Bu nedenle doğrusal olmayan sistemlere yaklaşım yapabilmek için bu yöntem özellikle uygun olmaktadır. Klasik mantığın temelinde aslında ihtimal hesapları yatar. Yani, bir olayın olabilirliği bu mantıkla çözümlenmeye çalışılır. Sonuç ise, yalnızca “evet” ya da “hayır” ile sınırlıdır. Ancak bulanık mantık bundan tamamen farklıdır ve

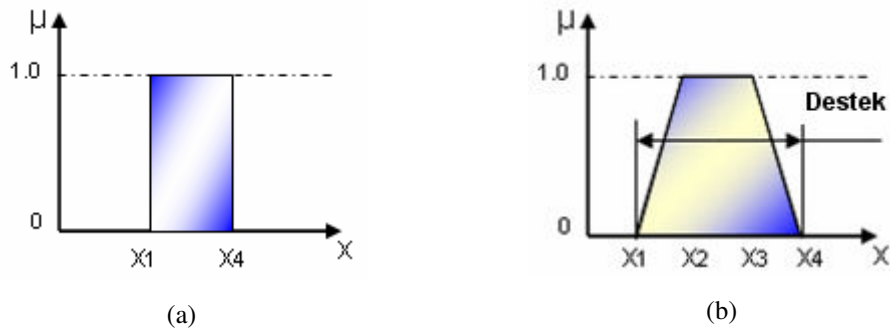


olabilirliđi deđil ne kadar olduđunu verir. Dolayısıyla alınacak cevap evet ya da hayır ile beraber bunların ara deđerlerini de ierir.

Bilindiđi üzere temel olarak iki eřit kontrol yntemi vardır. Kontrol sistemlerinde kullanılan en kolay ve genel yntem klasik kontroldür. Bu kontrol yntemi hassasiyetin pek önemsenmediđi her trl alanda uygulanabilmektedir. Genellikle “dođru” ya da “yanlıř” řeklinde tanımlanmaktadır. Bu nedenle bir Makinanın kontrolnde bařka bir deyiřle hassasiyetin önemli olduđu uygulama alanları iin iyi olmamaktadır.

rnek vermek gerekirse; klasik kontrolde lojik ifadelerle “1” ve “0” ile tanımlayabiliriz. Motorun dnmesi iin “1” verilir durması iin ise “0” lojik giriř verilmesi gerekmektedir. Bu durumda makinanın hızı ne yazık ki kontrol edilemez, motor sadece ya alıřır ya da durar. Ancak modern kontrol yntemlerinden bulanık mantık denetleyicisi kullanılırsa “1” ve “0” lojik ifadeler olmakla beraber bu deđerleri arası da (0.1,0.2,...,0.9) olabilmektedir [13,14].

Denetleyiciyi tasarlayan kiři ihtiya duyduđu sisteme gre bu aralıkların sayısını deđiřtirebilir. Bařka bir deyiřle, makinaya “0” lojik giriř uygulandıđında makina dururken bu deđer arttırıldıđıa ona bađlı olarak da hız da artabilmektedir. Bu sayede makina iin istenilen hız kontrol rahatlıkla yapılır. Buna benzer řekilde endstri iindeki diđer uygulamalarda (ısı, hız, yn, vb) da kullanılabilmektedir.



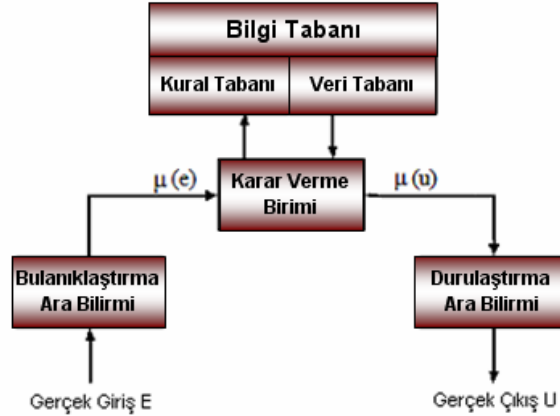
řekil 3.6: yelik fonksiyonlarının karřılařtırılması. (a) klasik kontrol, (b) bulanık kontrol

### 3.2.3 Bulanık mantık kontrol yapısı

Bir bulanık mantık denetleyicisi dört ana fonksiyon bloğundan oluşur [6,7].

- Bulanıklaştırma Arabirim Fonksiyonları (Fuzzification Interface)
- Bilgi tabanı (Knowledge Base)
- Karar verme Lojik Fonksiyonları (Decision Logic)
- Durulaştırma Arabirim Fonksiyonları (Defuzzification Interface)

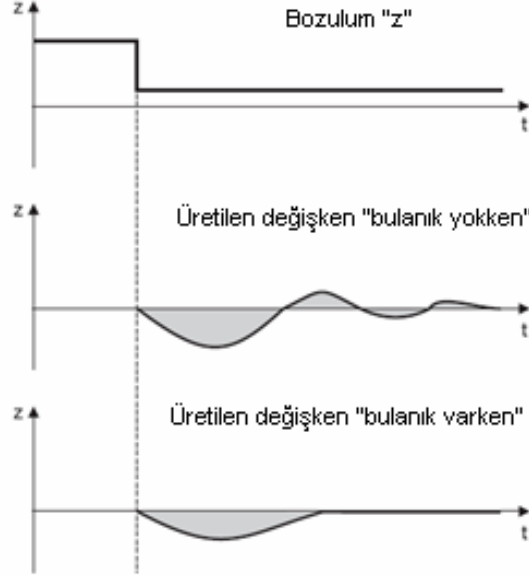
Bulanıklaştırma birimi gerçek sayılardan bulanık sayılara dönüşümü sağlamaktadır. Bilgi tabanı; kuralları ve üyelik fonksiyonlarını içermektedir. Fuzzy karar verme mantığı, sonuç çıkartımı için karar vermede kullanılmaktadır. Son aşama olan durulaştırma bulanık sayılardan tekrar gerçek sayılara ters dönüşümü sağlar. Bulanık denetleyicisinin ana yapısı, Şekil 3.7'de gösterildiği gibidir.



Şekil 3.7: Bulanık denetleyicisinin ana yapısı

Her blok kontrol sistemi içinde büyük önem taşımaktadır. Bu bloklar yardımıyla bulanık denetleyicisi çalışmakta ve oluşturulan kurallar ile istenilen kontrol sağlanabilmektedir. Bir bulanık denetleyicinin temel yapı parametreleri; ölçeklendirme faktörleri, bulanıklaştırma ve durulaştırma metotları, kural tabanı ve üyelik fonksiyon yapı ve gösterimini içermektedir. Bulanık denetleyicinin performansı ile ilgili olarak diğer yapı parametreleri olarak; üyelik fonksiyonlarının

seçimi, durulaştırma metotları, çıkartım mekanizması, kuralların şekli ve türetilmesi, ölçeklendirme faktörleri ve kararlılık analizlerinden de bahsedilebilmektedir [16].



Şekil 3.8: Bulanık mantık denetleyici olduğu ve olmadığı durumlardaki çıkış değişimi

Şekil 3.8’de, de bir sistemde bulanık denetleyicisinin olduğu ve olmadığı durumlarda çıkışı nasıl etkilediğini gösteren şekil yer almaktadır. Bulanık denetleyici kullanılmadığında çıkış ifadesi salınım yaparken denetleyicinin kullanılmasıyla beraber az salınım ve istenilen değerde sabitlenmesi sağlanabilmektedir.

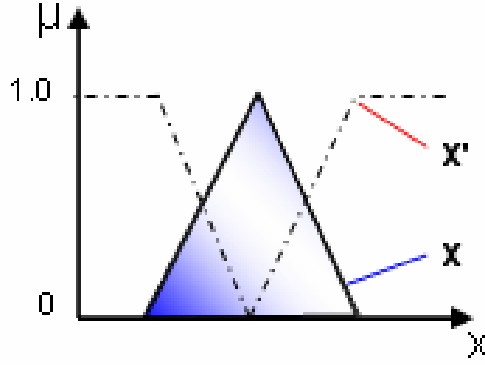
### 3.2.4 Üyelik fonksiyon çeşitleri

Bulanık mantık kontrolde kullanılan üyelik fonksiyonlar tek bir şekle sahip değildir. İhtiyaç duyulan sisteme göre kullanıcı tarafından en uygun şekil seçilebilir. Aşağıdaki şekilde de çeşitli üyelik fonksiyon şekilleri yer almaktadır. Eğer sistemde hassasiyet önemli ise üçgen şekil kullanılırken, değişimin çok hızlı olmasını istenilen yerlerde yamuk kullanılır. Ayrıca bunlara benzer şekilde tek darbe, ayrık gibi gösterimleri de bulunmaktadır [2,3].

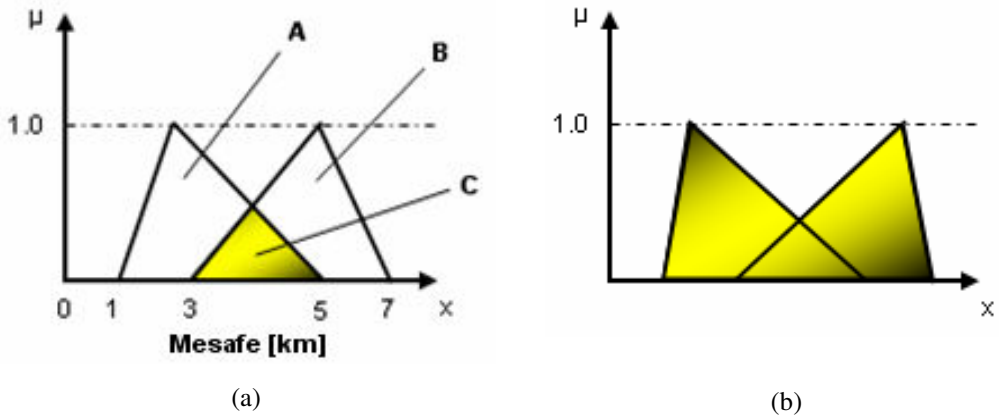


Şekil 3.9: Klasik bulanık kontrol üyelik fonksiyon çeşitleri

Bulanık mantıkta üyelik fonksiyonlarda “değil” ifadesini gösteren şekil de aşağıdaki gibidir. X ifadesine üyelik fonksiyon dersek,  $X' = 1 - X$  ise dışında kalan kısımdır.

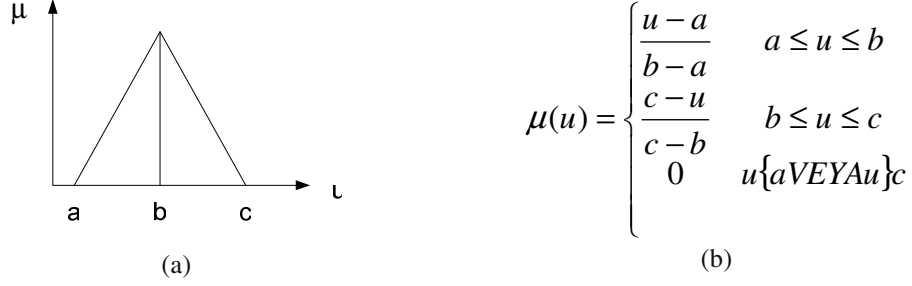


Şekil 3.10: Üyelik fonksiyon ve onun tamamlayıcısı



Şekil 3.11: (a) Üyelik fonksiyonlarında kesişimin gösterimi, (b) Birleşimin gösterimi

Bulanık denetleyicide üyelik fonksiyonları birden fazla olabilmektedir. Bu durumda üyelik fonksiyonları kendi aralarında denklem biçimindeki ifadelerle gösterilir. Örnek verilmek istenirse; küme işlemlerinde olduğu gibi iki üyelik fonksiyonun birleşimi ya da kesişimi yeni bir bölgeyi ifade eder.



Şekil 3.12: (a) Üçgen üyelik fonksiyonunun gösterimi, (b) sınırlarının tanımlanması

Şekil 3.11’de üyelik fonksiyonları için kesişim ve birleşim durumları gösterilmektedir. Küme mantığı ile bakıldığında kesişim ifadesi iki üyelik fonksiyonun kesiştiği, başka bir deyişle ortak buldukları alanı gösterirken birleşim ifadesi toplamını göstermektedir. Bu şekilde olan durumlarda “EĞER A VE/VEYA B İSE ÇIKIŞ C” şeklinde tanımlanmaktadır.

Kesişim durumunda ‘VE’, birleşim durumunda ise bağlayıcı olarak “VEYA” kelimeleri kullanılmaktadır. Üçgen üyelik işlevinde herhangi bir giriş veya çıkış değerinin (u) bulanık kümeye ne kadar üye olduğu, a, b ve c gibi sınır değerleri olan bir işlev için, Şekil 3.12(b)’deki denklem ile bulunur.

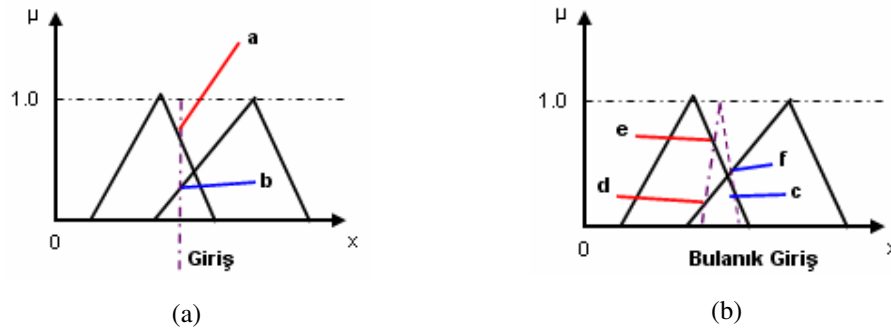
### 3.2.5 Bulanıklaştırma ve durulaştırma

En basit tanımıyla bulanık mantık, yaklaşık akıl yürütme mantığıdır. Geleneksel mantık yapısı olarak tanımlanan sembolik mantık, idealleştirilmiş kavram ve önermelerden çıkarılacak ideal sonuçlarla ilgilenirken, bulanık mantık gerçek dünyadaki bulanıklığı ve belirsizliği ele alarak yaklaşık çözümler üretir.

### 3.2.5.1 Bulanıklaştırma

Bulanık mantık, ikili hesaplama yerine, çok seviyeli hesaplama tekniğini kullanır. Temel yaklaşım, kesin yanlış ve kesin doğru ifadelerinin arasına sonsuz sayıda doğruluk değerini içeren fonksiyon yerleştirmektir. Daha önce de ifade edildiği gibi bu fonksiyona “üyelik fonksiyonu” (membership function) adı verilir. Son olarak, bulanık üyelik eğrilerini gösteren grafik şeklinde bir gösterim mümkündür. Bu gösterme şekli çıkartma işlemini diğer yöntemlerden daha net ortaya koyar.

Bulanıklaştırma, sistemden alınan denetim giriş bilgilerini dile bağlı niteleyiciler olan sembolik değerlere dönüştürme işlemidir. Bulanık mantığın dilsel terimleri genellikle “Eğer-İse” kuralı gibi mantıksal anlamlandırma formu şeklinde tanımlanır. Bu kurallar bulanık üyelik işlevi olarak bilinen değerlerin bir sınırıyla tanımlanır.



Şekil 3.13: Giriş için kullanılan veri üyelik işlevleri, (a) tek darbe girişi, (b) daha genel bulanık girişler

Sistemin verimli çalışması için üçgen, yamuk, çan eğrisi gibi değişik şekillerde üyelik işlevleri seçilebilir. Bu çalışmada, deneme yanılma yoluyla en verimli işlev olarak belirlenen üçgen üyelik işlevi seçilmektedir. Sistemde hata, hatanın değişimi ve çıkış için 3,5 veya 7 gibi tek sayılardan oluşan bulanık değişkenli üyelik işlevleri kullanılabilir. Şekil 3.13'te tek darbe ve üçgen biçiminde uygulanan üyelik işlevler gösterilmektedir. Bulanık kontrolde gerçek olan sayılar bulanıklaştırma kısmından geçerek dile bağlı ifadelere dönüştürülür. Bu sayede çok karmaşık olan sistemler bile kolaylıkla modellenebilmektedir. En büyük üstün yanlarından birisi de hiçbir

matematiksel işleme ihtiyaç duymadan sadece günlük yaşamda kullandığımız (sıcak, soğuk, ılık, hızlı, yavaş, ...vb) dilsel ifadelerle tanımlanmasıdır [2].

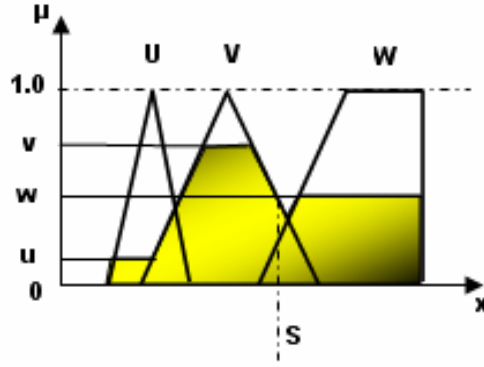
Tablo 3.4: Bulanık ilişki yöntemleri

$R_c$	$\mu A(u) \wedge \mu B(v)$	Mamdani
$R_p$	$\mu A(u) \cdot \mu B(v)$	Larsen
$R_{bp}$	$0 \vee [\mu A(u) + \mu B(v) - 1]$	Sınırlı Çarpım
$R_{dp}$	$\mu A(u), \mu B(v) = 1$ $\mu B(v), \mu A(u) = 1$ $0, \mu A(u), \mu B(v) < 1$	Kesin Çarpım
$R_a$	$1 \wedge [1 - \mu A(u) + \mu B(v)]$	Aritmetik Kural (Zadeh)
$R_m$	$[\mu A(u) \wedge \mu B(v)] \vee [1 - \mu A(u)]$	Maksimum Kural (Zadeh)
$R_b$	$[1 - \mu A(u) + \mu B(v)] \vee \mu B(v)$	Boolean
$R^*$	$1 - \mu A(u) + \mu A(u)\mu B(v)$	Bandler
$R_{\#}$	$[1 - \mu A(u) \vee \mu B(v)] \wedge [\mu A(u) \vee 1 - \mu A(u)] \wedge$ $[\mu B(v) \vee 1 - \mu B(v)]$	Bandler
$R_s$	$1, \mu A(u) < \mu B(v)$ $0, \mu A(u) > \mu B(v)$	Standart Seri
$R_g$	$1, \mu A(u) < \mu B(v)$ $\mu B(v), \mu A(u) > \mu B(v)$	Gödelian Mantiği
$R_{\Delta}$	$1, \mu A(u) < \mu B(v)$ $\mu B(v) / \mu A(u), > \mu A(u) > \mu B(v)$	Gougen

### 3.2.5.2 Durulaştırma

Durulaştırma işleminde ise, bulanıklaştırma sonucunda elde edilen ifadelerin kural tablosundan yararlandıktan sonra küçük bir matematiksel denklemden geçerek tekrar gerçek bir sayının elde edildiği kısım olmaktadır. Aşağıdaki şekilde tek darbe

ve bulanıklaştıma ile elde edilmiş şekil kullanılarak nasıl tekrar gerçek sayıların elde edildiği gösterilmektedir. Bulanık çıkartımın sonucu, bulanık bir kümedir. Bu sonucun tekrar sisteme uygulanması için giriş değeri gibi sayısal değere dönüştürülmesi gereklidir. Bu işleme “durulama” veya “durulaştırma” denir.



Şekil 3.14: Durulaştırma

Durulaştırma işleminde maksimum üyelik, ağırlık merkezi, ağırlık ortalaması gibi değişik yöntemler kullanılır. Bu çalışmada deneme yanılma yoluyla en çok verim alacak şekilde belirlenen ağırlık ortalaması yöntemi kullanılmıştır. Aşağıda farklı durulaştırma yöntemlerinde kullanılan denklemler gösterilmektedir.

(i) Merkez yöntemi ( $s$ ):

- Sürekli dağıtım:

$$s = \int_{x=0}^{x=\infty} x da / \int da \quad (3.7)$$

- Ayrık dağıtım:

$$s = \sum_{i=1}^{i=n} x_i \delta A_i / \sum \delta A_i \quad (3.8)$$

(ii) Maksimum yöntem

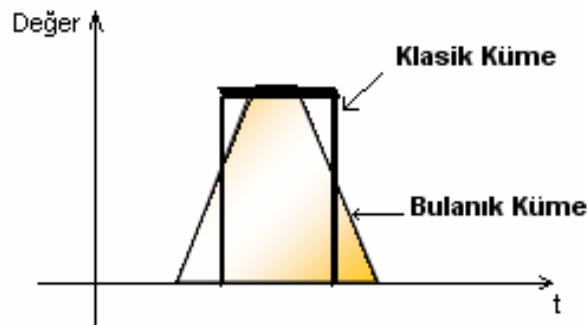
$$s = \sum_{i=1}^{i=n} \mu_i x_i (\max) / \sum \mu_i \quad (3.9)$$



Durulaştırma stratejisinde amaç normal kontrol sistemlerindeki kontrol mantığını en iyi şekilde geliştirmektir. Kontrol edilen sistemin çalışmasında durumlar arası geçişin yumuşak olmasını sağlamak için Fuzzy tablo haritaları hazırlanabilir. Bu haritalarda yine mantıksal ifadelerden oluşmalıdır. Bu ifadeler seçilirken durumu en iyi şekilde ifade edebilen tanımlamalar yapılmalıdır. Bu da maksimum ve minimum durumların ifadelendirilmesidir, ki bu yöntem de Fuzzy kontrolü olarak bilinir.

Bunu matematiksel olarak ifade edersek; bulanık kümeler ve bulanık mantık, insanın düşünme yeteneğini model olarak insanların bilgiyi özetleyebilmeleri ve insan beynindeki verilerden bilgi çıkarabilmelerine dayanır. İnsanların bulanık tarzdaki düşünme işlemleri bulanık kümeler kullanılarak temsil edilmektedir. Bu kümeler, üyelik fonksiyonu olarak adlandırılır ve üyelik değerleri, verilen fonksiyona göre ayarlanan değişkenlerden oluşturmaktadır.

Bulanık küme kavramı, klasik kümelerin bir uzantısıdır. Bu mantığın geçerlilik değeri sözel değişkenlerdir. Klasik kümelerde bir ifade ya doğrudur ya da yanlıştır. Bulanık kümelerde ise, bir ifade 0 ile 1 arasındaki herhangi bir üyelik değerine sahiptir. Şekilde klasik küme ile bulanık kümenin basit olarak karşılaştırılması gösterilmektedir [3]. Bulanık kümeler  $\mu_A(x)$  üyelik fonksiyonu ile temsil edilir.  $\mu_A(x)$  üyelik fonksiyonundaki bir  $x$  noktasının,  $A$  bulanık kümesindeki üyelik derecesidir.  $\mu_A(x)=1$  konumu,  $X$ 'in  $A$  bulanık kümesinin kesin bir elemanı olduğunu tanımlar.

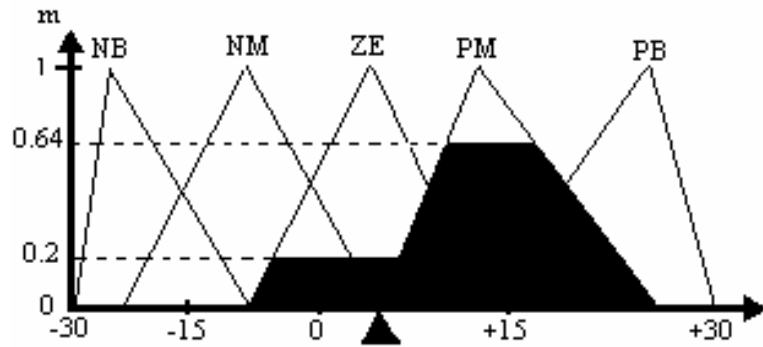


Şekil 3.15: Klasik küme ile bulanık küme arasındaki fark

Benzer bir şekilde  $\mu_A(x)=0$ , X'in A bulanık kümesinin dışında olduğunu belirtir.  $0<\mu_A(x)<1$  dışındaki her değer, X'in A bulanık kümesindeki üyeliğinin belirsiz değerdedir. Bu yüzden kesin olmayan büyüklükler üyelik fonksiyonu tarafından belirlenmiş bulanık kümelerle temsil edilirler. Aşağıda durulaştırma yöntemleri verilmektedir.

1. Ağırlık merkezi yöntemi (center-of-area) ile durulaştırma:

Sentroid yöntemi, genelde “ağırlık merkezi metodu” olarak bilinir. Çünkü çıkış bulanık terimini gösteren birleşik alanın merkez noktasını hesaplar. Şekil 3.16, bir kavramsal çıkış değişkeninin üyelik fonksiyonlarını göstermektedir. Beş üyelik fonksiyonu ve belirlenmiş bir bulanık çıkışının olduğunu varsayarak NB=0.0; NM=0.0; ZE=0.2; PM=0.64; PB=0.0; veya olasılık vektör formunda: {0.0,0.0,0.2,0.64,0.0} olsun [2,3].



Şekil 3.16: Sentroid veya C-o-A durulaştırması

Bu bulanık çıkış, iki farklı durum ZE ve PM’de sıfır olmayan üyelik derecelerine sahiptir. Şekil 3.16, bulanık küme teorisine göre birleştirme operatörü (union) ile oluşturulan ZE ve PM alanlarını göstermektedir. Bu sayede alanın dış hatları bulanık çıkışı meydana getirir. C-o-A durulaştırma metodu bu alanın ağırlık merkezini hesaplar. Sentroidin hesaplaması:

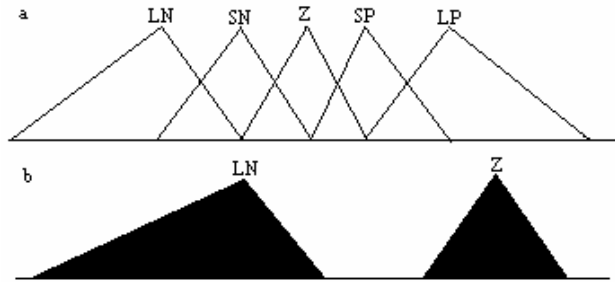
$$\sum_{j=1}^p y_j \mu(y_j) / \sum_{j=1}^p \mu(y_j) \quad (3.10)$$

$\mu(y_j)$ ; bulanık çıkartım sonucu (örneğin, 0.2 ya da 0.64) ile modifiye edilen bir üyelik fonksiyonunun (örneğin, ZE veya PM) alanıdır ve  $y_j$ ; ZE veya PM ayrı üyelik fonksiyonlarının sentroidlerinin konumlarıdır.

$$y_1 = -2.5, y_2 = 8, \mu(y_1) = 0.2, \mu(y_2) = 0.64$$

$$Durulama = [(-2.5)(0.2) + (8)(0.64)] / (0.2 + 0.64) = 4.62 / 0.84 = 5.5$$

C-O-A durulaştırma metodu iki olumsuz yön içermektedir. Bunlardan birisi, üyelik fonksiyonları eşit olmadığındaki durum Şekil 3.17(a)'da açıkça gösterilmektedir. Şekil 3.17(b)'de, LN'nin ZE üyelik fonksiyonundan daha büyük bir alana sahip olduğunu gösteriyor, bu şekilde LN, C-o-A durulaştırmada ZE'den daha büyük bir etkiye sahiptir [2,3].



Şekil 3.17: (a), (b) C-o-A durulaştırma eğilimi

ZE için seçilmiş daha küçük bir alana sahip olmasının nedeni, merkez denge noktası yakınında daha hassas bir kontrol sağlanmak istenmesiyken, bu seçim durulaştırmada bozucu bir etkiye sebep olmuştur. İkinci olarak, C-o-A'nın kötü yanı, istenen sayısal integrasyona göre yüksek bir hesaplama güçlüğüne gereksinim duymasındır.

## 2. Ağırlıklı ortalama (center-of-maximum (C-o-M)) durulaştırması:

Bu metot ile çıkış değişkeninin söylev alanında üyelik fonksiyonların pik değerleri kullanılıyor, üyelik fonksiyonlarının alanları ihmal ediliyor. Çıkış, vektörün sıfır

olmayan deęerleri piklere gre konumlandırılır. Bylece yelik fonksiyonlarının alanları rol oynamaz ve sadece maksimumlar (tek darbe yelik fonksiyonları) kullanılır.

Bulanık ıkış vektr: {NB, NM, ZE, PM, PB}={0.8,0.2,0,0,0} ve yelik fonksiyonların tepe noktaları -2.25(NB), -0.5(NM), 0(ZE), 0.5(PM), 2.25(PB) deęerlerinde meydana gelir. Hesaplama Őyle devam eder:

$$\sum_{j=1}^p y_j \mu(y_j) / \sum_{j=1}^p \mu(y_j) \quad (3.11)$$

$$\begin{aligned} \text{Durulama} &= (0.8)(-2.25) + (0.2)(-0.5) + (0)(0) + (0)(0.5) + (0)(0.5) + (0)(2.25) \\ \text{Durulama} &= -1.9/1.0 = -1.9 \end{aligned}$$

Birinci metodun eŐitlięi yelik fonksiyonlarının alanlarını, ikinci metod ise, sadece maksimumları kullanmaktadır. Beklenildięi gibi, elde edilen durulaŐtırma sonuları farklı olacaktır.

### 3. En byklerin ortalaması (mean-of-maximum (M-o-M)) durulaŐtırma metodu:

Bu metod ile tm maksimumların ortalaması alınır:

$$\max(\mu_1) / k \quad (3.12)$$

Burada  $\max(\mu_1)$  en yksek yelik dereceli bulanık ıkış terimidir. k, bu terimlerin sayıdır. M-o-M metodunda durulaŐtırma iin kullanılan sadece bir maksimum vardır. M-o-M yaklaŐımı en makul zm olarak sylenir.

### 4. Uygulama trne gre durulaŐtırma yntemlerinin seimi:

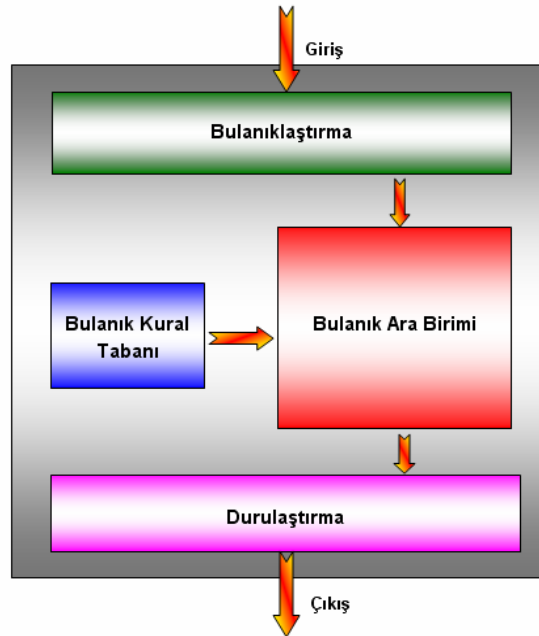
Sunulan durulaŐtırma yntemlerinden hangisinin en iyi olduęuna karar verilebilmesi iin kullanıcının ilgilendięi sorunla iliŐkili olarak bazı olayların nceden bilinmesi gereklidir. Bu bilgilerin baŐında incelenen olayın srekli olup olmadıęı gelir. Srekli



Bir giriş deęişkeninin sonsuz küçük bir deęişimi herhangi bir çıkış deęişkeninde ani bir deęişime neden olmadıysa, durulaştırma süreklidir. Burada, M-o-M metotları süreksiz, C-o-M ve C-o-A durulaştırma metotları ise süreklidir, çünkü girişlerdeki küçük bir deęişim için en iyi çözüm asla farklı bir değere atlayamaz.

### 3.2.6 Bulanık denetleyicinin yapısı

Bulanık mantık ilkelerinin klasik kümelerden temel farkı, bir elemanın herhangi bir kümeye ait olması konusunda verilecek yanıtın klasik kümelerdeki gibi “evet” ya da “hayır” gibi keskin olmayıp, bu elemanların ilgili kümeye ait olma olasılığının 0 ile 1 arasında değerler alabilen sürekli bir üyelik fonksiyonu ile ifade edilmesidir. Herhangi bir elemanın üyelik fonksiyonundan aldığı deęer üyelik derecesi olarak adlandırılır [24].



Şekil 3.19: Bulanık mantık denetleyici yapısı

Bulanık küme teorisinde üyelik derecesinin 0 ile 1 arasında deęerler alması, sözel bilgilerin, problemlerin çözümü sırasında sayısal verilerle birlikte kullanılmasını mümkün kılmaktadır. Sözel ifadelerin bulanık modellere katılması bulanık mantığın dięer yöntemlerden en büyük farklılığıdır. Bulanık sistemler genel olarak, mevcut

verilerden seçilen girdi değişkenlerinden çıktı değişkenlerinin elde edilmesini sağlamak amacıyla bulanık küme ilkelerini kullanan sistemlerdir. Bulanık sistemlerin en büyük üstünlüğü insan deneyimlerinin ve sözel verilerin bulanık modele katılması ile çözüme ulaşılmasıdır.

Bulanık model (bulanık çıkartım sistemi), bulanık “Eğer İse” kuralları adı verilen bulanık kurallara dayanan sistemlerdir. Bulanık modelin temeli, bulanık “Eğer İse” kurallarından anlaşılacağı üzere, temel ve sonuca ait kısımlardan oluşmaktadır. Temel kısımda sonuca sebep olan giriş değişkenleri ve bunlar arasındaki mantıksal ilişkiler, sonuca ait kısımda ise bu giriş değişkenlerine bağlı olarak ortaya çıkan sonuç değişkenleri yer alır. Genel olarak bulanık kurallar aşağıdaki formdadır;

- Kural 1: Eğer  $x=A_1$  ve  $y=B_1$  İse  $z=N_1$
- Kural 2: Eğer  $x=A_2$  ve  $y=B_2$  İse  $z=N_2$

Burada  $x$  ve  $y$ , temel kısımlardaki giriş değişkenlerince tanımlanan koşullar,  $z$  ise sonuca ait kısımdaki çıktı değişkenlerince tanımlanan sonuçlardır. Şekil 3.19’da genel bir bulanık model sistemin yapısı gösterilmektedir [5,6].

- 1) Genel Bilgi Tabanı Birimi: İncelenecek olayın etkilediği girdi değişkenlerini ve bunlar hakkındaki tüm bilgileri içerir. Genel veri tabanı denmesinin sebebi buradaki bilgilerin sayısal ve/veya sözel olabilmesidir.
- 2) Bulanıklaştırıcı: Sayısal girdi değerlerini sözel olarak nitelendirilmiş bulanık kümelerdeki üyelik derecelerine atayan bir işlemcidir.
- 3) Bulanık Kural Tabanı Birimi: Veri tabanındaki girişleri çıkış değişkenlerine bağlayan mantıksal olarak yazılabilen kuralların tümünü içerir. Bu kuralların yazılmasında sadece girdi verileri ile çıktılar arasında olabilecek tüm ara (bulanık küme) bağlantıları düşünülür. Böylece, her bir kural girdi uzayının bir parçasını çıktı uzayına mantıksal olarak bağlar. İşte bu bağlantıların tümü kural tabanını oluşturur.

- 4) **Bulanık Çıkarım Motoru Birimi:** Bulanık kural tabanında giriş ve çıkış bulanık kümeleri arasında kurulmuş olan ilişkilerin hepsini bir arada toplayarak sistemin tek çıkışlı davranmasını temin eden işlemler topluluğunu içeren bir mekanizmadır. Bu motor her bir kuralın çıkarımlarını bir araya toplayarak tüm sistemin girdiler altında nasıl bir çıktı vereceğinin belirlenmesine yarar.
- 5) **Durulaştırma:** Bulanık işlemler sonucu elde edilen bulanık çıkarım sonuçlarını keskin sayısal çıkış değerlerine dönüştürür.
- 6) **Çıktı Birimi:** Bilgi ve bulanık kural tabanlarının bulanık çıkarım motoru vasıtası ile etkileşimi sonunda elde edilen çıktı değerlerinin topluluğunu belirtir.

Kural tabanında denetim amaçlarına uygun dile bağlı denetim kuralları bulunur. Denetim sürecinde bu kurallar kullanılarak bulanıklaştırma, bulanık çıkarım ve durulaştırma aşamalarında bu kural tablosundan faydalanılır.



## 4. SİMÜLASYON VE DENEYSEL ÇALIŞMALAR

Bu bölümde, ilk olarak Sürekli Mıknatıslı Senkron Motor'un matematik modeli yer almaktadır. Matematik model kullanılarak hız kontrolü için PID ve Bulanık mantık yöntemlerin sonuçları verilmektedir.

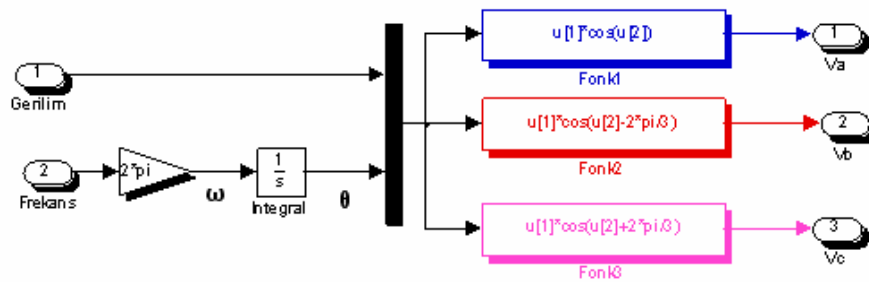
### 4.1 Matlab Programı Kullanılarak Sürekli Mıknatıslı Senkron Motor'un Matematik Modeli

Bu kısımda, Sabit Mıknatıslı Senkron Motor'un denetim yapısında kullanılan bloklar gösterilmektedir. İstenilirse motor için farklı parametreler de seçilebilmektedir. Bu çalışmada, bulanık mantık ile kural tablosu oluşturulurken bu parametreler de göz önünde bulundurulmaktadır. Başka bir deyişle, eğer seçilen parametreler değiştirilirse çıkış da değişime uğrayacağından bulanık mantık ile program yazılacağı zaman mutlaka makineye ait parametrelerin de bilinmesi gerekmektedir.

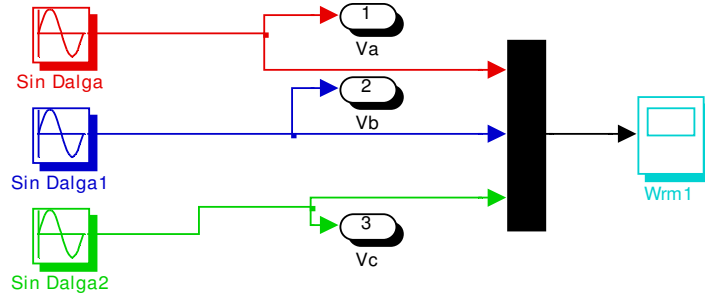
#### 4.1.1 Simülasyonda kullanılan bloklar

##### 4.1.1.1 Gerilim kaynak blokları ( $V_{abc}$ )

Gerilim kaynak bloklarının iki farklı şekilde yapımı mümkündür. Sadece gerilim ve frekans değişkenleri ile yapılabileceği gibi 3 farklı gerilim girişi ile de sağlanabilir.



Şekil 4.1: Gerilim kaynakları



Şekil 4.2: Gerilim sinyallerinin üretilmesi

$$v_a = v_{ref} \cdot \cos(2\pi f_{ref} t - 0) \quad (4.1)$$

$$v_b = v_{ref} \cdot \cos(2\pi f_{ref} t - \frac{2\pi}{3}) \quad (4.2)$$

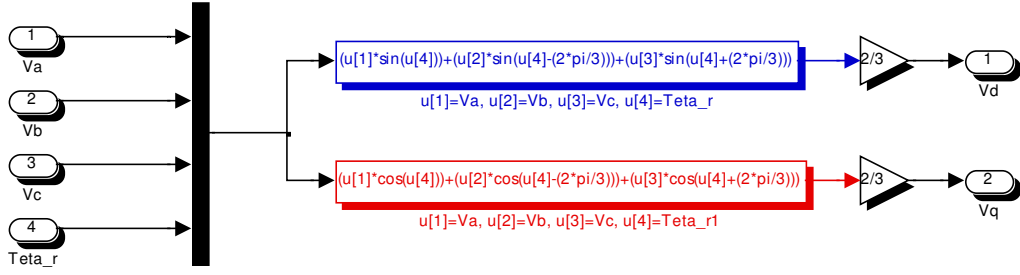
$$v_c = v_{ref} \cdot \cos(2\pi f_{ref} t + \frac{2\pi}{3}) \quad (4.3)$$

S.M.S.M. için hazırlanan matematiksel modelde motorun beslemesi için yukarıda görülen matematiksel denklemler kullanılmaktadır. Burada istenilirse tek bir besleme bloğu seçilip diğerleri de bu blok kullanılarak sadece faz açısının değiştirilmesi ile elde edilebilir. Yapılan bu çalışmada 3 faz için gerekli besleme, kontrol bloklarının sonucunda elde edildiği kabul edilmektedir.

#### 4.1.1.2 dq-abc dönüşüm denlemlerini kullanarak abc eksenlerinden ( $V_{abc}$ ) dq eksenlerine geçiş

Eksen dönüşümü için (abc-dq) 'Park Dönüşüm Matrisi' kullanılmaktadır. Aşağıdaki denklemde bu dönüşüm matrisi gösterilmektedir.

$$\begin{bmatrix} v_q \\ v_d \\ v_0 \end{bmatrix} = \frac{2}{3} \begin{bmatrix} \cos \theta & \cos(\theta - \frac{2\pi}{3}) & \cos(\theta + \frac{2\pi}{3}) \\ \sin \theta & \sin(\theta - \frac{2\pi}{3}) & \sin(\theta + \frac{2\pi}{3}) \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix} \cdot \begin{bmatrix} v_a \\ v_b \\ v_c \end{bmatrix} \quad (4.4)$$



Şekil 4.3: dq-abc dönüşüm bloğu

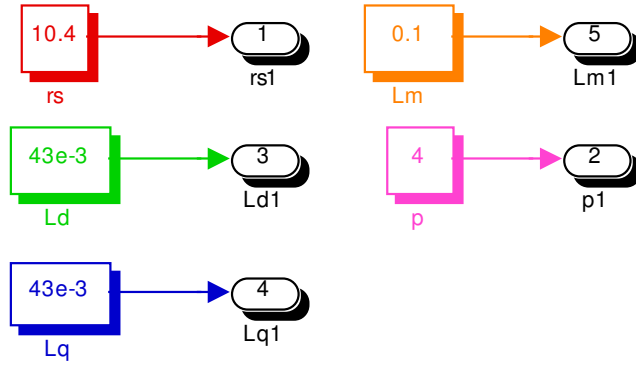
#### 4.1.1.3 Makina parametreleri

Bu çalışma için aşağıda gösterilen motor parametreleri kullanılmaktadır. Simülasyon sonucunun deneysel çalışmalarla da kıyaslanacağı göz önüne alınarak tamamen gerçek makina parametreleri tercih edilmiştir.

Tablo 4.1: S.M.S.M. parametreleri

PARAMETRELER	AÇIKLAMA	DEĞER
Rs	Stator faz direnci ( $\Omega$ )	10.4
p	Kutup çifti sayısı	4
Ld	d-eksen Endüktansı (mH)	43
Lq	q-eksen Endüktansı (mH)	43
$\lambda_m$	Manyetik Halkalama Akısı (Wb)	0.1
J	Eylemsizlik ( $\text{kgm}^2$ )	$0.94 \cdot 10^{-4}$
B	Sürtünme katsayısı (Nms/rad)	$289 \cdot 10^{-6}$

Simülasyon ve deneysel çalışmalarda makina parametreleri Matlab programı içinde hazır bloklar içinde tanımlanmaktadır. Bu sayede, makina parametreleri istenildiği gibi değiştirilebilmesine kolaylık sağlamaktadır [26].



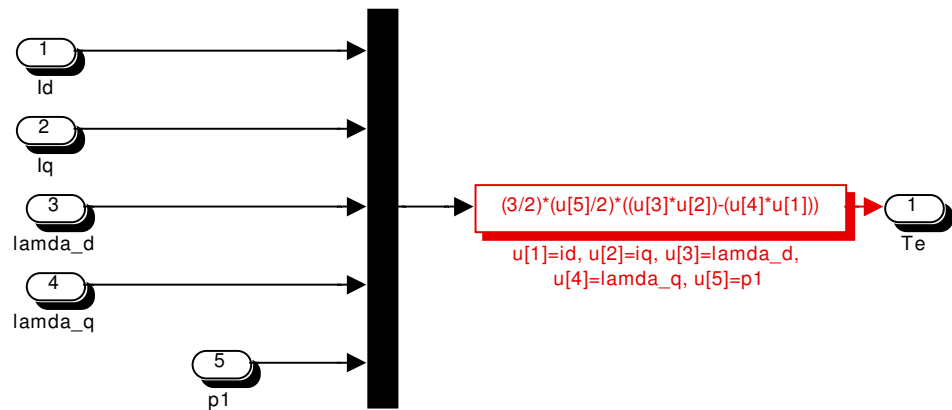
Şekil 4.4: Motor parametrelerin yazıldığı bloklar

#### 4.1.1.4 Moment denklemi

Elektriksel moment ve buna bağlı olarak da hız gibi değişkenler, yazılan denklemler kullanılarak hesaplanabilir;

$$T_e = \frac{3}{2} p (\lambda_d i_q - \lambda_q i_d) \quad (4.5)$$

$$\omega_m = \int \left( \frac{T_e - T_L - B \omega_{rm}}{J} \right) dt \quad (4.6)$$



Şekil 4.5: Elektriksel momente (Te) ait blok

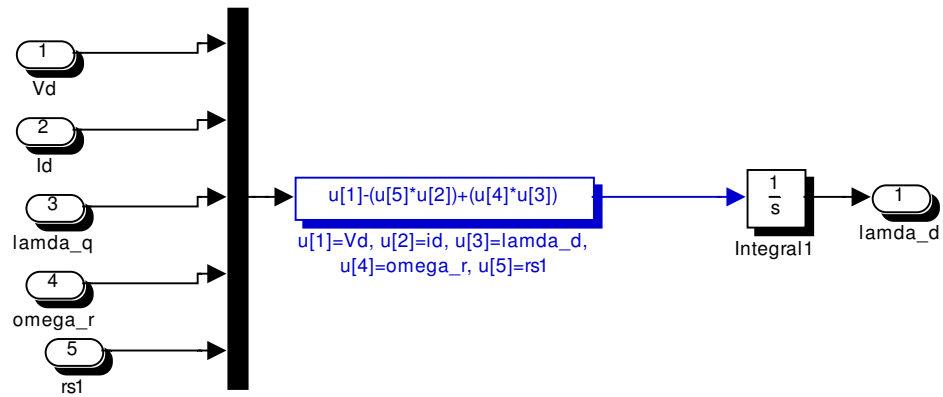
#### 4.1.1.5 Makina için d-eksenine ait stator halkalama akısı ve akım denklemleri

S.M.S.M.'nin d-eksenine ait stator halkalama akısını ve yine bu eksene ait stator akımını bulmak için aşağıdaki denklemler kullanılabilir;

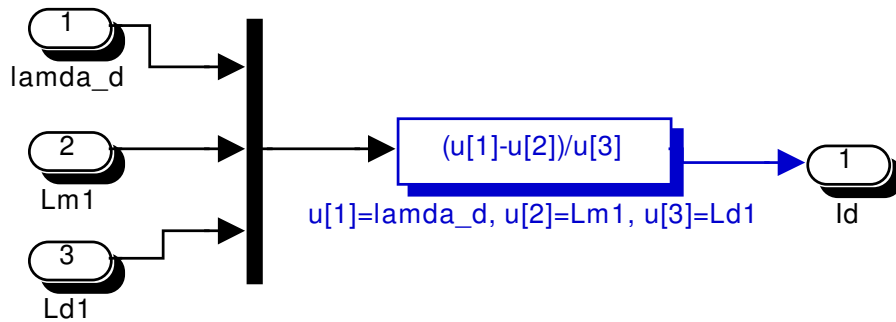
$$\lambda_d = \int (v_d - r_s i_d + \omega_r \lambda_q) dt \quad (4.7)$$

$$i_d = \frac{\lambda_d - \lambda_m}{L_d} \quad (4.8)$$

Denklemler yardımıyla Matlab içinde bloklar oluşturularak bağlantılar sağlanmaktadır.



(a)



(b)

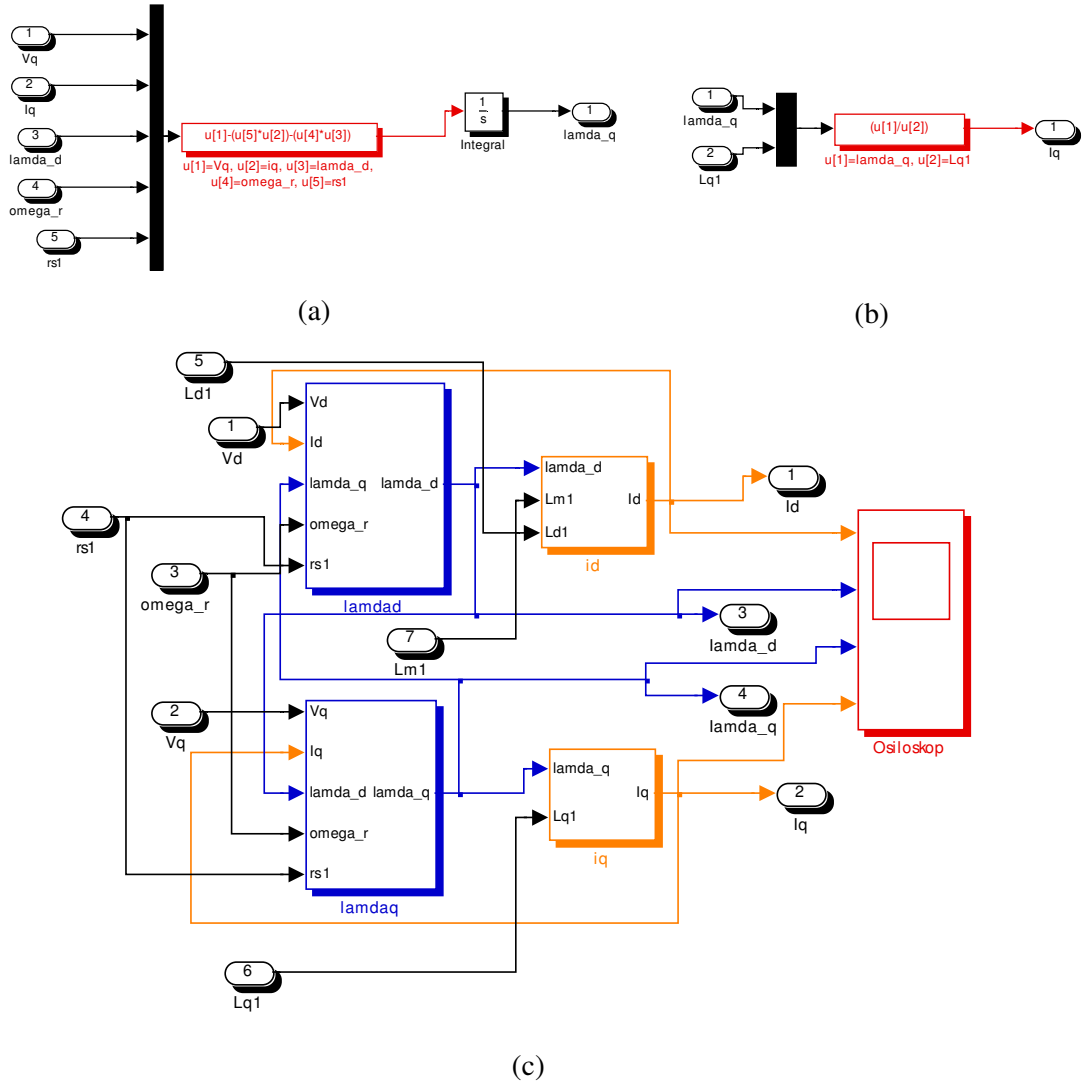
Şekil 4.6: (a)  $\lambda_d$  için blok, (b)  $i_d$  için blok

#### 4.1.1.6 Makina için q-eksenine ait stator halkalama akısı ve akım denklemleri

Denklem (4.9) ve Denklem (4.10) sayesinde makinanın matematiksel modellenmesi için gerekli olan halkalama akısı ve akım değişkenleri hesaplanabilir. Bu çalışmada S.M.S.M.'nin hız kontrolünde  $i_q$  akımı referans alınmaktadır;

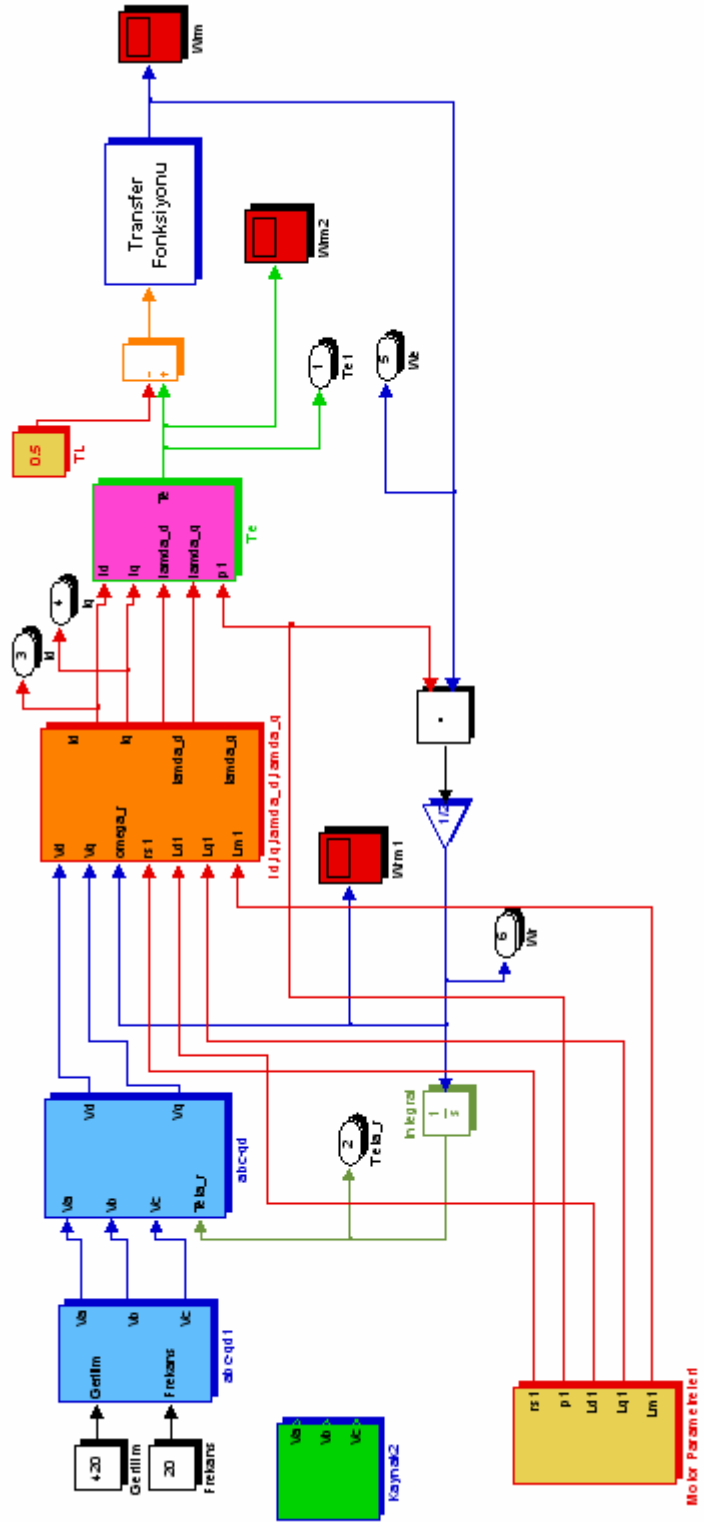
$$\lambda_q = \int (v_q - r_s i_q - \omega_r \lambda_d) dt \quad (4.9)$$

$$i_q = \frac{\lambda_q}{L_q} \quad (4.10)$$

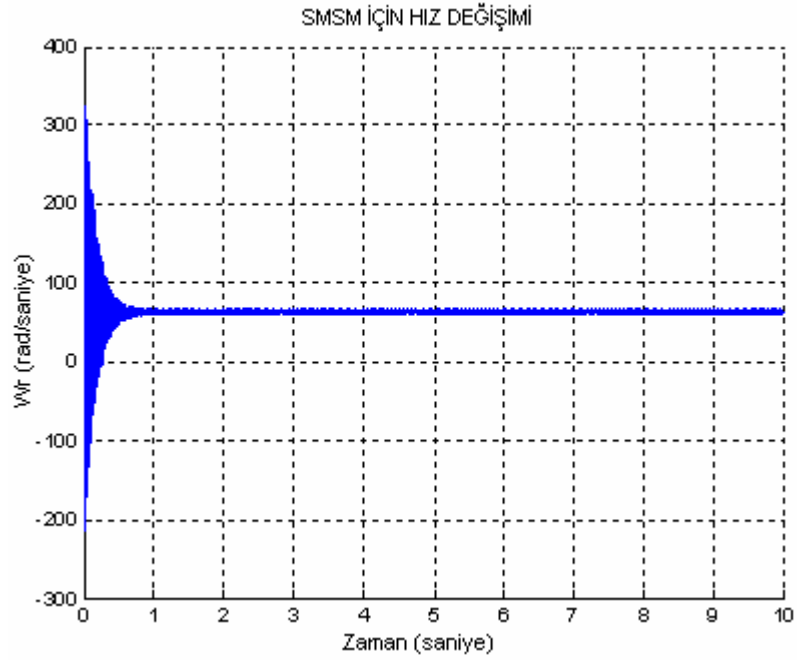


Şekil 4.7: (a)  $\lambda_q$  için blok, (b)  $i_q$  için blok, (c) d ve q-eksenlerine ait bloklar

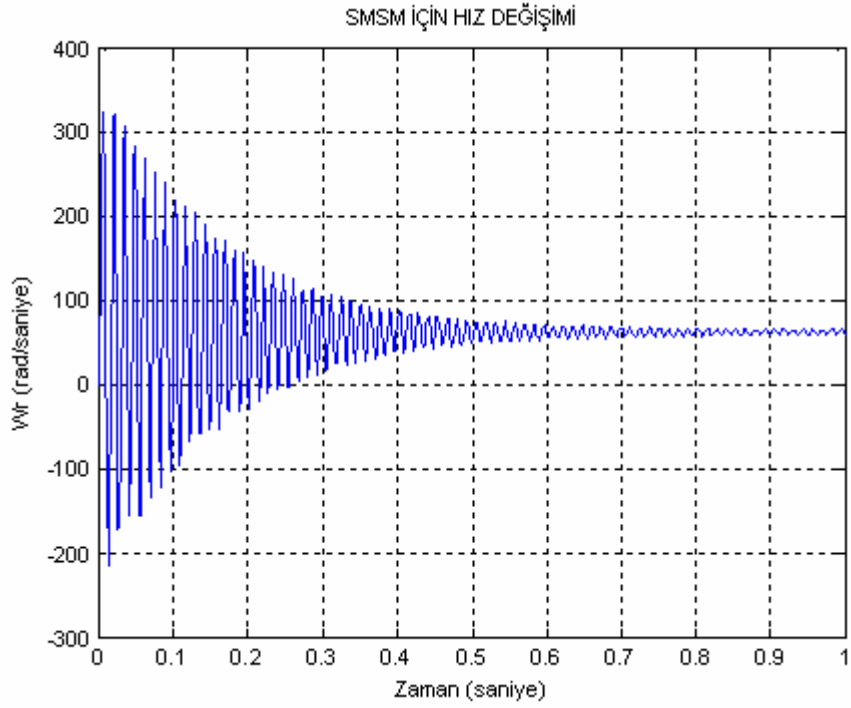
#### 4.1.2 Sürekli mıknatıslı senkron motorun simülasyon modeli



Şekil 4.8: S.M.S.M.'nin modeli



(a)



(b)

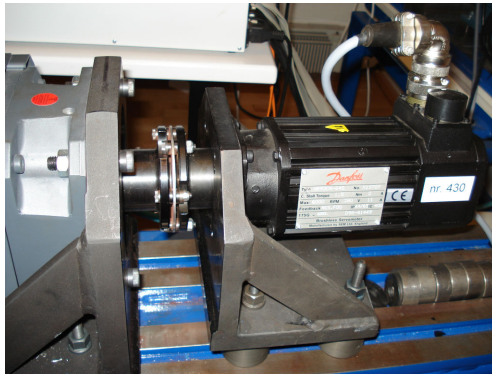
Şekil 4.9: (a) İlk 10 saniye için S.M.S.M.'nin hız değişimi, (b) İlk 1 saniye için S.M.S.M.'nin hız değişimi



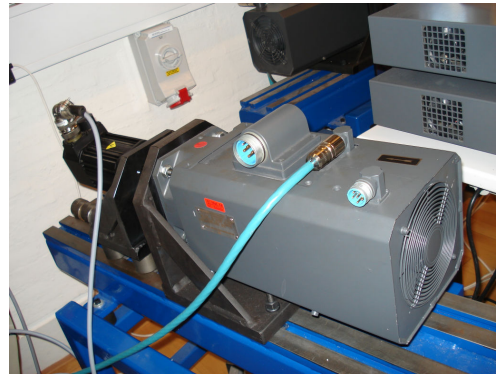


Şekil 4.10: Deneysel çalışmada kullanılan inverter ve PWM parçalarının gösterimi

Aalborg Üniversitesi'nde yapılan deneysel çalışmada, Sürekli Mıknatıslı Senkron Motor'a ait Bulanık mantık ile hız kontrolü için Şekil 4.10'da gösterilen inverter ve PWM devreleri ve Şekil 4.11'da görülen motor kullanılmaktadır. Deneysel çalışmaya başlamadan önce, bu devrelerin simülasyonda çıkartılmış olan sisteme uygun şekilde bağlantıları gerçekleştirilmektedir. Daha sonra simülasyon olarak elde edilen sonuçlar pratik sonuçlarla karşılaştırılmaktadır.

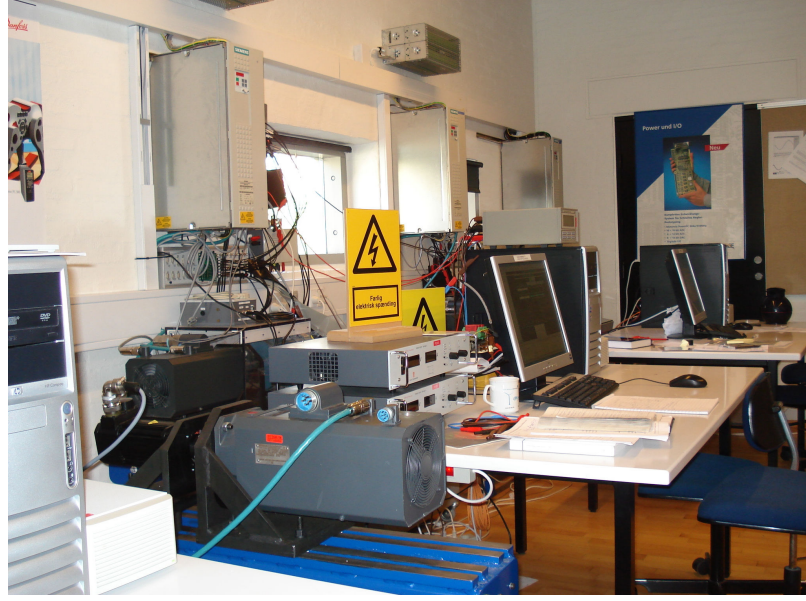


(a)



(b)

Şekil 4.11: (a) Deneysel çalışmada kullanılan S.M.S.M., (b) motora bağlı yükün gösterimi



(a)



(b)

Şekil 4.12: (a) Deneysel çalışmada kullanılan deney düzeneğinin gösterimi, (b) makina laboratuvarının gösterimi

Deneysel çalışma için Aalborg Üniversitesi içerisinde yer alan “Green Laboratory” diye bilinen Güç Elektroniği ve Makina Laboratuvarı’nda çalışılmıştır. Pratik çalışmada ihtiyaç duyulan tüm malzemeler bu oda içerisinde yer almaktadır. Ayrıca Şekil 4.12’de, uygulama için oluşturulan çalışma düzeneği görülebilmektedir.

## 4.2 PID ve Bulanık Denetleyiciler İçin Simülasyon Sonuçları

Yapılan bu çalışma sayesinde tasarlanmış olan Sürekli Mıknatıslı Senkron Motor'un matematiksel modeli kullanılarak, iki farklı denetim yöntemiyle hız kontrolü yapılmıştır. Bu bölümde, makinaya ait matematiksel model yardımıyla önce klasik denetim yöntemleri içinde yer alan PID ve daha sonra da modern denetim yöntemleri içinde yer alan Bulanık Denetleyici yardımıyla bilgisayar ortamında belirlenen referans hız için hız değişimleri incelenmiştir.

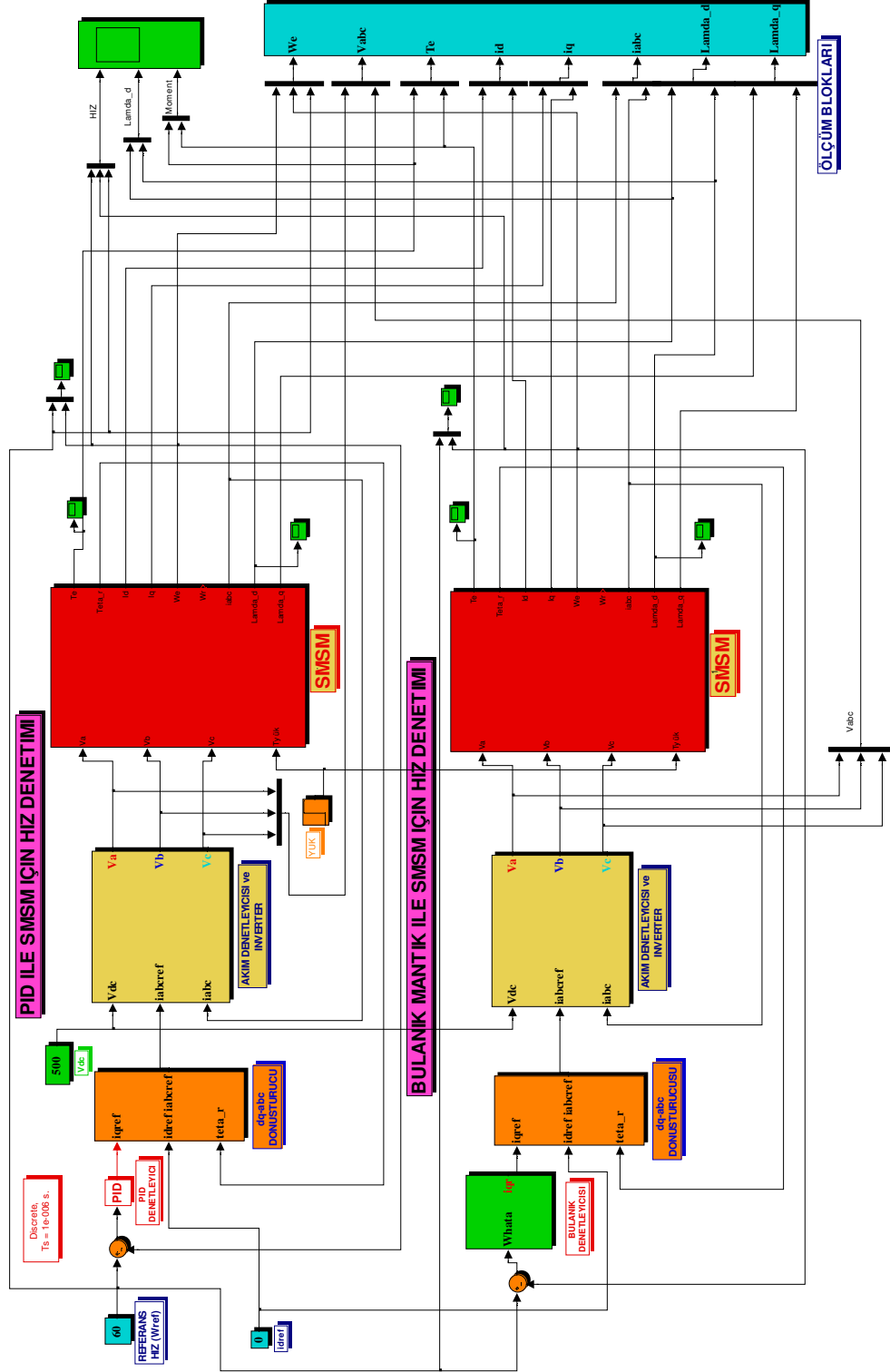
Referans hız değeri olarak ilk başta 60 [rad/sn] seçilmiş daha sonra da farklı referans hız değerleri için simülasyonlar tekrar denenmiştir. Referans hız değeri değişse bile çıkış için elde edilen sonuçlar birbirine yakın olmaktadır. Tek bir fark, referans hız değeri çok büyük seçildiğinde yükselme zamanının da buna bağlı olarak artmasıdır. Bu durum beklenen bir özelliktir. Ancak yine de gerçek zaman ile karşılaştırıldığında bu sürenin 1 saniyeden daha kısa süreli olması yapılan simülasyon denemelerinin doğru olduğunu göstermektedir. Elde edilen benzetim sonuçlarından hız değişimi incelendiği gibi buna ilave olarak da makinaya ait akı, akım, moment ve gerilim gibi değişkenlerinin de değerleri gözlemlenmektedir. Bu şekilde iki farklı denetim için sonuçlar karşılaştırılarak hangi yöntemin daha üstün olduğu genel anlamda görülebilmektedir. Simülasyon ve deneysel çalışmada S.M.S.M.'nin hızı, q-ekseni stator akımı  $i_q$  ile denetlenebilir.

Bu çalışmada, moment denetimi yapılmadığı için d-ekseni stator akımı id sıfır alınmaktadır. Ayrıca S.M.S.M.'nin mekanik açısal hızı ile referans hız karşılaştırılarak elde edilen hata ve hata değişimi bulanık mantık hız denetleyicisine uygulanmaktadır. Bulanık mantık hız denetleyici ile gerekli  $i_q^*$  akımı elde edildikten sonra dq-abc dönüşüm bloğu ile abc eksenlerine geçiş sağlanmaktadır.

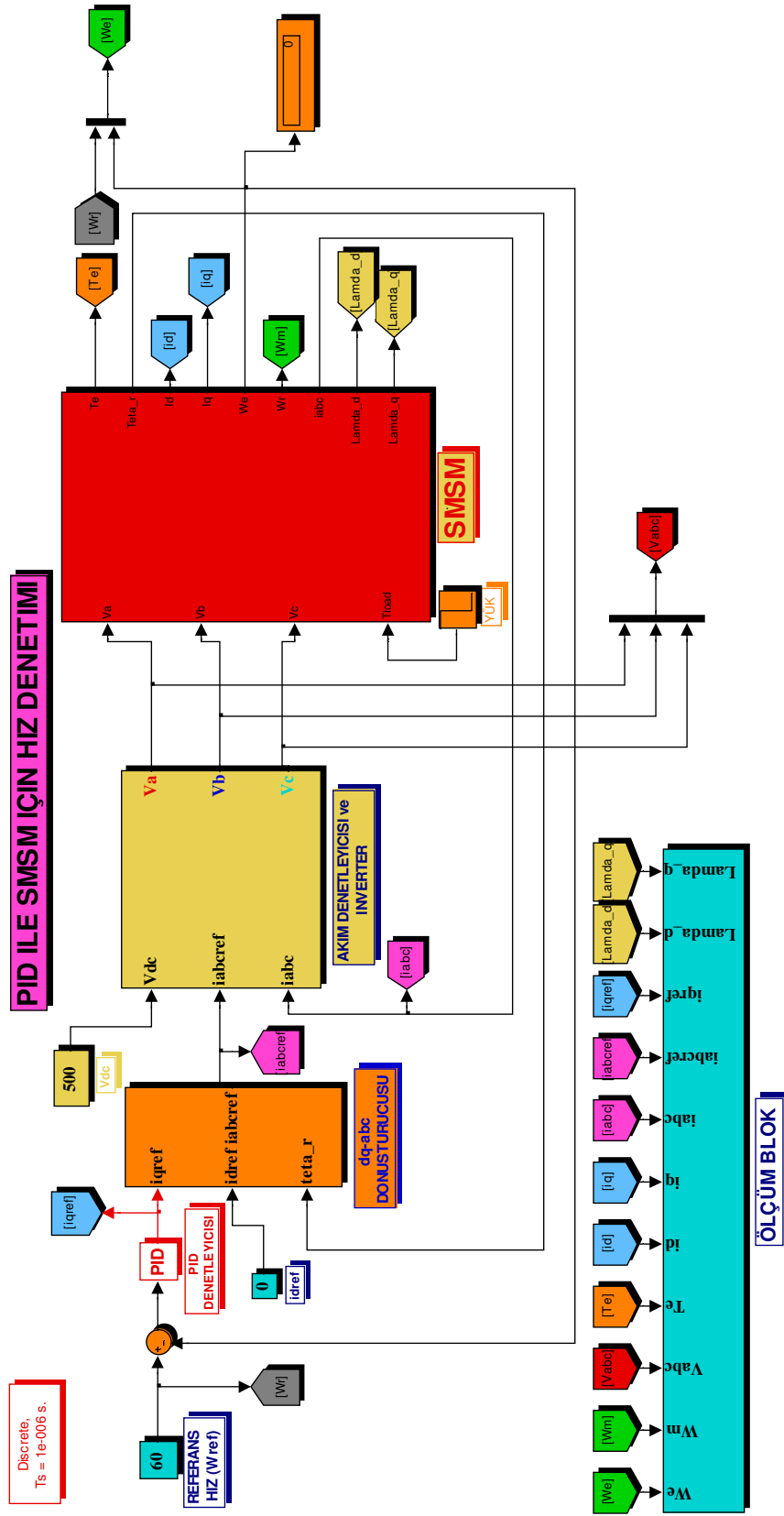
### 4.2.1 PID ve bulanık denetleyiciler ile S.M.S.M.'nin hız denetimi

Bu kısımda ise, oluşturulan S.M.S.M.'nin matematik modeli kullanılarak PID ve Bulanık denetleyiciler ile motorun hız kontrolü anlatılmaktadır. Oluşturulan sistemde

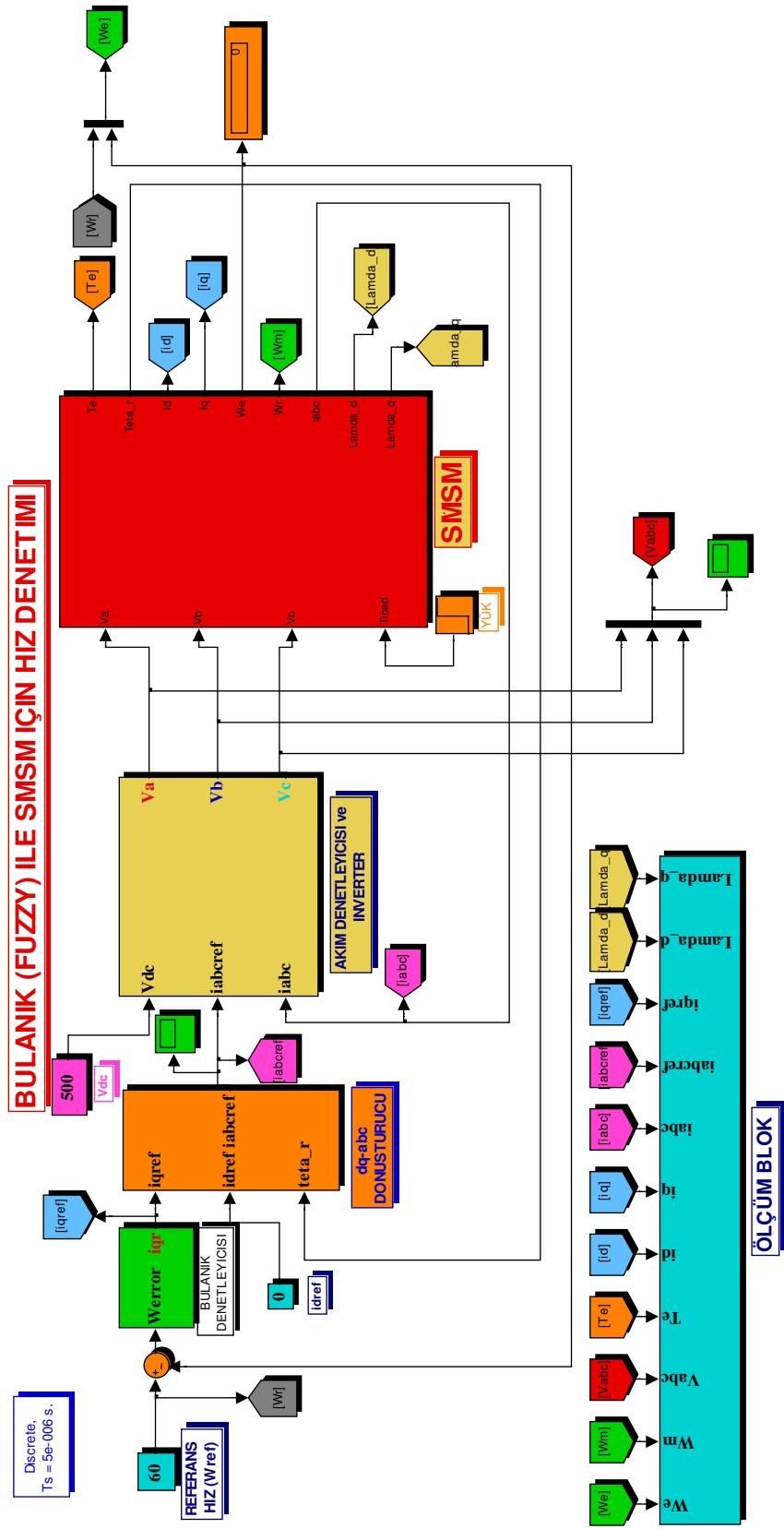
PID ve Bulanık denetleyiciler aynı zamanda çalıştırılarak denetleyicilerinin karşılaştırılması sağlanmıştır.



Şekil 4.13: PID ve Bulanık denetleyicilerle hız kontrolü

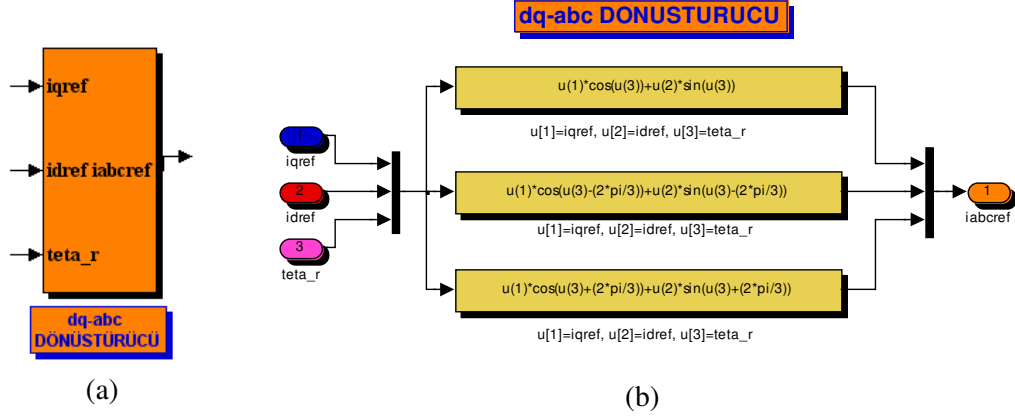


Şekil 4.14: PID denetleyicisine ait kontrol blokları



Şekil 4.15: Bulanık denetleyicisine ait kontrol blokları

#### 4.2.2 Park dönüşüm denklemleri kullanılarak dq-eksenden abc-eksenine ters dönüşüm



Şekil 4.16: (a), (b) Referans akımlar için dq-abc eksen dönüşüm blokları

$$\begin{bmatrix} ia_{ref}^* \\ ib_{ref}^* \\ ic_{ref}^* \end{bmatrix} = \begin{bmatrix} \cos(\theta) & \sin(\theta) & 1 \\ \cos(\theta - \frac{2\pi}{3}) & \sin(\theta - \frac{2\pi}{3}) & 1 \\ \cos(\theta + \frac{2\pi}{3}) & \sin(\theta + \frac{2\pi}{3}) & 1 \end{bmatrix} \cdot \begin{bmatrix} iq_{ref}^* \\ id_{ref}^* \\ io_{ref}^* \end{bmatrix} \quad (4.11)$$

Bilindiği üzere motorun düşük hızlı uygulamaları için d-ekseni referans akımı, sıfır seçilmektedir. Bunun nedeni rotorda d-ekseni boyunca manyetik akı etkisinin de olması ve dolayısıyla oluşan d-ekseni akımı çok küçük değer almasıdır. Düşük hız uygulamalarında da bu değer önemsenmeyecek kadar küçük olduğundan sıfır olarak seçilebilmektedir. Bunun işlemsel gösterimi aşağıdaki gibidir:

$$\lambda_d = \lambda_m \quad (4.12)$$

$$T_e = \frac{3p\lambda_m i_q}{2} \quad (4.13)$$

Kontrol sırasında ilk olarak referans moment dikkate alınır daha sonra bu momente bağlı q-ekseni akımı yapılan PID kontrol bloğu ile ayarlanabilmektedir. Burada halkalama akısının da içinde yer aldığı kısım denklemde sabit şeklinde gösterilip

momentle ilişkisi Denklem (4.14) ve Denklem (4.15)'deki gibi ifade edilebilmektedir:

$$T_e = K_t i_q \quad (4.14)$$

$$K_t = \frac{3p\lambda_m}{2} \quad (4.15)$$

### 4.2.3 PWM akım kontrol modeli

Bu modülasyon türü üç farklı şekilde elde edilebilir. PWM işareti yaklaşık olarak, sürekli dalga modülasyonunun bir türü olan açılı modülasyonuna benzemektedir. Bu nedenle, zaman ve frekans domenindeki ifadelerin kesinlikle analitik olarak ifade etmek mümkün değildir. Ancak, büyük bir yaklaşıklıkla, Fourier serisi açılımından yararlanarak bazı sonuçlar elde edilebilir.

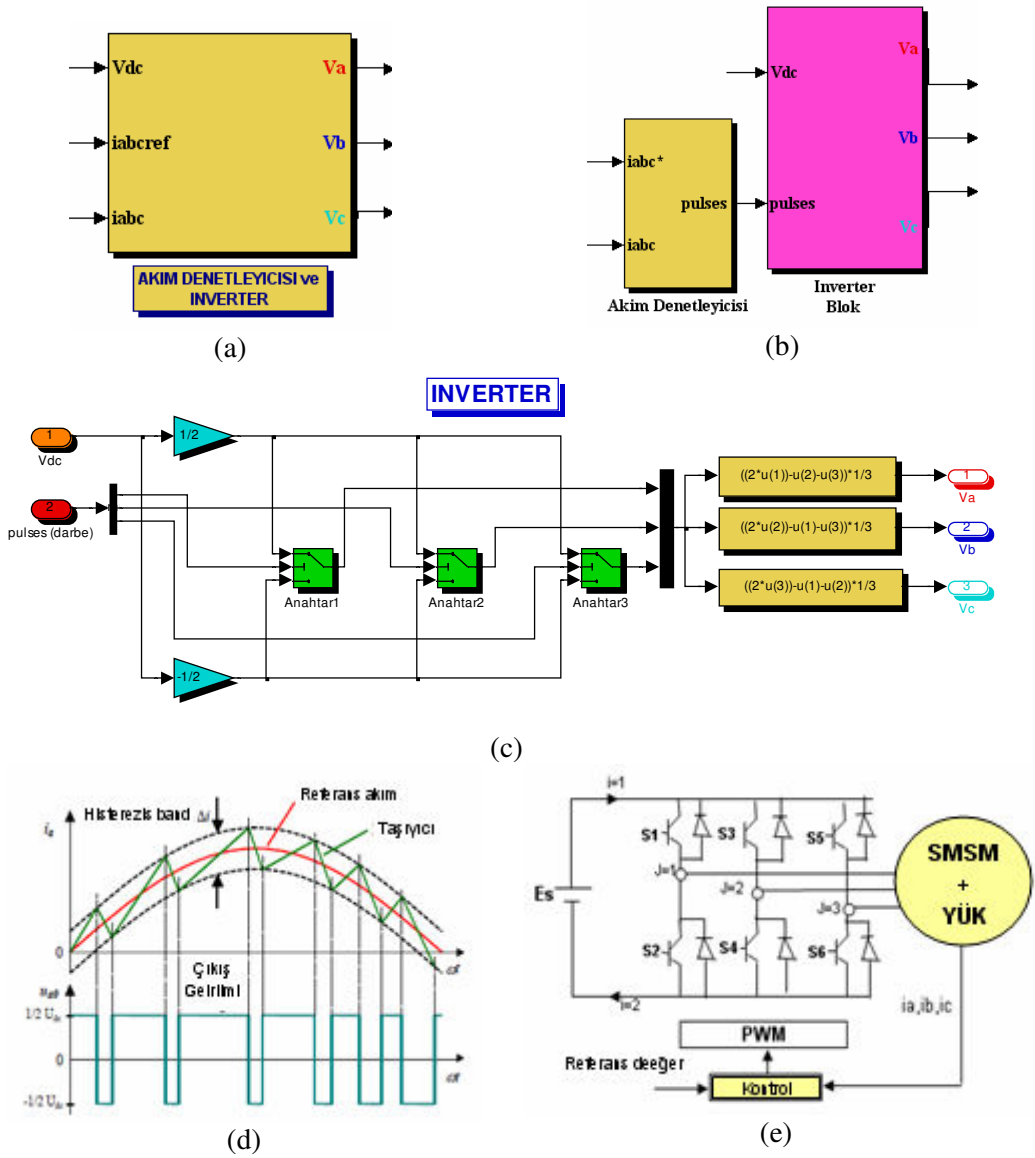
Bu yaklaşıklıkların dayandığı iki gözlem sonucu şunlardır;

1. Mesaj işareti  $x(t)$ 'nin komşu örnek değerleri arasında büyük değer farkları yoktur. Yani  $x(t)$ 'nin frekans bileşenleri genellikle, band genişliğinin çok altında yoğunlaşmıştır.
2. Pratikte, modülasyonlu darbeler için izin verilen maksimum darbe genişliği darbeler arasındaki süreden çok küçüktür.

Bu kısımda amaç, belirlenen band genişliği sınırları altından PWM inverter içindeki anahtarların açılıp kapanmasını sağlamaktır. Referans abc akımları ile makinanın abc akımları karşılaştırılarak buna bağlı sinyal üretilmesi sağlanır. Eğer darbe sinyali pozitif ise çıkış sinyali artar. Band genişliğinin üst seviyesine gelince sinyal durur, buna benzer şekilde darbe sinyali negatif olduğunda çıkış sinyalimiz azalmaktadır.

Bu şekilde anahtar kullanıyormuş gibi band genişliği sınırları altında çıkış sinyali oluşur. Burada Matlab içinde yer alan A.A. gerilim kaynağı üreticinden yararlanılmaktadır.





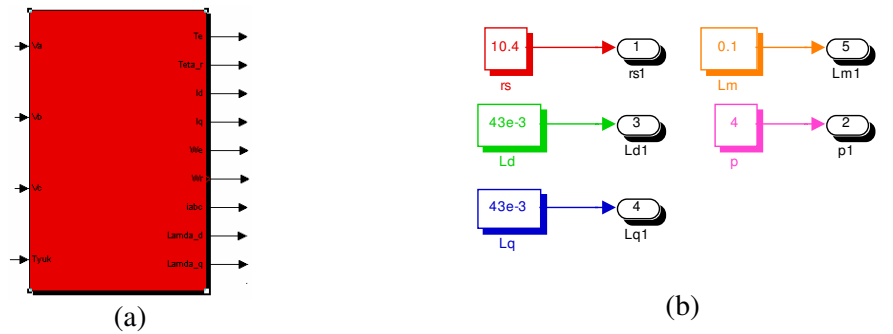
Şekil 4.17: (a), (b), (c) Matlab’da yer alan akım kontrol ve inverter blokları, (d) histeresiz PWM, akım kontrolü ve anahtarlama mantığı, (e) PWM inverter

Anahtarların lojikel değişimi aşağıda görüldüğü gibidir;

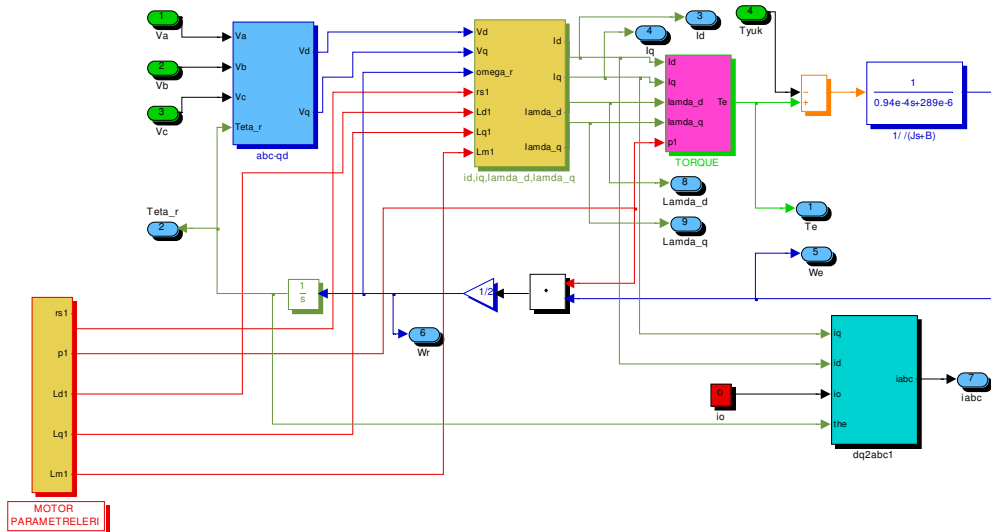
- Eğer  $i_a < (i_a^* - hb)$  anahtar 1 AÇIK ve anahtar 4 KAPALI
- Eğer  $i_a > (i_a^* + hb)$  anahtar 1 KAPALI ve anahtar 4 AÇIK
- Eğer  $i_b < (i_b^* - hb)$  anahtar 3 AÇIK ve anahtar 6 KAPALI
- Eğer  $i_b > (i_b^* + hb)$  anahtar 3 KAPALI ve anahtar 6 AÇIK
- Eğer  $i_c < (i_c^* - hb)$  anahtar 5 AÇIK ve anahtar 2 KAPALI
- Eğer  $i_c > (i_c^* + hb)$  anahtar 5 KAPALI ve anahtar 2 AÇIK

PWM tekniđi içinde yer alan akım kontrolünde hizteresis mantıđın çalıřma tarzı Őekil 4.17(d) içinde gsterilmektedir. Burada belirlenen sınırlar içinde sinyalin ¼retilmesi sađlanır. Pozitif ynde sınırlama yapıldıđı gibi buna benzer olarak negatif ynde de bu Őekilde bir sınırlama yapılmaktadır. Őekil 4.17(d)'de gr¼len  $\Delta i$  band geniřliđini ve sinyalde oluřabilecek hatayı gstermektedir. Kullanılan malzemenin anahtarlanma s¼relerine uygun olarak bu aralık kullanıcı tarafından deđiřtirilebilir. Histeresis band akım denetleyici ile referans akımlar ve motor stator akımları karřılařtırılarak evirici anahtarlanma sinyalleri elde edilmektedir. Evirici, anahtarlanma elemanlarının anahtarlanması ile motora istenilen ayarlı A.A. gerilim uygulanmaktadır.

#### 4.2.4 Motor parametreleri



**MATLAB/SIMULINK İLE SUREKLİ MIKNATİSLİ SENKRON MOTOR (SMSM) İÇİN MATEMATİK MODEL**



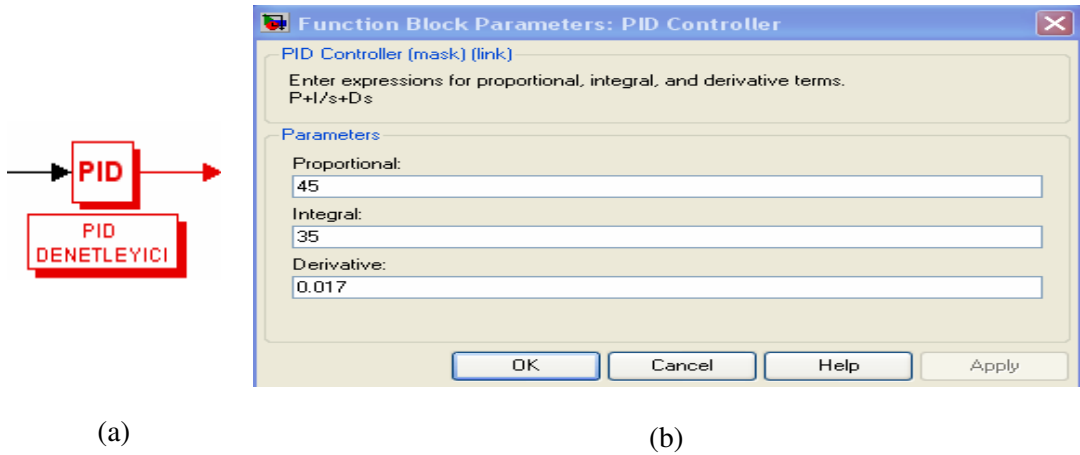
(c)

Őekil 4.18: (a), (b), (c) S.M.S.M. ve motor parametrelerine ait bloklar

Genel olarak Sürekli Mıknatıslı Senkron Motor'un matematiksel modeli Şekil 4.18'de görüldüğü gibidir. Daha önce de belirtildiği gibi simülasyonda gerçek makina parametreleri kullanılmıştır. Simülasyon için Matlab programından yararlanılmıştır.

#### 4.2.5 PID blok seçimi

Bu çalışmada referans hıza ulaşabilmek için önce oransal (P) ve integral (I) katsayıları kullanılmaktadır. Fakat belirlenen referans hıza daha çabuk ulaşabilmek için ve en iyi klasik kontrolü yapabilmek için bu katsayılara ek olarak türevsel (D) katsayısı da kullanılmaktadır. Şekil 4.19'dan da görülebileceği üzere PID blok içerisinde P katsayısı 45 seçilirken, I katsayısı 35 ve D katsayısı 0.017 olarak seçilmektedir.



Şekil 4.19: (a), (b) Kontrol için PID blok gösterimi

Eğer P katsayısı 45 yerine daha küçük bir değer seçilirse hız cevabı referans hıza daha çabuk ulaşır, ancak bununla birlikte de ani çıkış aşma miktarını da arttırabilmektedir. Ayrıca gerçek sistemlerde makinanın bu şekildeki ani değişimleri pek istenmemektedir. Bu durumda, daha yüksek akımlar çekebilecek ve oluşan darbe akımları makinaya zarar verebilmektedir. Bunun tam tersi olarak bu katsayı daha büyük seçilirse de sistemin cevap hızında düşüş olmaktadır. Uygulayıcı tarafından bu koşullar göz önünde tutulup en uygun katsayılar kullanılan sisteme göre seçilmelidir.

## 4.2.6 Bulanık blok

Bulanık denetleyicide amaç, en küçük hata ile sonuca ulaşmaktır. Bu nedenle bu tür kontrolde giriş değişkenleri için hata ve hatadaki değişim miktarları göz önüne alınmaktadır. Kullanılan bu giriş değişkenleri ile bir kural tablosu oluşturularak çıkışın nasıl olması gerektiğine karar verilir.

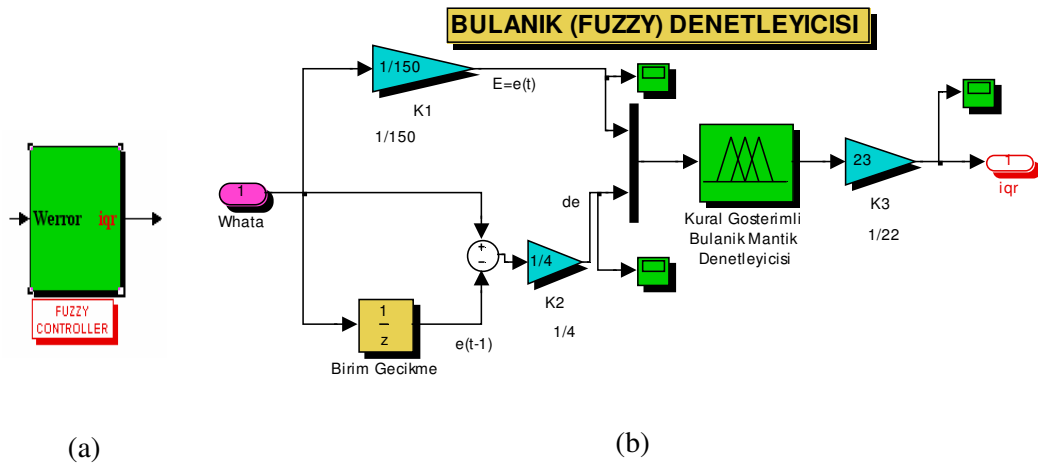
### 4.2.6.1 Bulanık denetleyicinin tasarımı

Aşağıdaki denklemlerden de bu anlatılan ifadeler görülebilir. Burada dikkat edilmesi gereken en önemli nokta  $K_1$ ,  $K_2$  ve  $K_3$  ile gösterilen katsayıların ne amaçla kullanıldığını bilmektir. Bu katsayılar kullanılarak giriş ve çıkışlar için üyelik fonksiyonların sınırları değiştirilebilmektedir. Normalde sadece üyelik fonksiyonların genişlikleri değiştirilerek bu yapılabilir. Ancak bu şekilde yapmak simülasyon sırasında mümkün olmadığı için katsayı olarak eklemek uygulama açısından daha kolay olmaktadır.

$$e(t) = [\omega_{ref}(t) - \omega(t)].K_1 \quad (4.16)$$

$$de(t) = [e(t) - e(t-1)].K_2 \quad (4.17)$$

$$du(t) = [u(t) - u(t-1)].K_3 \quad (4.18)$$



Şekil 4.20: Kontrol için bulanık blok

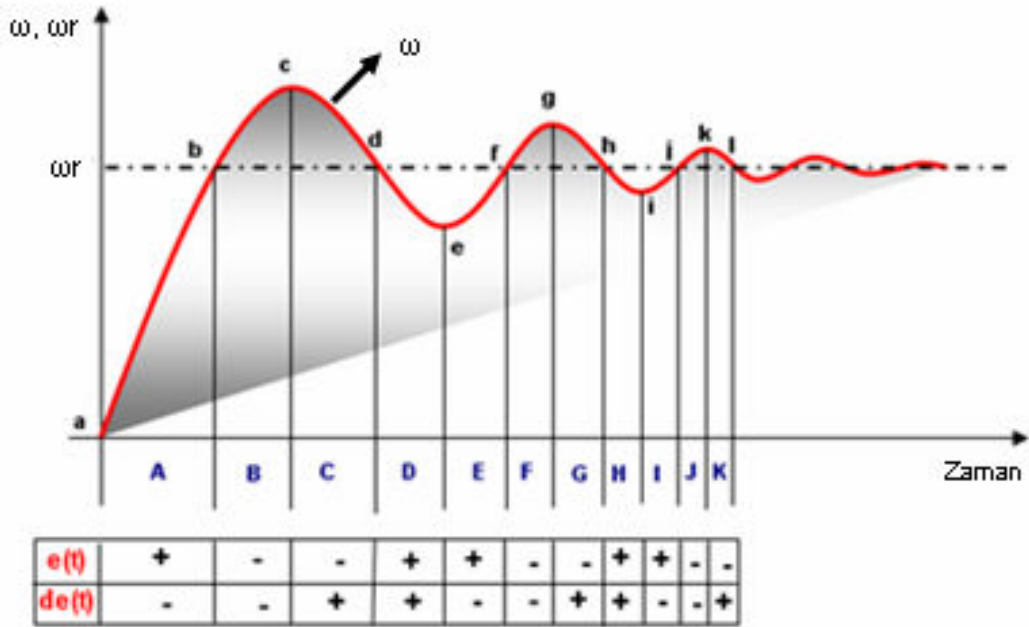
Yapılan bu çalışmada hata, hatadaki değişimler ve çıkış katsayıları aşağıdaki gibi seçilmektedir. Referans hızı için 60 seçildiği için sistem için yeterli ve gerekli katsayılar aşağıdaki gibi seçilmektedir.

$\omega_w$  : -150 ve +150 rad/s

$d\omega_w$ : -4 ve +4 rad/s/s

$u$  : 23 ve -23

Seçilen sınır değerler için simülasyonda kullanılmak amacıyla katsayılar  $K_{1e}=1/150$ ,  $K_{2de}=1/4$  ve  $K_{3u}=23$  olarak seçilmiştir.



Şekil 4.21: Kurak tablosu için çıkışlar

Bulanık mantık yönteminin çalışmasında hata ( $e(t)$ ) ve hatadaki değişim ( $de(t)$ ) göz önünde tutulmaktadır. Öncelikle her bölgeye ait işaretlerin belirlenmesi gerekmektedir. Yukarıdaki şekildeki tabloyu daha ayrıntılı incelemek gerekirse;

- A bölgesi için;

$$e_A(t) = \omega_r - a = +,$$

$$(\omega_r > a),$$

$$de_A(t) = e_A(t) - e_A(t-1) = -,$$

$$(e_A(t-1) > e_A(t)),$$

- B bölgesi için;

$$e_B(t) = w_r - b = -, \quad (w_r < b),$$

$$de_B(t) = e_B(t) - e_B(t-1) = -, \quad (e_B(t-1) > e_B(t)),$$

- C bölgesi için;

$$e_C(t) = w_r - c = -, \quad (w_r < c),$$

$$de_C(t) = e_C(t) - e_C(t-1) = +, \quad (e_C(t-1) < e_C(t)),$$

- D bölgesi için;

$$e_D(t) = w_r - d = +, \quad (w_r > d),$$

$$de_D(t) = e_D(t) - e_D(t-1) = +, \quad (e_D(t-1) < e_D(t)),$$

- E bölgesi için;

$$e_E(t) = w_r - e = +, \quad (w_r > e),$$

$$de_E(t) = e_E(t) - e_E(t-1) = -, \quad (e_E(t-1) > e_E(t)),$$

- F bölgesi için;

$$e_F(t) = w_r - f = -, \quad (w_r < f),$$

$$de_F(t) = e_F(t) - e_F(t-1) = -, \quad (e_F(t-1) > e_F(t)),$$

- G bölgesi için;

$$e_G(t) = w_r - g = -, \quad (w_r < g),$$

$$de_G(t) = e_G(t) - e_G(t-1) = +, \quad (e_G(t-1) < e_G(t)),$$

- H bölgesi için;

$$e_H(t) = w_r - h = +, \quad (w_r > h),$$

$$de_H(t) = e_H(t) - e_H(t-1) = +, \quad (e_H(t-1) < e_H(t)),$$

- I bölgesi için;

$$e_I(t) = w_r - i = +, \quad (w_r > i),$$

$$de_I(t) = e_I(t) - e_I(t-1) = -, \quad (e_I(t-1) > e_I(t)),$$

- J bölgesi için;

$$e_J(t) = w_r - j = -, \quad (w_r < j),$$

$$de_J(t) = e_J(t) - e_J(t-1) = -, \quad (e_J(t-1) > e_J(t)),$$

- K bölgesi için;

$$e_K(t) = w_r - k = -, \quad (w_r < k),$$

$$de_K(t) = e_K(t) - e_K(t-1) = +, \quad (e_K(t-1) < e_K(t))$$

#### 4.2.6.2 Üyelik fonksiyonları ve kural tablosunun oluşturulması

Oluşturulan kurallar yardımıyla hız değişkeninin önceden belirlenen referans hız değerine ne kadar ve ne şekilde ulaşacağı belirlenmektedir. Aşağıdaki Tablo 4.2 üzerinden örnek verilmek istenilirse; ilk kural için , “Eğer (e(t)=NB) ve (de(t) =NB) İse Çıkış (u(t)=NB olur)” şeklinde bir tanımlama yapılabilmektedir. Burada ilk durumda belirlenen referans hızdan negatif yönde çok uzak olduğu için ani değişime izin verilmesi gerektiği ve üyelik fonksiyonlarından da Negatif Büyük (NB) olarak tanımlanan üyelik fonksiyonunun seçilmesi gerektiği görülmektedir. Referans hıza yaklaşıldıkça hızdaki salınımın başka bir değişle değişimin daha az olması beklenir. Bu şekilde daha hassas bir yaklaşım yapılmış olunmaktadır. Hassasiyetin önemsenmediği durumlarda kural sayısı azaltılabildiği gibi üyelik fonksiyonlar da daha farklı seçilebilmektedir.

Tablo 4.2: Bulanık denetleyici için kural tablosu

$\begin{matrix} de(t) \\ e(t) \end{matrix}$	NB	NO	NK	SIFIR	PK	PO	PB
NB	NB	NB	NB	NO	NS	PO	PS
NO	NB	NB	NB	NB	NO	NO	PB
NK	NB	NB	NO	NS	PS	PS	PO
SIFIR	NB	NO	NS	SIFIR	PS	PO	PB
NK	NO	NS	NS	PS	PO	PO	PB
NO	NS	NO	PS	PO	PB	PB	PB
NB	NB	PS	PO	PB	PB	PB	PB

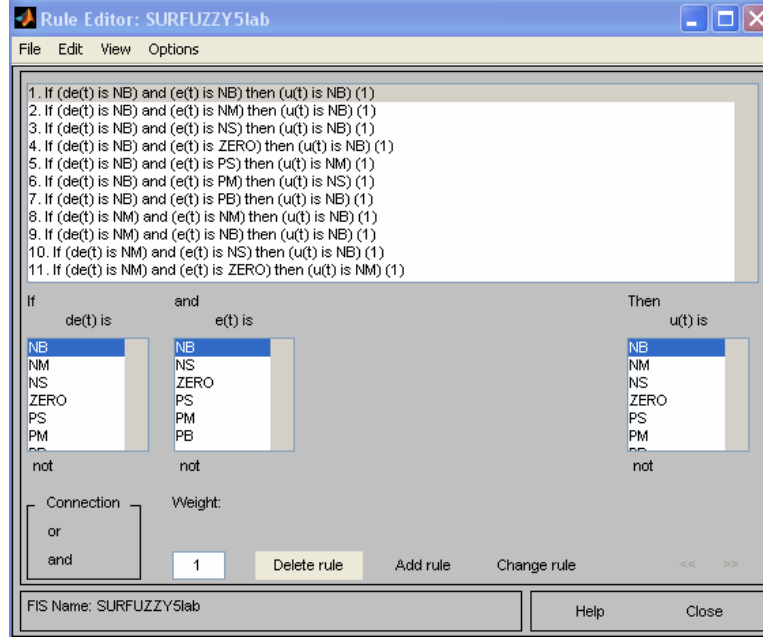
Belirtilen formatta da giriş değişkenleri eksenlere yerleştirilmiş çıkış değişkenleri de tablonun iç kısmındadır. Bu tabloda boş bir hücre olması, bir kuralın atlandığını göstermektedir. Bu format, kuralların tam olduğunu kontrol etmek açısından daha kullanışlı olup “dilsel tabanlı” (linguistic phase plane) olarak adlandırılmaktadır.  $n > 2$  sayıda giriş değişkeni olduğunda, tablo  $n$  boyutlu bir diziye dönüşmektedir. Basitçe, kavramsal bir denetleyici “if-then (eğer-ise)” kalıbındaki kuralları içerir, ama farklı kalıplarda da gösterilebilirler. Bazı sistemlerde, kurallar son kullanıcıya aşağıdakilere benzer şekillerde sunulmaktadır;

1. Eğer ( $e(t) = NB$ ) ve ( $de(t) = NB$ ) İse ( $u(t) = NB$ )
2. Eğer ( $e(t) = NO$ ) ve ( $de(t) = NB$ ) İse ( $u(t) = NB$ )
3. Eğer ( $e(t) = NS$ ) ve ( $de(t) = NB$ ) İse ( $u(t) = NB$ )
4. Eğer ( $e(t) = SIFIR$ ) ve ( $de(t) = NB$ ) İse ( $u(t) = NB$ )
5. Eğer ( $e(t) = PS$ ) ve ( $de(t) = NB$ ) İse ( $u(t) = NO$ )
6. Eğer ( $e(t) = PO$ ) ve ( $de(t) = NB$ ) İse ( $u(t) = NS$ )
7. Eğer ( $e(t) = PB$ ) ve ( $de(t) = NB$ ) İse ( $u(t) = NB$ )
8. Eğer ( $e(t) = NB$ ) ve ( $de(t) = NO$ ) İse ( $u(t) = NB$ )
9. Eğer ( $e(t) = NO$ ) ve ( $de(t) = NO$ ) İse ( $u(t) = NB$ )
10. Eğer ( $e(t) = NS$ ) ve ( $de(t) = NO$ ) İse ( $u(t) = NB$ )
11. Eğer ( $e(t) = SIFIR$ ) ve ( $de(t) = NO$ ) İse ( $u(t) = NO$ )
12. Eğer ( $e(t) = PS$ ) ve ( $de(t) = NO$ ) İse ( $u(t) = NS$ )
13. Eğer ( $e(t) = PO$ ) ve ( $de(t) = NO$ ) İse ( $u(t) = NO$ )
14. Eğer ( $e(t) = PB$ ) ve ( $de(t) = NO$ ) İse ( $u(t) = PS$ )
15. Eğer ( $e(t) = NB$ ) ve ( $de(t) = NS$ ) İse ( $u(t) = NB$ )
16. Eğer ( $e(t) = NO$ ) ve ( $de(t) = NS$ ) İse ( $u(t) = NB$ )
17. Eğer ( $e(t) = NS$ ) ve ( $de(t) = NS$ ) İse ( $u(t) = NO$ )
18. Eğer ( $e(t) = SIFIR$ ) ve ( $de(t) = NS$ ) İse ( $u(t) = NS$ )
19. Eğer ( $e(t) = PS$ ) ve ( $de(t) = NS$ ) İse ( $u(t) = NS$ )
20. Eğer ( $e(t) = PO$ ) ve ( $de(t) = NS$ ) İse ( $u(t) = PS$ )
21. Eğer ( $e(t) = PB$ ) ve ( $de(t) = NS$ ) İse ( $u(t) = PO$ )
22. Eğer ( $e(t) = NB$ ) ve ( $de(t) = SIFIR$ ) İse ( $u(t) = NO$ )
23. Eğer ( $e(t) = NO$ ) ve ( $de(t) = SIFIR$ ) İse ( $u(t) = NB$ )
24. Eğer ( $e(t) = NS$ ) ve ( $de(t) = SIFIR$ ) İse ( $u(t) = NS$ )



25. Eğer (e(t) = SIFIR) ve (de(t) = SIFIR) İse (u(t) = SIFIR)
26. Eğer (e(t) = PS) ve (de(t) = SIFIR) İse (u(t) = PS)
27. Eğer (e(t) = PO) ve (de(t) = SIFIR) İse (u(t) = PO)
28. Eğer (e(t) = PB) ve (de(t) = SIFIR) İse (u(t) = PB)
29. Eğer (e(t) = NB) ve (de(t) = PS) İse (u(t) = NS)
30. Eğer (e(t) = NO) ve (de(t) = PS) İse (u(t) = NO)
31. Eğer (e(t) = NS) ve (de(t) = PS) İse (u(t) = PS)
32. Eğer (e(t) = SIFIR) ve (de(t) = PS) İse (u(t) = PS)
33. Eğer (e(t) = PS) ve (de(t) = PS) İse (u(t) = PO)
34. Eğer (e(t) = PO) ve (de(t) = PS) İse (u(t) = PB)
35. Eğer (e(t) = PB) ve (de(t) = PS) İse (u(t) = PB)
36. Eğer (e(t) = NB) ve (de(t) = PO) İse (u(t) = PO)
37. Eğer (e(t) = NO) ve (de(t) = PO) İse (u(t) = NO)
38. Eğer (e(t) = NS) ve (de(t) = PO) İse (u(t) = PS)
39. Eğer (e(t) = SIFIR) ve (de(t) = PO) İse (u(t) = PO)
40. Eğer (e(t) = PS) ve (de(t) = PO) İse (u(t) = PO)
41. Eğer (e(t) = PO) ve (de(t) = PO) İse (u(t) = PB)
42. Eğer (e(t) = PB) ve (de(t) = PO) İse (u(t) = PB)
43. Eğer (e(t) = NB) ve (de(t) = PB) İse (u(t) = PS)
44. Eğer (e(t) = NO) ve (de(t) = PB) İse (u(t) = PB)
45. Eğer (e(t) = NS) ve (de(t) = PB) İse (u(t) = PO)
46. Eğer (e(t) = SIFIR) ve (de(t) = PB) İse (u(t) = PB)
47. Eğer (e(t) = PS) ve (de(t) = PB) İse (u(t) = PB)
48. Eğer (e(t) = PO) ve (de(t) = PB) İse (u(t) = PB)
49. Eğer (e(t) = PB) ve (de(t) = PB) İse (u(t) = PB)

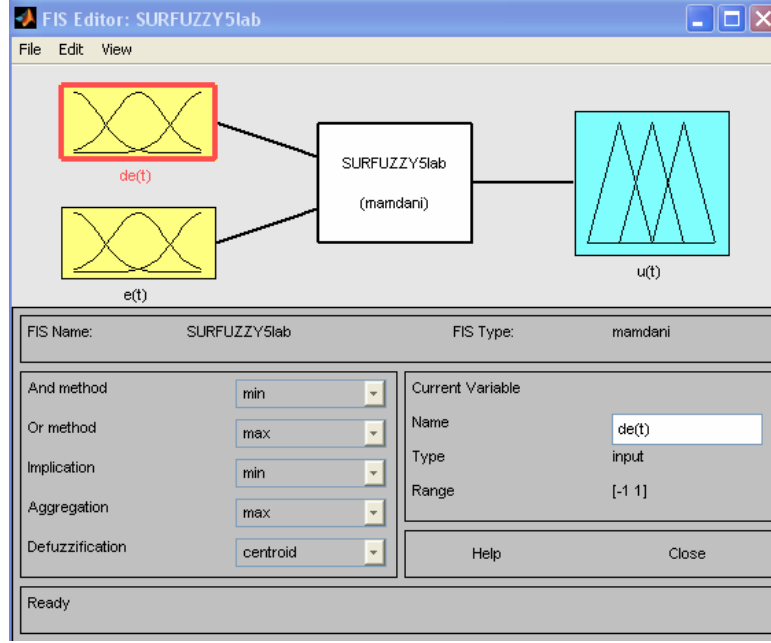
Burada, soldaki iki sütunun girişleri, sağdaki sütunun çıkışları olduğu ve her satırın bir kuralı gösterdiği anlaşılabilir. Bu format hızlı bir şekilde kuralları anlamak isteyen deneyimli bir kullanıcı için daha uygun olabilir. Şekil 4.22’de kural tablosu editörü gösterilmektedir. Oluşturulmuş kurallar bu editor içerisinde yazılarak sistem için gerekli olan kural tablosu elde edilir. Kural tablosunda üyelik fonksiyonlarının seçimi önem kazanmaktadır. Bunun nedeni ise, seçilen üyelik fonksiyonun şekline göre kurallar farklı olabilmesidir.



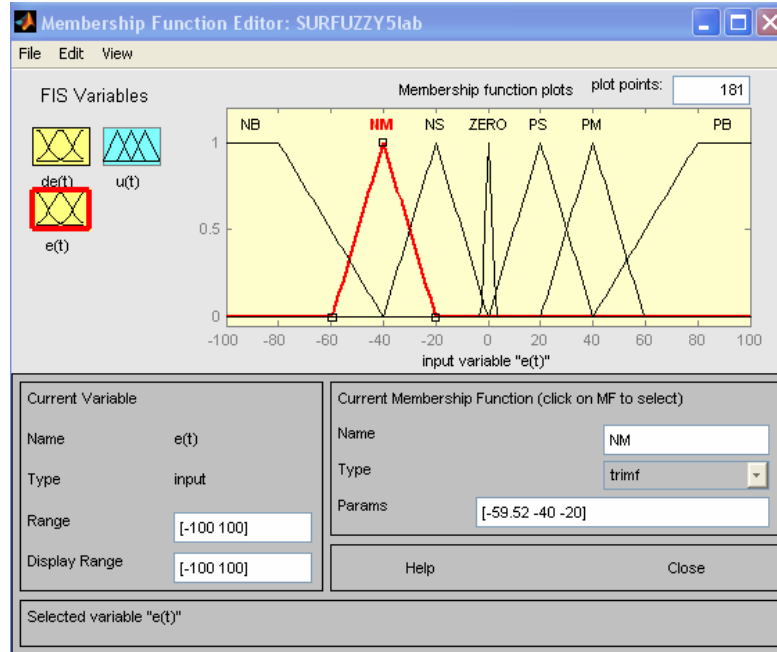
Şekil 4.22: Bulanık control için kural editörü

Matlab içinde yer alan Bulanık Mantık editöründe (Fis Editor) hem giriş hem de çıkış değişkenleri için isimleri ve bulanıklaştırmada uygulanacak yöntem olarak Matlab içinde hazır olarak bulunan Mamdani yöntemi seçilmektedir. Mamdani yöntemi için Tablo 4.1’de de gösterilen çok basit bir matematik denkleminde yararlanılmaktadır. Şekil 4.23’te bu editor görülmektedir. Burada üyelik fonksiyonlarında sayılar yerine dilsel ifadeler kullanılmaktadır. Çeşitli yöntemlerle (max, min, vb) durulaştırma yapılır. Bu şekilde gerçek sayılardan bulanık ifadelere geçilip daha sonra durulaştırma ile de tam tersine dönülmektedir.

Giriş ve çıkış değişkenleri daha hassas bir seviye vermek için negatif büyük (NB), negatif orta (NO), negatif küçük (NK), sıfır (SIFIR, ZERO), pozitif küçük (PK), pozitif orta (PO), pozitif büyük (PB) olmak üzere 7 bulanık kümeye ayrılmaktadır. Matlab Bulanık Mantık Araç Kutusu (Matlab Fuzzy Logic Toolbox) simülasyon programı kullanılarak S.M.S.M. hız denetimi için oluşturulmaktadır. Bulanık mantık denetleyici yapısında Mamdani min-max çıkartım yöntemi ve ağırlık merkezi durulama yöntemi kullanılmaktadır. Hata, hata değişimi ve bulanık mantık çıkışına ait üyelik fonksiyonları aşağıda verilmektedir.

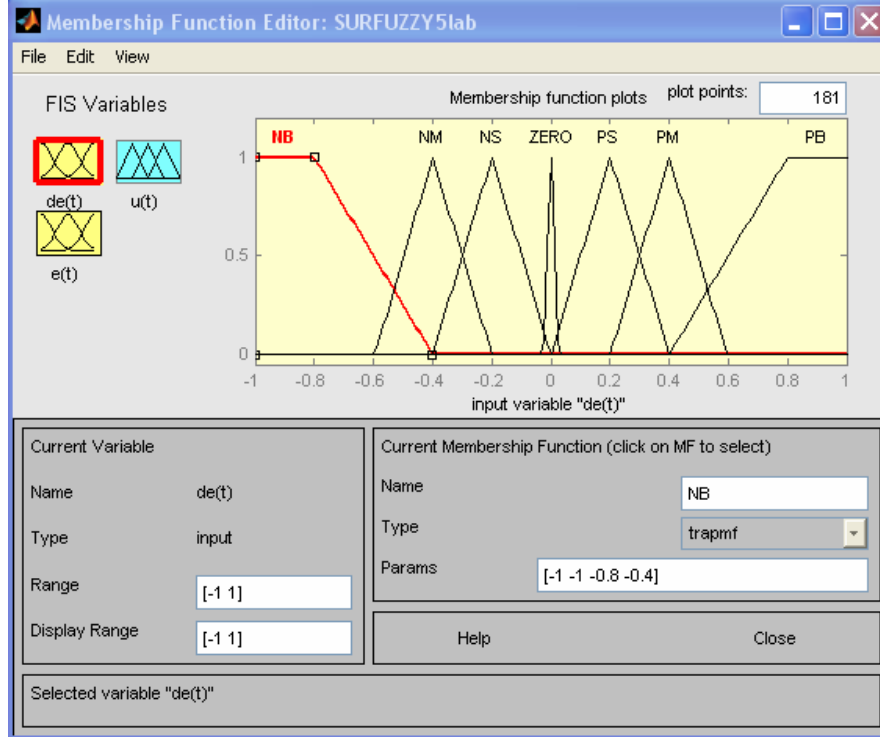


Şekil 4.23: Bulanık için fis editör

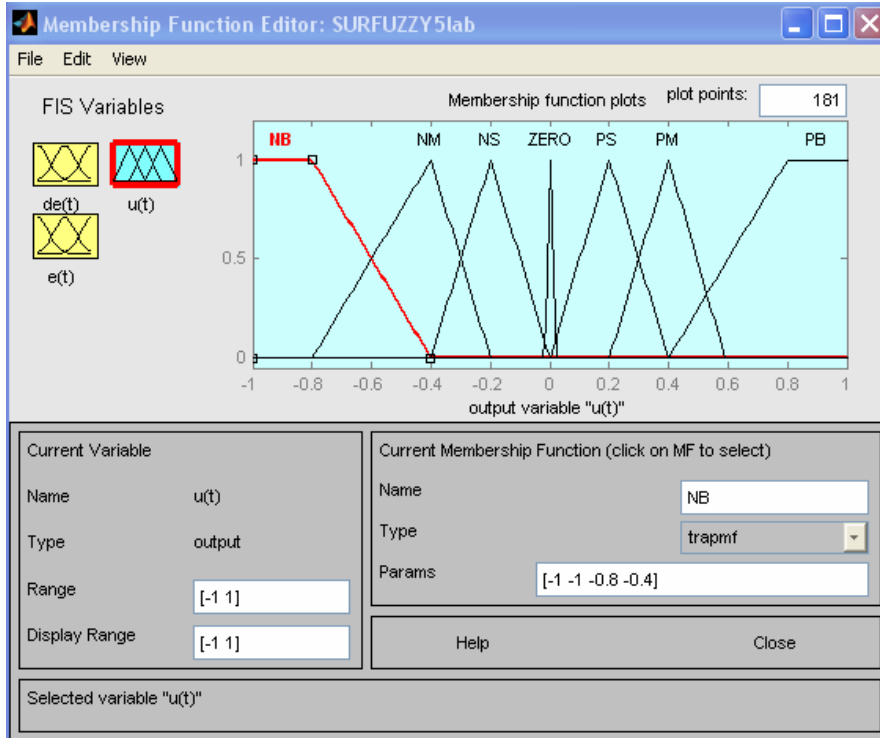


Şekil 4.24: Giriş-1 (e(t)) için üyelik fonksiyon editörü

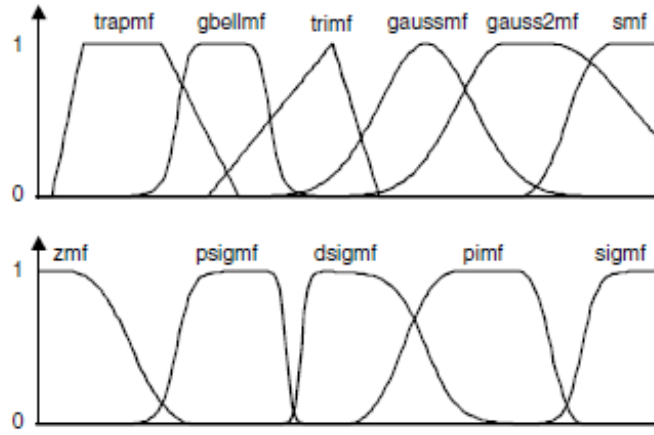
Şekil 4.24, Şekil 4.25 ve Şekil 4.26'da giriş ve çıkış için üyelik fonksiyonlar gösterilmektedir. Burada üyelik fonksiyon olarak üçgen, tek darbe, gauss gibi farklı şekiller de seçilebilir. Bu tamamen kullanıcının isteğine bağlı olarak değişmektedir. Şekil 4.27'de farklı üyelik fonksiyonları yer almaktadır.



Şekil 4.25: Giriş-2 için ( $de(t)$ ) üyelik fonksiyon editörü

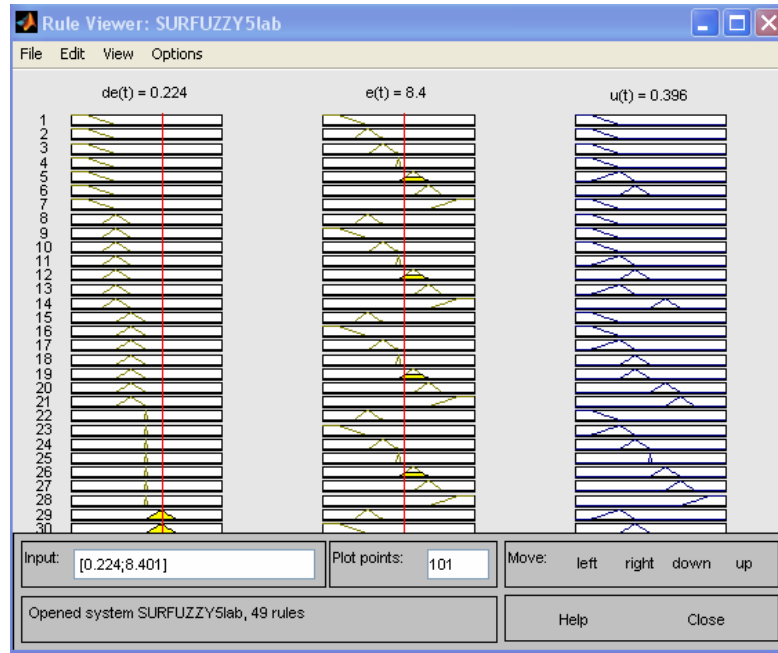


Şekil 4.26: Çıkış ( $u(t)$ ) için üyelik fonksiyon editörü



Şekil 4.27: Bulanık denetleyici araç çubuğunda yer alan üyelik fonksiyon çeşitleri

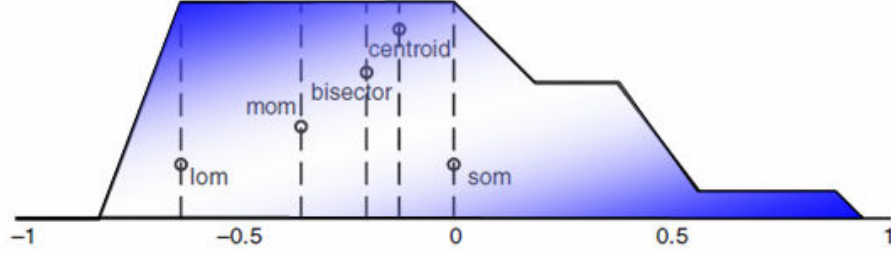
Şekil 4.28’de bulanık denetleyici içinde yer alan kural gösterici editör (rule viewer) verilmektedir. Bu editör sayesinde simülasyonun çalıştığı sırada giriş ve çıkış değişkenler için kuralların ne şekilde çalıştığını görmek mümkün olur.



Şekil 4.28: Bulanık mantık için kuralların çalışmasını gösteren editör

Bu bölümün başında Bulanık mantık denetleyici kısmında, durulaştırma yöntemleri hakkında bilgi verilmişti. Şekil 4.29’da da durulaştırmada kullanılabilecek yöntemler

gösterilmektedir. Kullanıcı sahip olduğu sisteme en uygun yöntemi seçerek kontrol için bu yöntemden yararlanılmaktadır.



Şekil 4.29: Bulanık mantık teorisinde durulaştırma

Tablo 4.3: Bulanık mantık için Matlab içinde kullanılan kurallar

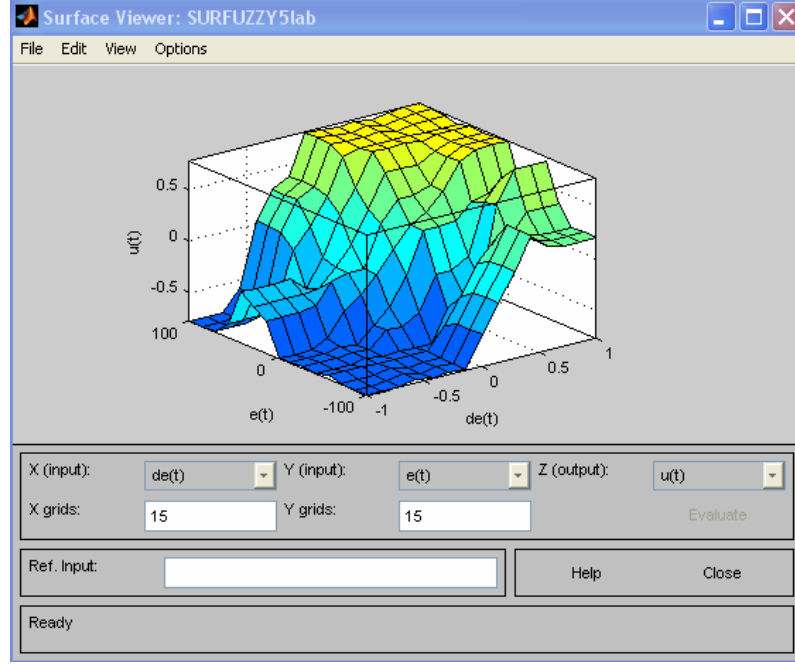
KURALLAR	KURALLARIN AÇIKLAMASI
showfis(w)	Bulanık denetleyici yapısının tamamını göster
plotmf(w, 'input', 1)	Giriş-1 için tüm üyelik fonksiyonlarını göster
getfis(w)	Bulanık sistem seçenekleri
plotfis(w)	Giriş ve çıkış şekillerini çiz
gensurf(w)	Yüzeysel olarak göster
defuzzdm	Durulaştırma yöntemleri
rmmf	Üyelik fonksiyonlarını kaldır
newfis	Yeni bir dosya oluştur
writefis	Dosyayı kaydet
rmvar	Değişkenleri sil

Durulaştırma yöntemlerinin sonuçları çok farklılık göstermemektedir. Sadece matematiksel işlemlerde yaklaşık değerler elde edilip hassasiyetin önemliliğine bağlı olarak kullanılacak yöntem seçilmektedir. Aşağıdaki denklemlerde bu yöntemlerden maksimum, minimum ve ağırlık merkezi yöntemleri için kullanılabilir denklemler gösterilmektedir [2,6].

$$\max(\mu_1) / k \quad (4.20)$$

$$\min(\mu_1) / k \quad (4.21)$$

$$\sum_{j=1}^p y_j \mu(y_j) / \sum_{j=1}^p \mu(y_j) \quad (4.22)$$



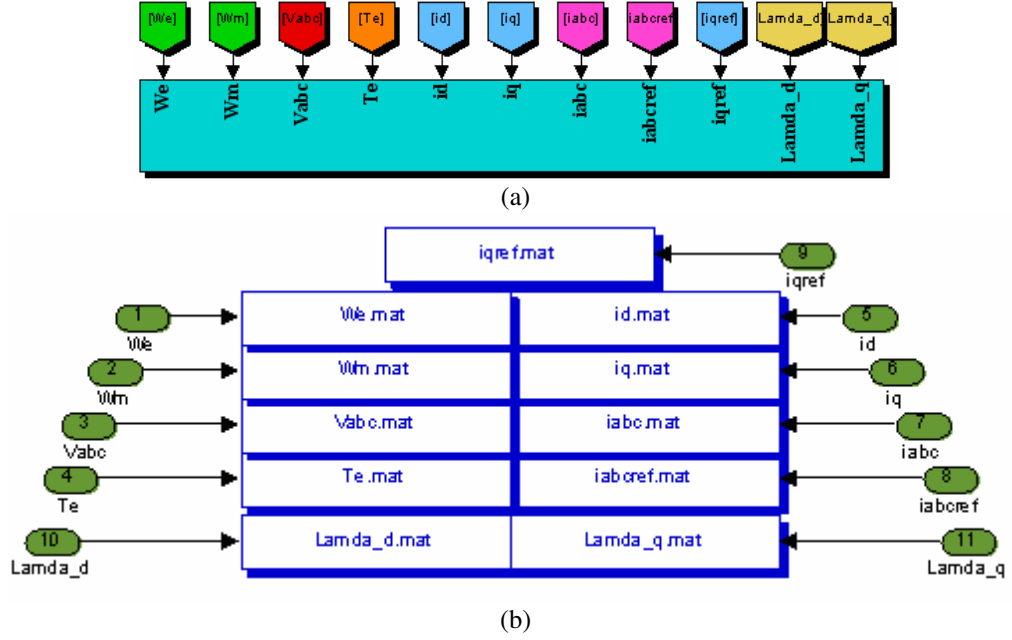
Şekil 4.30: Bulanık denetleyicinin yüzeysel görünümü

Tablo 4.3'te Matlab programı içinde bulanık denetleyici için yazılabilecek kurallar gösterilmektedir. Program içinde simülasyon kısmında araç kutuları kullanılarak denetim yapılabileceği gibi sadece kurallar kullanılarak ve bu şekilde yazılan program dili yardımıyla istenilen her şey yapılabilmektedir.

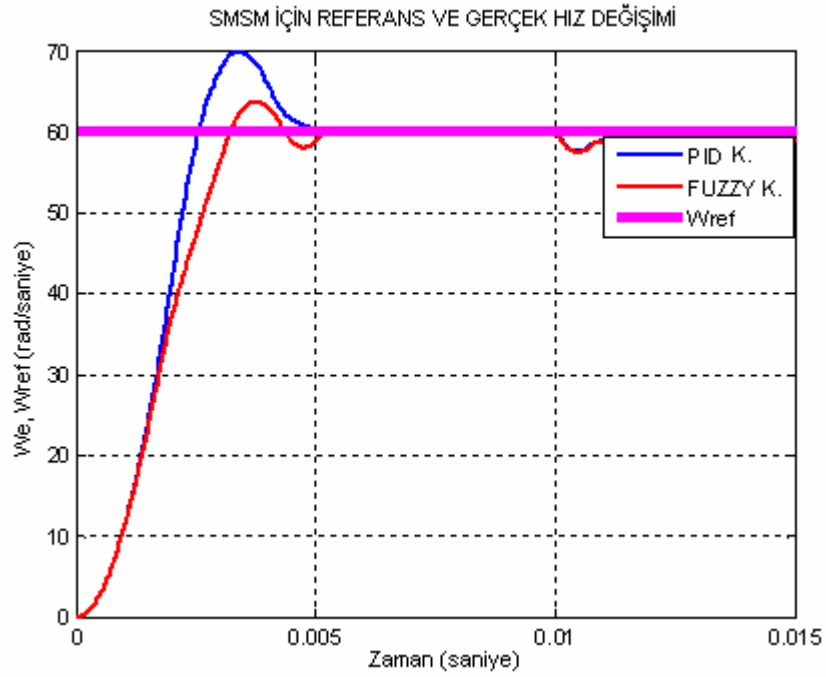
#### 4.2.7 Ölçüm blokları ve simülasyon sonuçları

Makina daha önceden ayarlanmış referans hıza PID denetleyicisi sayesinde yaklaşık olarak 0.0031 saniyede ulaşıyor (Şekil 4.32). Bulanık denetleyici kullanıldığında ise, bu süre daha da azalıp 0.0025 saniye gibi çok kısa bir sürede istenilen hıza ulaştığı görülmektedir. Buna ilave olarak da taşma (overshoot) miktarların bulanık denetleyicide daha az olduğu, PID kontrolde ise sistemin cevap hızının biraz daha yavaş olmasına rağmen bu taşma miktarı da daha fazla olmaktadır. Yapılan deneyde

en iyi PID kontrol yapılarak Bulanık kontrol ile karşılaştırıldığından karşılaştırma sonuçları daha doğru sonuç vermektedir.

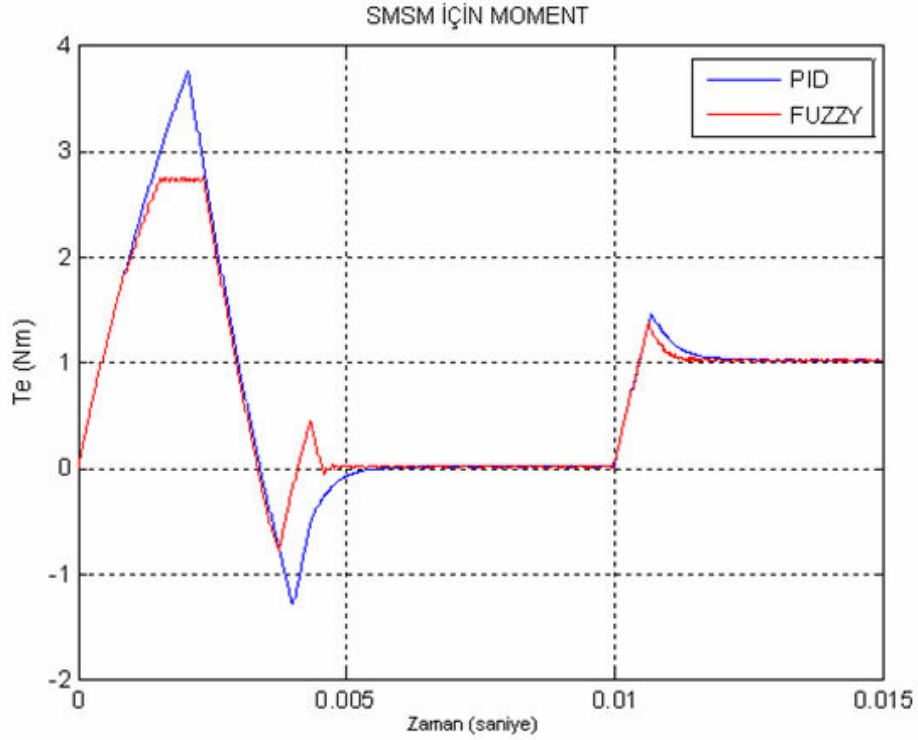


Şekil 4.31: (a), (b) Ölçüm blokları



Şekil 4.32:  $t=0.015$  saniye için için referans ve gerçek hızların ( $w_e$ ,  $w_r$ ) Bulanık kontrol ile PID kontrol için değişimleri



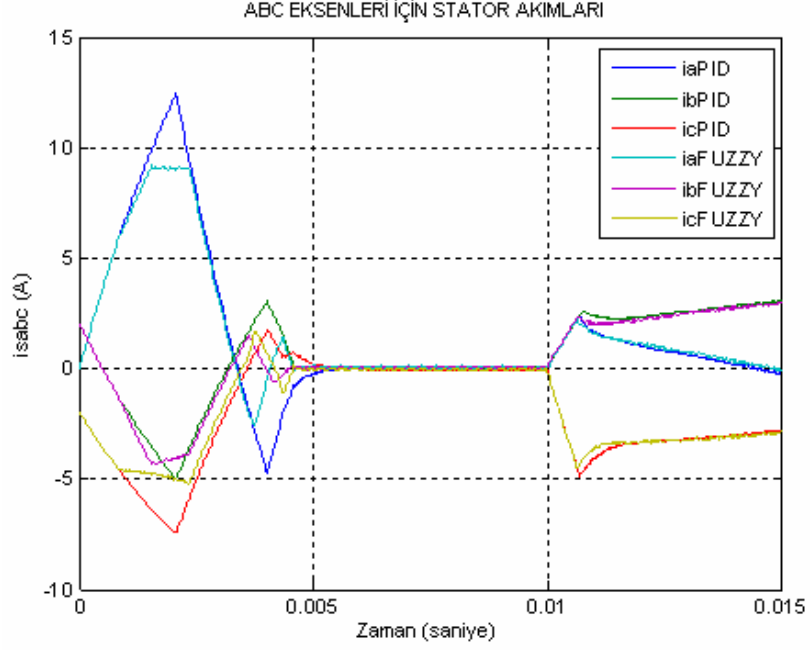


Şekil 4.33: Bulanık ve PID kontroller için moment ( $T_e$ ) değişimi

Ayrıca ilk 0.01 saniye süresince sisteme yük ilave edilmemiş bu süreden sonra ise yük eklenmektedir ve makinaya yük ilave edilmesine rağmen, yine bulanık kontrol bu değişimden çok fazla etkilenmemektedir ve istenilen referans hıza tekrar en kısa zamanda ulaşmaktadır. Yapılan bu simülasyon sayesinde farklı yüklenme durumlarında makinanın nasıl çalıştığı ve çıkış değerlerine nasıl etki ettiği kolaylıkla görülebilir.

Şekil 4.33'ten de görüleceği üzere, hız referans değere oturana kadar moment değerimiz PID kontrolde en fazla 3.8 Nm olurken Bulanık kontrolde bu değer 2.8 Nm seviyesine düşmektedir. Şekilden de anlaşılacağı üzere Bulanık kontrolün kullanılmasıyla momentte bir kısıtlama olmaktadır. Fakat makina kontrollerinde bu değerın kısıtlanması asla istenmez. PID kontrolde momentte bir kısıtlama olmadığı görülmektedir. Buradan moment ifadesi için PID kontrol Bulanık kontrole göre daha üstün denebilir. Bulanık kontrolün olumsuz yönleri içinde bu durum söylenebilir. Ancak bu problemi kural tablosunda yer alan kuralların değiştirilmesiyle kısmen de olsa düzeltmek ya da azaltmak mümkün olabilir. Kural tablosunun değiştirilmesiyle

bu sorun azaltılırken başka sorunların da meydana gelebileceğinden kullanıcının çok dikkatli olması gerekmektedir.

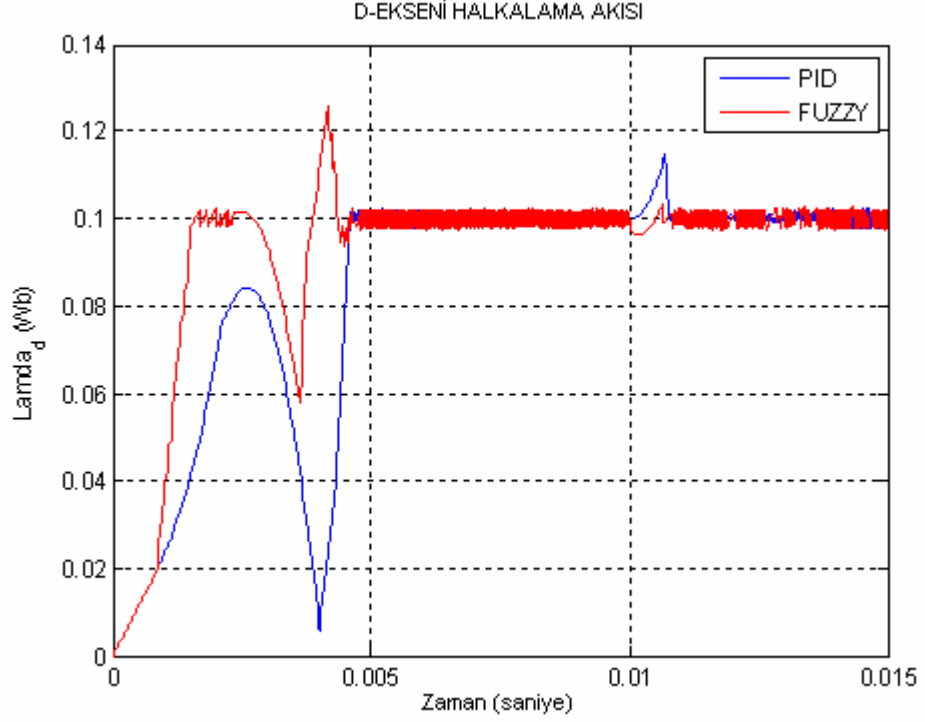


Şekil 4.34: PID ve Bulanık kontrollerde abc eksenlerine ait stator akımları ( $i_{abc}$ )

Şekil 4.34’de, abc eksenlerine ait stator akımları gösterilmektedir. Şekilden de görüldüğü üzere, makina yüksüz çalışırken çok düşük seviyede akım çekilirken yük bağlandığında bu akım değerleri artmaktadır. Bu beklenen bir durumdur. Asıl önemli olan kısım ise, bu akım seviyelerinin ulaştıkları en yüksek değerdir. Akımların tepe değerleri incelendiğinde yüksüz durumda 1 [A] seviyesinin altında iken yük eklendiğinde bu değer 2 [A] seviyelerine ulaşmaktadır. Kullanılan makinanın çalışması için uygulanabilecek nominal akım değeri yaklaşık bu değerlerde olduğu için herhangi bir tehlike içermez. Ayrıca akımın bu seviyede olması da deneysel olarak uygulanabileceğini de göstermektedir.

Şekil 4.35’te ise d-eksenine ait halkalanma akısını göstermektedir. Bu değer kontrol açısından çok büyük önem taşır. Bunun nedeni bu değerın çok fazla olması başka bir değişle sınırların üzerinde olması büyük tehlikeler oluşturabilir. Grafik incelendiğinde makina ilk çalıştığı zaman hızın ani yükselmesi sırasında Bulanık kontrol için bu halkalanma akı değeri en fazla yaklaşık 0.125 [Wb], en az da 0.06

[Wb] olurken PID kontrolde ise en fazla 0.09 [Wb], en az da 0.01 [Wb] olmaktadır. Bu akı ifadesi manyetikleşmeyle ilgili olduğundan pozitif yönde olmasından çok ters yönde olan değer daha çok önem taşır.

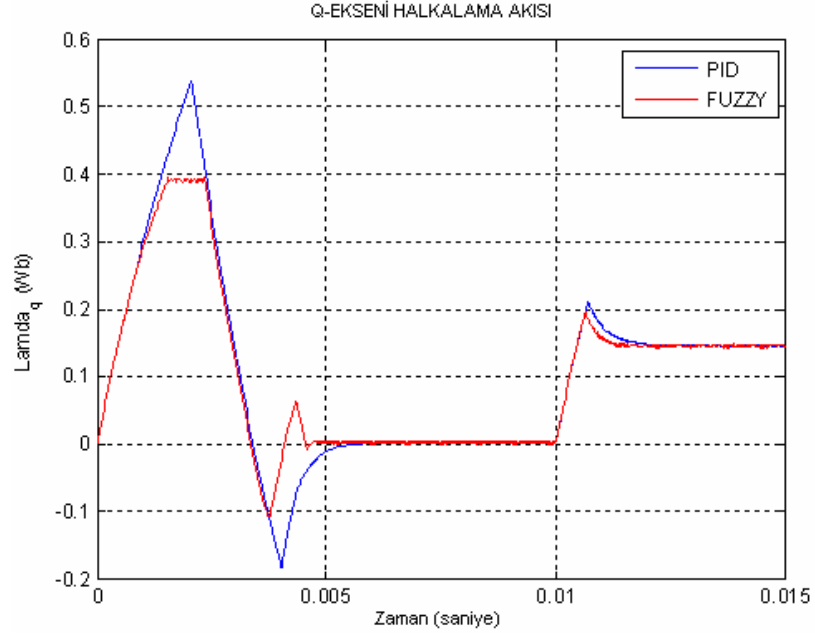


Şekil 4.35: Bulanık ve PID kontrollerde d-ekseni halkalanma akısının ( $\lambda_d$ ) değişimi

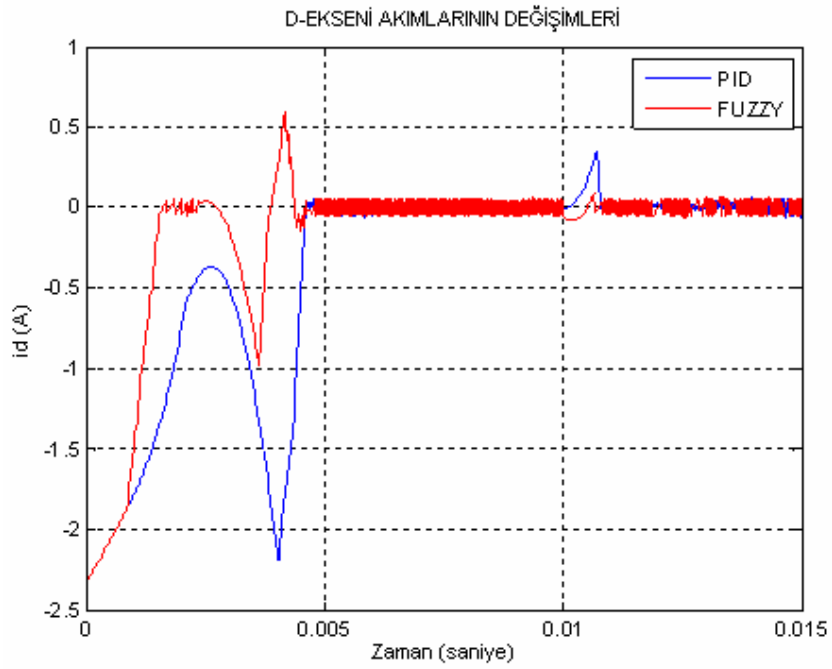
Başka bir deyişle, ters yönde (sıfıra yakın ya da altında) olması ters manyetikleşmeye neden olur. Bu daha büyük bir tehlike oluşturmaktadır. Dolayısıyla burada Bulanık kontrolün daha üstün olduğu söylenebilmektedir. Dikkat edilmesi gereken bir başka husus da, yüklenme sırasında bu değer nasıl etkilendiğidir, makinaya yük eklendiğinde akı değerinde darbe şeklinde ani bir artış olmaktadır. Bu her iki kontrol için de görülebilir ancak önemli olan bu darbenin genlik değeridir. PID kontrolde bu değer daha net görülmektedir. Bilindiği üzere makinada aniden oluşabilecek yüksek değerli darbe şeklindeki değişimler makinaya zarar verebilir ve bu durumdan yola çıkıldığında yine Bulanık kontrolün daha üstün olduğu rahatlıkla söylenebilmektedir.

Şekil 4.36'da, her iki kontrol için q-ekseni halkalanma akı değişimleri gösterilmektedir. Grafik incelendiğinde bu değişimin momentteki değişime benzer

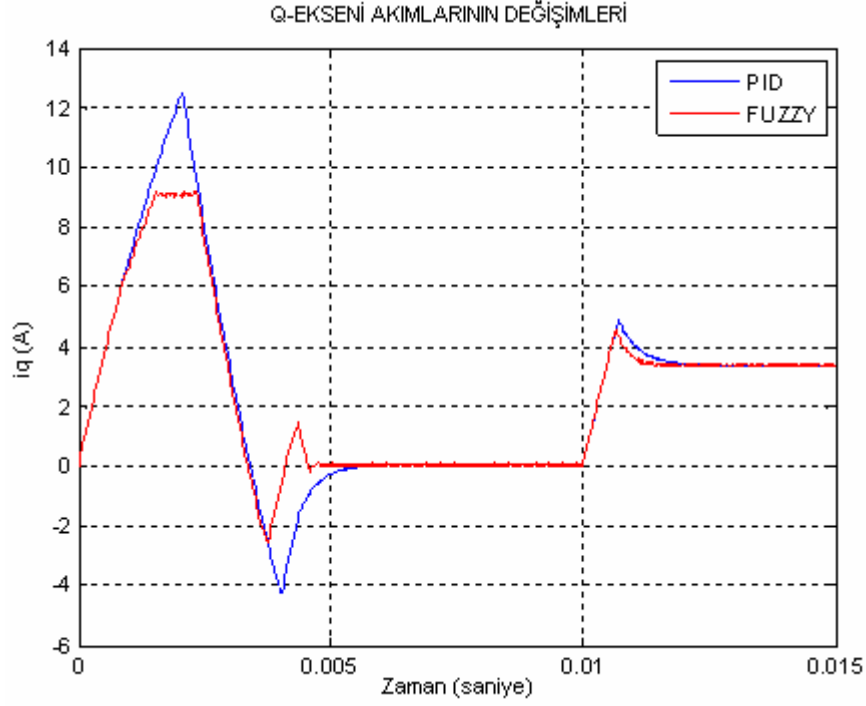
olduđu grlmektedir. Bu da bize yapılan simlasyon sonularının dođru olduđunu ispatlamaktadır.



Şekil 4.36: Bulanık ve PID kontrollerde q-ekseni halkalanma akısının ( $\lambda_q$ ) deđişimi



Şekil 4.37: Bulanık ve PID için d-ekseni akımının ( $i_d$ ) deđişimi



Şekil 4.38: Bulanık ve PID için q-ekseni akımının ( $i_q$ ) değişimi

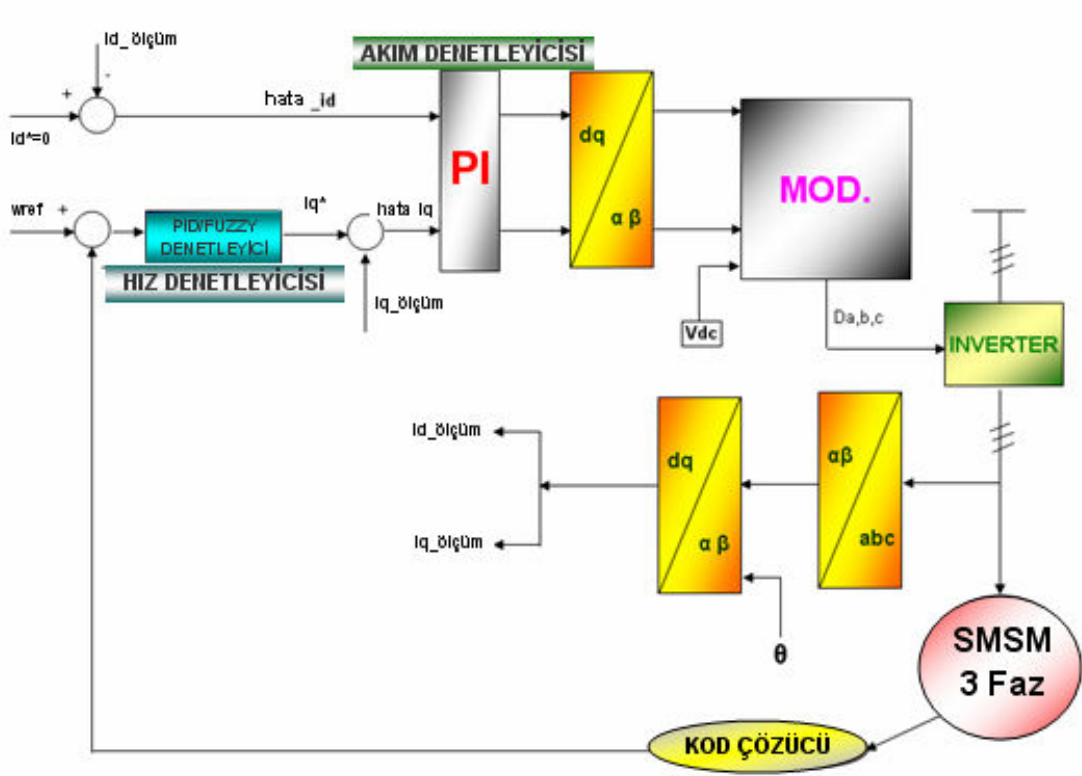
Şekil 4.37’de ise d-eksenine ait stator akımları görülebilir. Stator akımının değişimi halkalanma akısının değişimiyle aynı olduğu anlaşılmaktadır. Şekilde görülen parazitler kuralların değiştirilmesiyle ya da eklenecek filtre devreleriyle azaltılabilir. Buna benzer olarak da Şekil 4.38’deki q-ekseni akımı moment grafiği ile aynıdır.

### 4.3 Deneysel ve Simülasyon Sonuçlarının Karşılaştırılması

Bölüm-2’de Sürekli Mıknatıslı Senkron Motor’un simülasyon sonuçları verilmektedir. Bu bölümde ise, aynı işlemi deneysel olarak yapılmasıyla elde edilen sonuçlar verilmektedir. Simülasyon ve deneysel sonuçlar karşılaştırılıp gerçek bir sistemde Bulanık Mantık yönteminin kullanılabileceği gösterilmeye çalışılmaktadır.

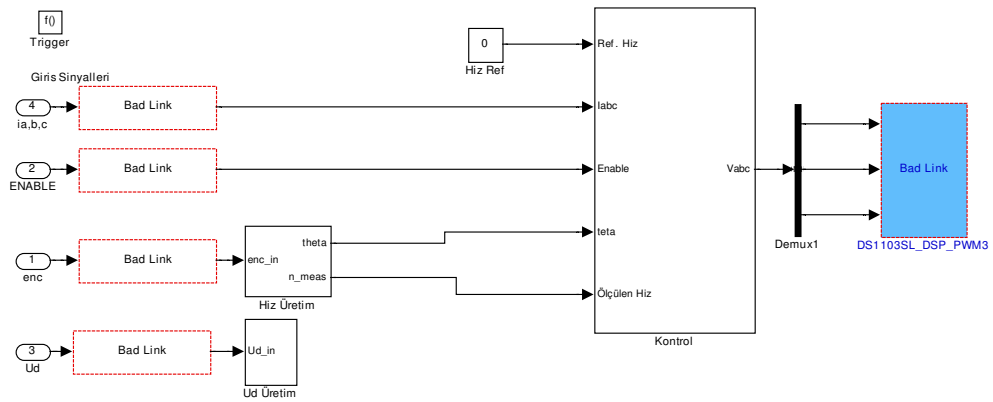
Şekil 4.39’da deneysel çalışma için oluşturulan düzeneğin genel görünüşü yer almaktadır. Simülasyonda oluşturulan kontrol mantığı ile aynı olmasına rağmen gerçek bir sisteme uygulanacağı için bazı küçük eklemeler de yapılmıştır. Simülasyondan farklı olarak motora ait hız değerini ölçebilmek için kod çözücü

(encoder) kullanılmaktadır. Buna ilave olarak da yine gerçek sistem olduğu için inverter bloğun beslemesi üç fazlı olarak şebekeden alınması planlanmıştır.



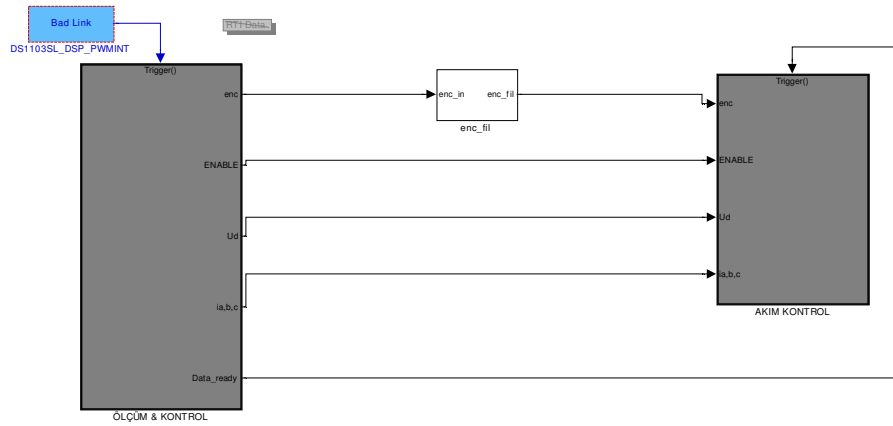
Şekil 4.39: Deneysel çalışma için kontrol prensibi

### 4.3.1 Dspace programında oluşturulan kontrol ve ölçüm blokları



Şekil 4.40: Dspace kontrol blokları

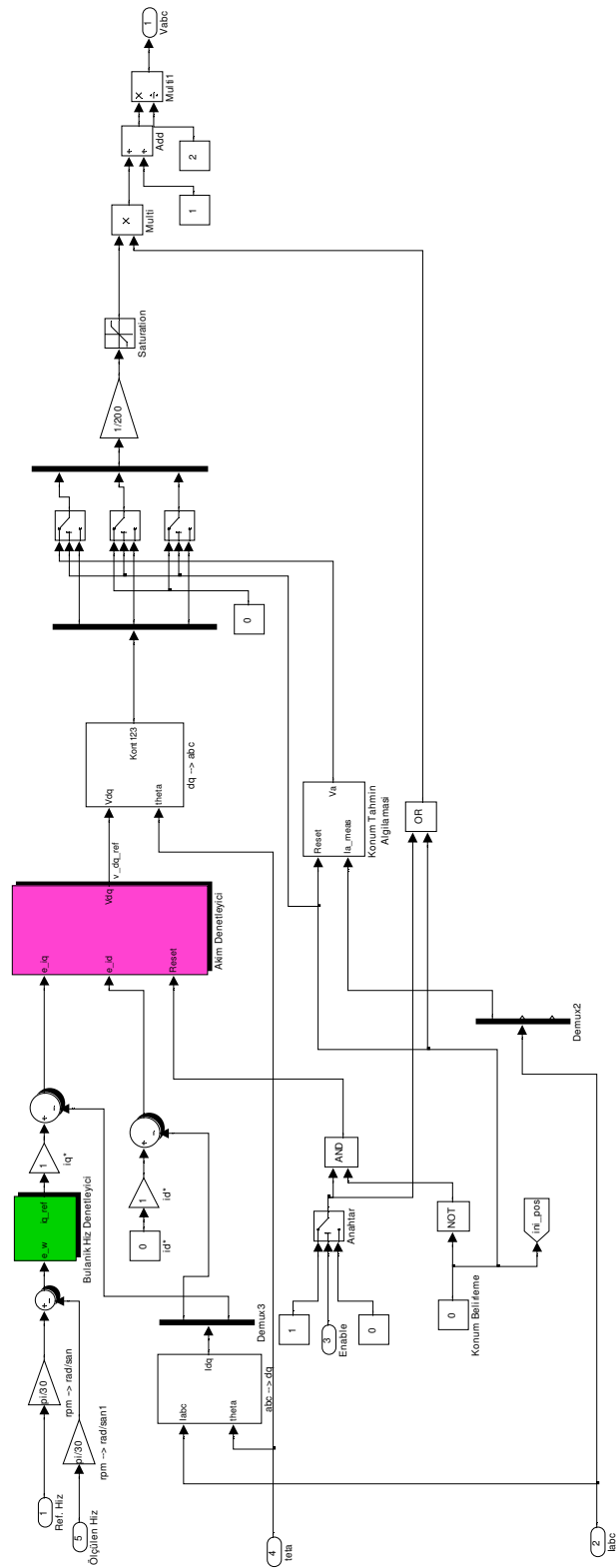
Şekil 4.40'ta ise oluşturulan simülasyonun gerçek motora uygulanabilmesi için kullanılan Dspace isimli bilgisayar programının gereksinim duyduğu bağlantılar gösterilmektedir. Daha önceden oluşturulan simülasyona ek olarak ölçümlerin alınabilmesi ve veri alış verişinin sağlanabilmesi için sadece bu programda yer alan özel bloklar kullanılmaktadır.



Şekil 4.41: Ölçüm ve kontrol blokları

Ayrıca oluşturulan simülasyondan farklı olarak inverter, hız okuyucusu ve motor modeli simülasyondan çıkartılıp gerçek bağlantılarla dışarıdan uygulanmaktadır. Başka bir deyişle bunların modelini kullanmak yerine gerçekleri simülasyona eklenmektedir. Bu sayede yapılan çalışma tam bir gerçek hız kontrolü sağlar. Ölçümlerin ve kontrolün de daha kolay yapılabilmesi için pratik uygulamada ölçüm bloğu ve kontrol bloğu olarak iki farklı bloktan yararlanılmaktadır. Şekil 4.41'de görülen bloklardan faydalanılmaktadır.

Şekil 4.42'de ise, kontrol bloğunun içi görülmektedir. Hız kontrolü için kullanılan Bulanık denetleyici ve akım kontrolünde kullanılan PID denetleyiciler bu blok içerisinde yer almaktadır. Şekil incelendiğinde pratik çalışmada oluşturulan bu mantığın simülasyonda oluşturulan mantıkla neredeyse bire bir aynı olduğu görülebilir. Bu sayede hem simülasyon sonuçları hem de deneysel sonuçlar her iki kontrol yöntemiyle karşılaştırılması daha kolay hale gelmektedir.



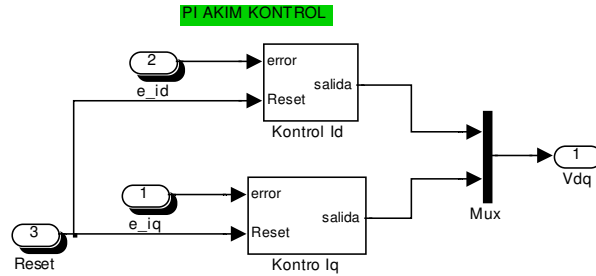
Şekil 4.42: Hız ve akım kontrol blokları



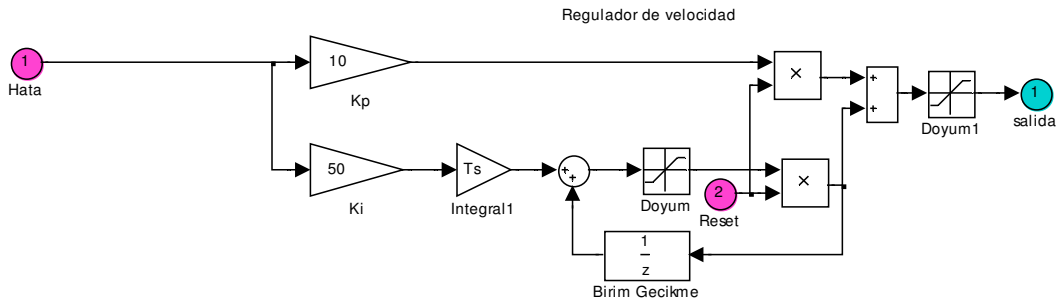
Yapılan simülasyon ve deneysel çalışmaların her ikisinde de Matlab programı içindeki hiçbir hazır blok kullanılmayıp tamamen en baştan tasarlanmaktadır. Bu şekilde hazır araç kutularına ayrıca bir ücret ödemek de gerekmeyeceğinden dolayı maliyeti de düşmektedir. Bu da endüstriyel uygulamalarda büyük önem taşımaktadır.

Şekil 4.42 incelendiği zaman, simülasyonda kullanılan kontrol bloklarından bazı farklılıklar bulunduğu görülmektedir. Simülasyonda akım kontrolüne gerek duyulmadığından bloklar kullanılmamaktadır. Deneysel çalışmada ise, çıkış değerlerinde istenilen değerlerin akım kontrol blokları olmadan elde edilemediği için bu bloklar eklenmektedir. Akımları kontrol etmek için PI kontrolünden yararlanılmaktadır.

Kontrol blokları içinde yer alan kazanç parametrelerin seçimi hız kontrolü yokken yapılarak en uygun değerler deneysel olarak da belirlenmektedir. Ayrıca makina bağlantısında kullanılan Dspace bağlantısında ek bazı bloklardan da yararlanılmaktadır. Bu bloklar kontrol için değil sadece motorun kontrol paneline bağlanmasında önem taşımaktadır.



(a)



(b)

Şekil 4.43: (a) Akım kontrolünde kullanılan PI denetleyici bloğu, (b) PI bloğun iç görünümü

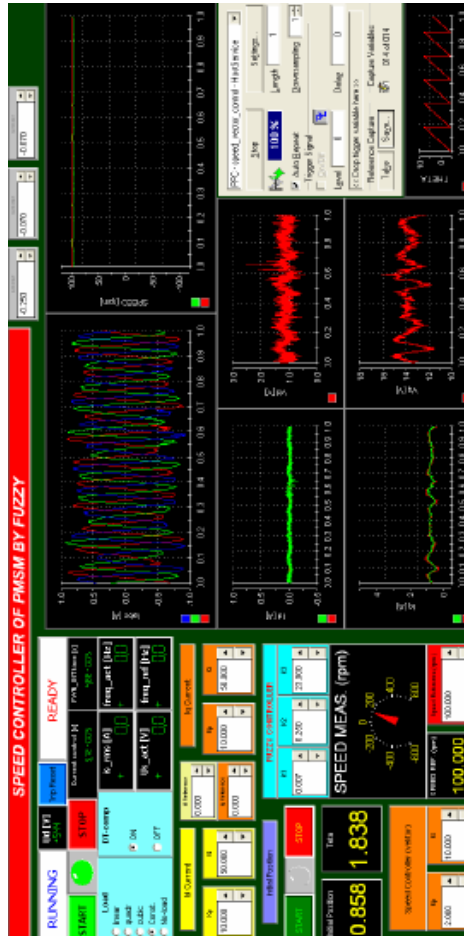
### 4.3.2 Deneysel çalışmada karşılaşılan sorunlar

Deneysel çalışma sırasında birçok problemle karşılaşmıştır. Her bir problemin çözümü için oldukça zaman harcanıp sonunda tüm sorunlar başarıyla çözüme ulaştırılmıştır.

- Darbe genişlik modülasyon (PWM) bloğu: Yapılan deneysel çalışmada karşılaşılabilecek en büyük sorunlardan birisi laboratuvarında bulunan PWM'ye ait bağlantının ne şekilde olması gerektiğinin bulunmasıdır. Bilindiği üzere, Matlab araç kutusu içerisinde birden çok çeşitte PWM blokları yer almaktadır. Ayrıca deneysel sette de yer alan PWM için hangi model olduğu ve ne şekilde bağlanması gerektiğini bulmak çok zor olmaktadır. Bu bağlantılar bulunabilmesi oldukça zaman alabilmektedir.
- İnverter bağlantısı: Karşılaşılabilecek başka bir sorun ise doğru inverter bağlantısının seçilmesidir. Laboratuvarında iki ayrı inverter yer almakta ve bunlardan birisi kontrol amaçlı kullanılırken diğeri ise şebeke için kullanılmaktaydı. Bu nedenle deneysel çalışmada öncelikle doğru inverterin seçilmesi ve yine doğru bir şekilde sisteme bağlanması gerekmektedir.
- PI kontrol bloğu: Akım kontrolünde kullanılan PI blok simülasyonda çok iyi çalışmaktadır. Bilindiği üzere simülasyonda sadece q-ekseni akımı kontrol edilip d-ekseni akımı sıfır seçildiğinden q-ekseni akımının kontrol edilmesi yeterli olmaktadır. Ancak gerçek sistemde bu uygulandığında d-ekseni akımın da bazı sorunlar olduğu görülebilir. Bu nedenlerden dolayı q-ekseni akımı gibi d-ekseni akımın da kontrol edilmesi gerekmektedir. Yapılan deneysel çalışmada her iki akım da PI denetleyici yardımıyla kontrol edilip meydana gelen sorun ortadan kaldırılmıştır. Bu sayede gerçek sistemle teorik sistem arasında farklılık olduğu da açıkça görülmektedir.
- Bulanık kontrolde zaman taşması hatası (task overrun error): Bulanık denetleyicide 7x7 olarak toplam 49 kurallı kural tablosu oluşturulmuştur. Bu gerçek sisteme uygulandığında kullanılan Dspace programında gerçek zaman

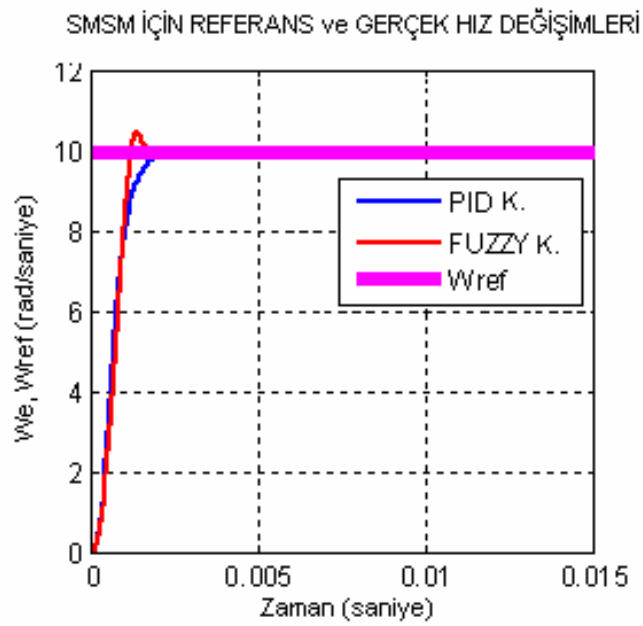
hatası vermiştir. Bunun nedeni kullanılan Dspace programının oluşturulan bulanık kontrol kuralları için yeteri kadar hızlı olmamasıdır. Başka bir neden ise, gerçek sistemde inverter ve PWM içinde yer alan anahtarlama elemanların anahtarlama sürelerinin çok hızlı olmamasıdır. Anahtarlama frekansına ait problemi çözmenin 2 yolu vardır. Birinci yol; Matlab’da oluşturulan kuralların “C, C+ veya Asamble” gibi bir bilgisayar programlama dilinde yazılarak daha hızlı çalışmasını sağlanabilir, ancak çalışmanın kısıtlı bir süre içinde yapılması gerektiğinden bunu başarmak zor olmuştur. Bu amaçla çözüm için tercih edilen ikinci yol ise, oluşturulan kuralların sayısını azaltmaktır. Yapılan çalışmada bu yöntem benimsenmiştir. Kuralların sayısını azaltırken sistemin hassasiyetine dikkat edilerek büyük hataların ortaya çıkması engellenebilmiştir.

### 4.3.3 Deneysel çalışma sonuçları

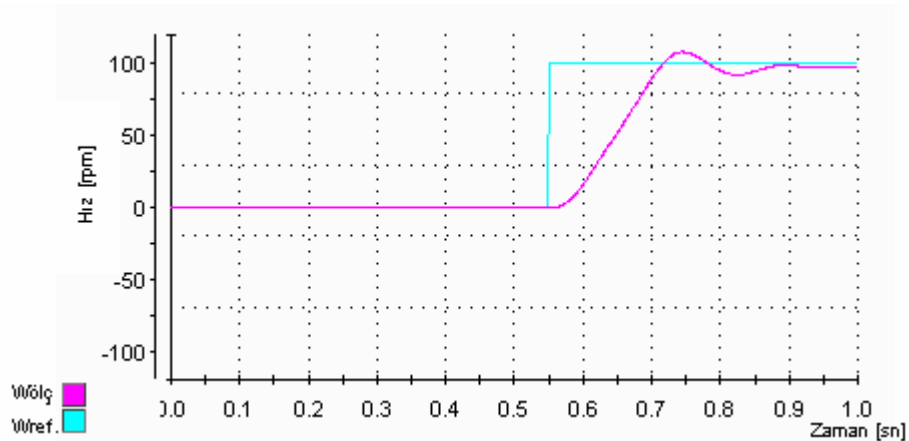


Şekil 4.44: Dspace için deneysel çalışma düzeneği

Laboratuvarında deneysel düzenek kurulduktan sonra Dspace için kontrollerin daha rahat yapılabilmesi ve çıkışların bir arada görülebilmesi için Şekil 4.43'te de görülen düzenek kurulmuştur. Şekil 4.45(a)'da 25 kural ile oluşturulmuş bulanık mantık denetleyicinin hız sonuçları görülmektedir. Daha önce 49 kuralla yapılmış bulanık denetleyicide Dspace'in çok hızlı olmaması nedeniyle ve buna bağlı olarak programın çalışmamasından dolayı kural sayısı azaltılıp bu şekilde karşılaştırma yapılması mümkün olmaktadır. Şekil 4.45(b)'de ise deneysel çalışma ile elde edilen hız değişimi gösterilmektedir.



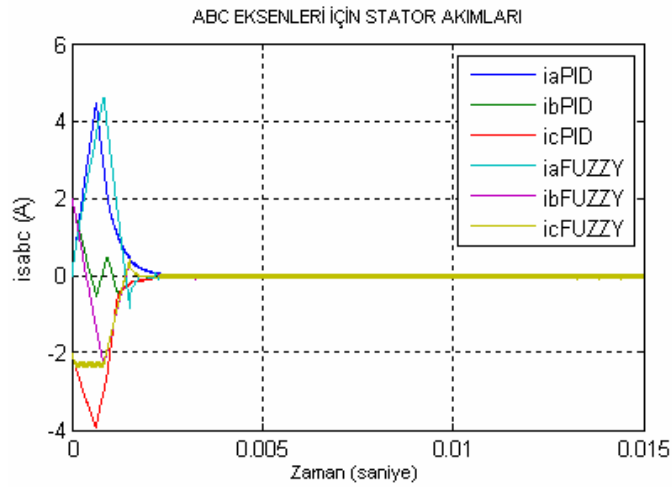
(a)



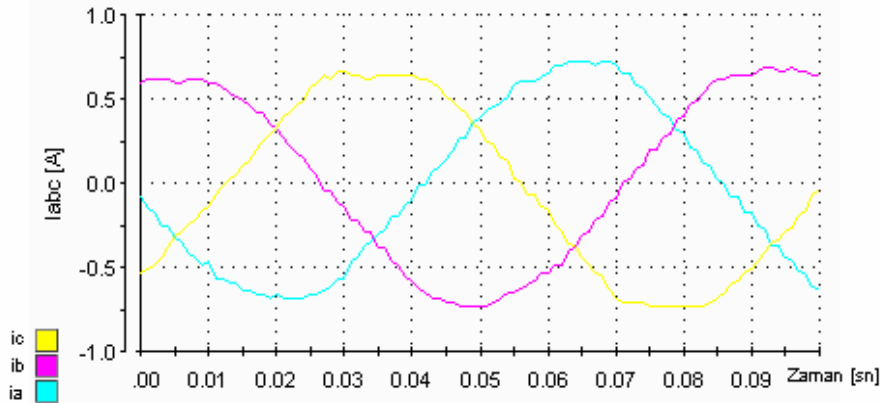
(b)

Şekil 4.45: (a) 25 kural ile yapılmış simülasyon sonucu, (b) deneysel sonuç için hız çıkışı

Şekil 4.46(a)'da 25 kural ile oluşturulmuş bulanık denetleyicinin abc eksenlerine ait stator akımları yer almaktadır. Karşılaştırmak amacıyla da Şekil 4.46(b)'de deneysel olarak elde edilen akım ifadeleri gösterilmektedir. Akım şekillerinde görülen salınımların nedeni Dspace programında seçilen çalışma frekansından dolayıdır. Başka bir deyişle bunun nedeni olarak anahtarlama frekansı da denebilir. Bulanık denetleyicide kural sayısı fazla seçildiği için anahtarlama frekansı da buna bağlı olarak düşük seçilmek zorunda kalmıştır.



(a)



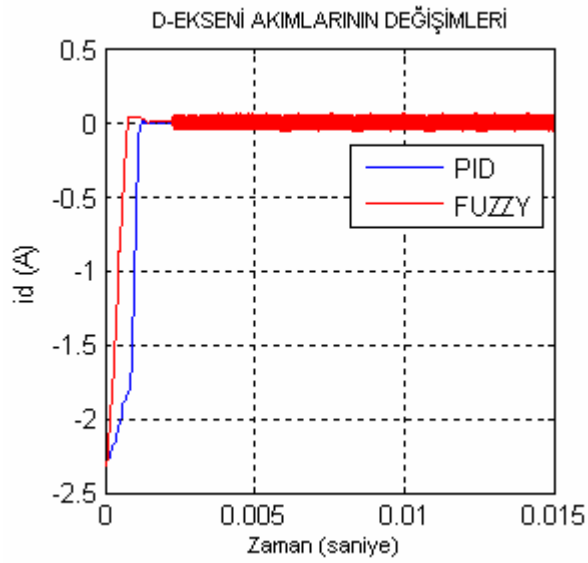
(b)

Şekil 4.46: (a) Simülasyon için abc-eksenlerine ait stator akımlarının gösterilmesi, (b) deneysel olarak abc eksenlerine ait stator akımların gösterilmesi

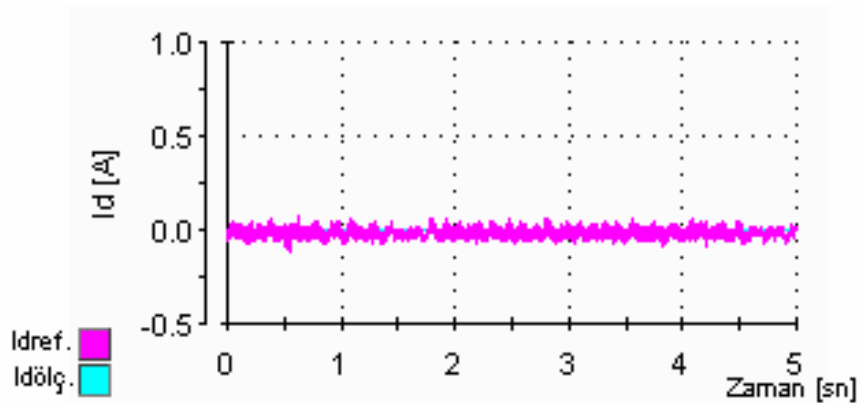
Şekil 4.47(a)'da simülasyon ile elde edilmiş d-ekseni akımı görülmektedir. Burada hem bulanık hem de PID kontrol ile elde edilmiş sonuçlar yer almaktadır. Şekiller incelendiğinde d-ekseni akımının sıfır seviyesinde olduğu deneysel sonuçta biraz

salınım da olsa sıfır olduğu görülmektedir. Bunun nedeni daha önce de belirtildiği gibi, anahtarlama frekansının 5 [kHz] değerinden biraz daha düşük olmasıdır. Sorundan kurtulmanın başka bir yolu da kurallarda değişiklik yapmaktır.

Şekil 4.47(b)'de deneysel olarak bulanık denetleyici sonucu d-ekseni akımının değişimi görülebilir. Şekilden de görüldüğü üzere, d-ekseni akımı sıfır olmaktadır.



(a)



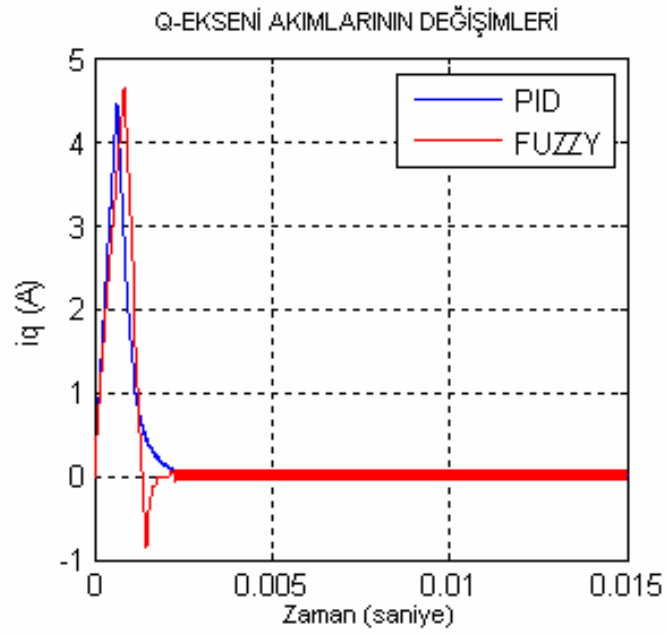
(b)

Şekil 4.47: (a) Simülasyonda d-ekseni akımının gösterilmesi, (b) deneysel olarak elde edilen d-ekseni akımının gösterilmesi

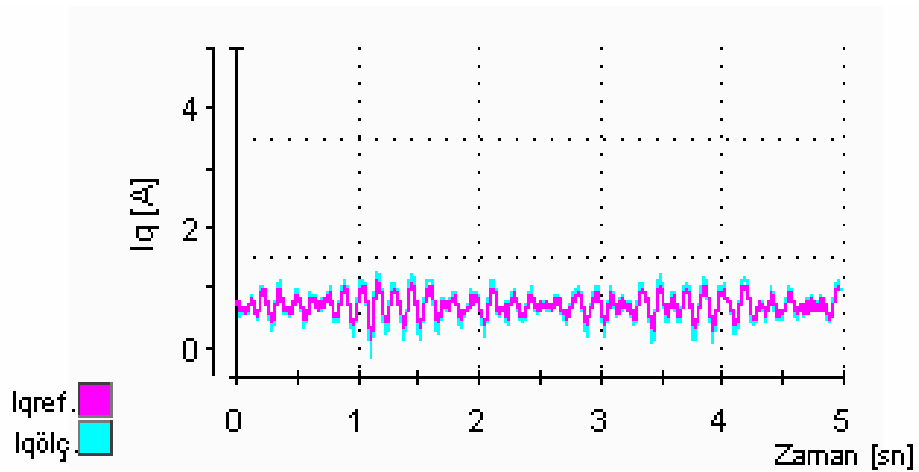
Şekil 4.48(a) ve Şekil 4.48(b)'de bulanık kontrol ile elde edilmiş q-ekseni akımının değişimi gösterilmektedir. Kullanılan kontrole göre bu akım değeri değişerek

istenilen hız kontrolü sağlanmaktadır. Simülasyon ve deneysel sonuçlardan elde edilen çıkış değerlerin birbirleriyle benzer olmalarından dolayı yapılan her iki çalışmanın da doğru olduğu söylenebilmektedir.

Şekiller incelendiğinde akım değerlerinin 1 [A] seviyesinden küçük olduğu görülebilir. Makina için maksimum akım seviyesinin altında çalışıldığından makinaya zarar vermemektedir.



(a)



(b)

Şekil 4.48: (a) Simülasyonda q-ekseni akımının gösterilmesi, (b) deneysel olarak elde edilen q-ekseni akımının gösterilmesi

## 5. SONUÇLAR ve ÖNERİLER

Klasik mantığın temelinde ihtimal hesapları yatar. Sonuç evet ya da hayır kelimeleri ile sınırlıdır. Ancak bulanık mantık cevap olarak olabilirliği değil, ne kadar olduğunu verir. Dolayısıyla, alınan cevap “evet“ ya da “hayır“ kelimelerinden başka bunların ara değerleri de olabilir. Bulanık kontrol yönteminin sağladığı en büyük fayda ‘insana özgü tecrübe ile öğrenme’ olayının kolayca modellenebilmesi ve belirsiz kavramların bile matematiksel olarak ifade edilebilmesine olanak tanınmasıdır. Bu sayede, özellikle lineer olmayan sistemlere yaklaşım yapabilmek uygun olmaktadır. Klasik denetim uygulamalarında karşılaşılan zorluklar nedeniyle bulanık mantık denetimi alternatif bir yöntem olarak ortaya çıkmakta ve modern denetim alanında geniş uygulama alanları bulmaktadır. Örneğin, sesle kumanda edilebilen sistemlerin (makina, tesisat, otomobil telefonu, ev cihazları, bilgisayar) çalışmasını sağlamak amacıyla bulanık denetleyiciler uygulanabilir.

Bu çalışmada amaç, Sürekli Miknatıslı Senkron Motor’un hız kontrolünün önce PID denetleyiciyle, daha sonra da bulanık denetleyici ile gerçekleştirilerek, hem simülasyon hem deneysel olarak karşılaştırılmasıdır. Tezin beşinci ve son bölümünde, tez akışı esnasında karşılaşılan temel adımlar belirtilmekte ve elde edilen sonuçlar, tablo halinde kıyaslanarak yorumlanmaktadır. Ayrıca bu bölümde, bu konuda çalışmak isteyen araştırmacılar için önerilere de yer verilmektedir.

Hız denetiminin hatasız yapılabilmesi için Dspace programının çok iyi bilinmesi gerektiğinden, deneysel çalışmada en uzun zaman alan kısım bu programın öğrenilme süreci olmuştur. Yapılan bu çalışmada Matlab içinde oluşturulan kontrol bloklarında bazı değişiklikler yapılarak gerçek sisteme, başka bir deyişle kontrol edilecek makinaya bağlanılarak, simülasyonda yapılan her durumun deneysel olarak da görülmesi sağlanmıştır.

Deneysel çalışmada kullanılan tüm elemanların birbirleriyle ne şekilde ve hangi sırayla bağlanacağını, bir öğrenme süreci gerektirdiği aşikârdır. Bağlantılarda



yapılabilecek hatalar denetleyicinin çalışmamasına, çalışsa bile motora zarar vermesine neden olabilir. Bu nedenle inverter, PWM kartı ve diğer bağlantılar birçok kez kontrol edilmiştir. Yaşanan en büyük sorunlardan birisi simülasyonda seçilen PWM bloğun gerçek sistemde kullanılan PWM kartıyla uyum sağlamamasıdır. Çünkü PWM için bağlantı giriş ve çıkış sayıları simülasyon içinde gösterilmediğinden, bağlantıların ne şekilde yapılacağına karar verilememiştir. Dspace için bağlantı şekli öğrenildikten sonra doğru bloğun seçilmesi mümkün olmuştur.

Tüm kontroller yapıldıktan sonra, en son aşamada, oluşturulan bulanık denetleyici sisteme eklenerek çıkışlar gözlemlenmiştir. Daha sonra da ek olarak hız kontrolü için vektör kontrol bloğu eklenip genel bir karşılaştırma yapılmıştır. Bu çalışmada vektör kontrolü sadece oluşturulan blokların doğru tasarlanıp tasarlanmadığını anlamak için kullanılmıştır. Bu sayede, bulanık denetleyicinin gerçekten düzgün çalıştığına karar verilebilmiştir.

- Simülasyon ve deneysel çalışma sonuçları değerlendirildiğinde, bulanık denetleyicinin PID denetleyiciye göre daha hızlı ve hassas hız kontrolü sağladığı görülmektedir. Yapılan simülasyon ve deneysel çalışmalar sonucunda elde edilen verilerin daha iyi karşılaştırılabilmesi için Tablo 5.1 ve Tablo 5.2'deki sayısal değerlerden faydalanılabilir.

Tablo 5.1: Bulanık denetleyici için deneysel sonuçlar

	DENEYSEL SONUÇLAR (FUZZY 25 Kural)			
	Simülasyon	Deneysel	FARK	İŞARET
Yükselme zamanı, (s)	0.00125	0.08	0.07875	(-)
Oturma zamanı, (s)	0.0015	0.3	0.2985	(-)
Aşma, (%)	3	3	0	(+)
Referans hıza ulaşma süresi, (s)	0.0013	0.1	0.0987	(-)

Tablo 5.1'de, 25 kuraldan oluşmuş bulanık denetleyici ile yapılan hız denetimindeki, simülasyon ve deneysel çalışma sonuçlarına ait yükselme zamanı

( $T_r$ ), yerleşme zamanı ve aşma ( $\%M_p$ ) miktarları gibi değerler gösterilmektedir. Tablo 5.1 içinde yer alan “+” işareti; deneysel çalışmanın hassasiyet ve hız bakımından daha iyi olduğunu, “-“ işareti ise; simülasyon çalışmasının deneysel çalışmaya göre hız ve hassasiyet bakımından daha iyi olduğunu göstermektedir.

- Tablo 5.2’de ise, Bulanık ve PID denetleyiciler için simülasyon çalışması sonucunda elde edilen değerler gösterilmektedir. Simülasyon çalışmasında, bulanık kontrol içindeki kural tablosu 49 tane kuraldan oluşmaktadır. Tablo 5.2’de yer alan “+” işareti; bulanık denetleyicinin hız ve hassasiyet bakımından daha üstün olduğunu, “-“ işareti ise; PID kontrolün daha üstün olduğunu ifade etmektedir. Her iki tablodan da görüleceği üzere, bulanık kontrol bazı durumlarda çok iyi sonuçlar vermemektedir. Ancak “-“ işaretin olduğu durumlarda bile sayısal veriler birbirlerine çok yakındır. Ayrıca hız denetiminde, bulanık kontrol tasarımının daha kolay, maliyetinin de düşük olmasından dolayı PID kontrol yerine tercih edilmektedir.

Tablo 5.2: Bulanık ve PID denetleyiciler için simülasyon sonuçları

	SİMÜLASYON SONUÇLARI (FUZZY, PID)			
	FUZZY (49 Kural)	PID	FARK	İŞARET
Yükselme zamanı, (ms)	1.8	2.2	0.4	(+)
Oturma zamanı, (ms)	5	5,1	0,1	(+)
Aşma, (%)	1.67	6	4.33	(+)
Referans hıza ulaşma süresi, (ms)	2.5	3.1	0.6	(+)
Maksimum Moment Değeri (Nm)	2.8	3.8	1	(-)
$\lambda_d$ için yük eklendiğinde oluşan maksimum değer, (Wb)	0.108	0.118	0.01	(+)
$\lambda_d$ için yük yokken oluşan maksimum değer, (Wb)	0.125	0.09	0.035	(-)
$\lambda_d$ için yük yokken oluşan minumum değer, (Wb)	0.06	0.01	0.05	(+)

Elde edilen sonuçlar dikkate alındığında, bu tezde sunulan çalışmanın geliştirilip daha da ileriye taşınması için göz önünde tutulması gereken konular aşağıda belirtilmektedir.

- Programı yazan kişinin kural sayısını belirlerken dikkatli olması gerekmektedir. Fazla kuralın olması sistemi hassas hale getirirken kontrol sisteminin yavaşlamasına neden olur. Az kuralın olması ise, kontrol sisteminin daha hızlı çalışmasını sağlarken, hassasiyetten uzaklaştırır. Bulanık denetleyicinin olumsuz yanları da göz önüne alınacak olursa, kural sayısı en uygun şekilde seçilerek sistemdeki yavaşlık sorunu giderilebilir.
- Bulanık kontrol için Matlab programı üzerinde çalışmak yerine, tüm kurallar başka bir programlama dilinde (C, basic, vb) yazılarak sistemin daha hızlı çalışması sağlanabilir. Hazır kontrol bloklarının kullanılması, gerçek sistemlerde bilgisayar ile motor arasında veri alışverişini yavaşlatıp, makina için çıkış değerlerinin hatalı olmasına ya da makinanın hiç çalışmamasına neden olmaktadır.

## KAYNAKLAR

- [1] Hendershot, J.R., Miller, 'Design of Brushless Permanent-Magnet Motors', *UK*, 10-100, (1994).
- [2] Harris, J., 'Fuzzy Logic Applications in Engineering Science', *National University of Science and Technology, Zimbabwe, Published by Springer*, ISBN-13 978-1-4020-4078-8 (2002).
- [3] Mcneill, F. M., Thro, E., 'Fuzzy Logic A Practical Approach', *Published by AP Professional*, ISBN: 0-12-485965-8, (1999).
- [4] Pillay, P., Krishnan, R., 'Modelling of P.M.S.M. drives', *IEEE Transactions of Industrial Electronics*, Vol. 35, No. 4, (1988).
- [5] Zhang, L., Rahman, M. F., Hu, W. Y., and Lim, K. W., 'A direct torque controller for P.M.S.M. drives', *IEEE Transactions of Energy Conversion*, Vol. 14, No. 3, (1999).
- [6] De Azevedo, H. R., Wong, K. P., 'A Fuzzy Logic Controller for P.M.S.M. Sliding Mode Approach', *Department of Electrical and Electronic Engineering, The University of Western Australia, Nedlands, W.A., Australia, IEEE*, TH0406-9/93, (1993).
- [7] Ogasawara, K., Murata, T., 'High Performance Control of P.M.S.M. Based on Magnetic Energy Model by Sliding Mode Control', *Kitami Institute of Technology, Japan, EPE 2005*, ISBN: 90-75815-08-5, (2005).
- [8] Cascella, G. L., Cupertino, F., 'P.M.S.M. Rotor Double-Alignment by PI and Sliding-Mode Controllers', *IEEE, Italy*, ISBN: 0-7803-7817-2/03, (2003).
- [9] Koy, F., Gözde, H., Taplamacıoğlu, C., 'Klasik PID ve Bulanık Mantık Denetleyici ile Senkron Makina Kontrolü', *Doktora Tezi, Elektrik Elektronik Mühendisliği Bölümü, Gazi Üniversitesi, Ankara*, (2000).
- [10] Jing, Z., Yu, C., Chen, G., 'Complex Dynamics in a P.M.S.M. Model', *Published by Elsevier, China*, (1037 1037, A0324626), 831-848, (2004).
- [11] Faiz, J. and Azan, A., Keyhani, A. and Proca, A., 'Closed-Loop Control Stability for P.M.S.M.', PII: SS0142-0615(97)00001-X, 331-337, (1997).
- [12] Wang, S. J., Fang, C. H., Lin S. K., 'A Flux Estimation Method for a P.M.S.M.', *Published by Elsevier*, (034-8853), 335-359, (2004).

- [13] Fung, R. F., Chen, K. W., Yen, J. Y., 'Fuzzy Sliding Mode Controlled Slider-Crank Mechanism using a P.M. Synchronous Servo Motors', **Published by PERGAMIN**, PII: S0020-7403(98)00063-0, 337-355, (1998).
- [14] Emiroğlu, İ., Atkaş, H., 'The Effects of the number of rules on the output of a Fuzzy Logic Controller employed to a P.M. D.C. Motor', PII: S0045-7906(97)00021-9, 245-261, (1998).
- [15] Zhang, L., Rahman, M. F., 'Modelling and Experimental Studies of an Instantaneous Torque and Field Weakening Control Scheme for an Interior P.M.S.M. drive', **Published by ELSEVIER**, PII: S0378-4754 (97) 00142-0, 289-300, (1998).
- [16] Lin, F. J., Fong, R. F., 'A Supervisory Fuzzy Neural Network controller for Slider-Crank mechanism', **Published by PERGAMON**, PII: S0957-4158 (99) 00070-7, (1999).
- [17] Simon, D., 'Design and rule base reduction of a fuzzy filter the estimation of motor currents', **Published by ELSEVIER**, PII: S0888-613X (00)00054-2, (2000).
- [18] Kim, H., Harke, M. C., Lorenz, R. D., 'Sensorless Control of Interior P.M. Machine Drives with Zero-Phase Position Estimation', (2002).
- [19] Rasmussen, H., Vadstrup, P. and Borsting, H., 'Adaptive observer for speed sensorless P.M. motor control', **Aalborg University, DENMARK**, (2006)
- [20] Ghasemi, H., Zadeh, S. V., 'A very fast direct torque control for interior P.M.S.M. start up', **Published by ELSEVIER**, 1-12, 196-890, (2004).
- [21] Bolopion, A., Jouve, D., Pacaut, R., 'Control of P.M.S.M.: A Simulation Comparative Survey', **IEEE**, CH2853-0/90/0000-0374, (1990).
- [22] Shet, D. S., 'Position Sensorless Control of Surface mount P.M.A.C. motors at Low Speeds', **Texas University, IEEE**, (1999).
- [23] Stewart, P., 'Dynamic model reference PI control of P.M.A.C. drives', **Published by PERGAMON**, PII: S0967-0661(01)00071-5, 1255-1263, (2001).
- [24] Kadjoudj M., Abdessemed, R., Golea, N., Benbouzid, M. E., 'Electric Power Components and Systems, Adaptive Fuzzy Logic Control for High Performance P.M. Synchronous Drives', DOI: 10.1080/153250001317094207, (1998).
- [25] Doeuff, R. L., 'A mathematical model for static converters allowing digital simulation of associate machines transients', **IEEE Transactions on Industrial Electronics and Control Instrumentation**, Vol. IECI-24, 789-807, (1977).
- [26] Gayed, A., Benkhoris, M. F., Siala, S. & Doeuff, R. L., 'Simulation Tool for the Study of P.M.S.M. Control Techniques', **IEEE, FRANCE**, 0-7803-2091-3/94, (1994).

[27] Jun, W., Hong, P. and Yu, J., 'A Simple Direct-Torque Fuzzy Control of Permanent Magnet Synchronous Motor Driver', *Proceedings of the 5th World Congress on Intelligent Control and Automation, IEEE, China*, 0-7803-8273-0/04, (2004).

[28] Ang, K.H, Chong, G., 'PID Control System Analysis, Design, and Technology', *IEEE Transactions on Control Systems Technology*, Vol. 13, NO.4, (2005).

[29] Gieras, J.F., Wing, M., 'Permanent Magnet Motor Technology: Design and Applications', Chapter5, 169-177, (2000).

## EKLER

Simülasyon sonuçları için Matlab programı içinde bir dosya (m-dosyası) oluşturulup simülasyona ait çizimlerin daha kolay elde edilmesi sağlanmaktadır. Çizimler için aşağıdaki program kullanılmaktadır.

```
clc;
close all;
'SPEED CONTROL OF PMSM BY PID and FUZZY'
'Student : Aykut TEKER'
'Supervisors: Ewen RITCHIE and Feriha ERFAN KUYUMCU'
'Aalborg University-Kocaeli University-2007'
load('We.mat');
% load('Wm.mat');
load('Vabc.mat');
load('iabc.mat');
% load('iabcref.mat');
load('Te.mat');
load('id.mat');
load('iq.mat');
% load('iqref.mat');
load('lamda_d.mat');
load('lamda_q.mat');

figure(1);plot(Lamda_q(1,:),Lamda_q(2,:),'b',Lamda_q(1,:),Lamda_q(3,:),'r');xlabel('
Time (sec)'); ylabel('Lamda_q (Wb)'); title ('FLUX LINKAGE OF Q-AX='); grid;
legend('PID','FUZZY');

figure(2); plot(Lamda_d(1,:),Lamda_d(2,:),'b', Lamda_d(1,:), Lamda_d (3,:),'r') ;
xlabel ('Time (sec)');ylabel('Lamda_d (Wb)');title('FLUX LINKAGE OF D-AX='
);grid; legend('PID','FUZZY');
```

```
figure(3); plot(iq(1,:), iq(2,:), 'b', iq(1,:), iq(3,:), 'r'); xlabel('Time (sec)') ; ylabel('iq (A)'); title('THE CURRENT OF Q-AX='); grid ; legend ('PID','FUZZY');
```

```
%figure(4);plot(iqref(1,:),iqref(2,:));xlabel('Time(sec));ylabel('iqref (A)'); title('THE REFERENCE CURRENT OF Q-AX='); grid; legend ('PID','FUZZY');figure(5); plot(id(1,:), id(2,:), 'b', id(1,:), id(3,:), 'r'); xlabel('Time (sec)'); ylabel('id(A)'); title('THE CURRENT OF DAX='); grid; legend('PID','FUZZY');
```

```
figure(6);plot(iabc(1,:),iabc(2,:),iabc(1,:),iabc(3,:),iabc(1,:),iabc(4,:),iabc(1,:),iabc(5,:),iabc(1,:),iabc(6,:),iabc(1,:),iabc(7,:));xlabel('time in sec'); ylabel('=abc (A)'); title ('STATOR CURRENTS OF ABC-AX='); grid; legend ('iaPID', 'ibPID','icPID', 'iaFUZZY', 'ibFUZZY', 'icFUZZY');
```

```
%figure(7);plot(iabc(1,:),iabc(2,:),iabc(1,:),iabc(3,:),iabc(1,:),iabc(4, :)); xlabel('time in sec');ylabel('=abc (A)'); title('REFERENCE STATOR CURRENTS OF ABC-AX='); grid;legend('PID','FUZZY');
```

```
figure(8); plot(Te(1,:), Te(2,:), 'b', Te(1,:), Te(3,:), 'r') ; xlabel ('Time (sec)') ; ylabel ('Te (NO)') ; title('TORQUE OF PMSM'); grid ; legend('PID','FUZZY');
```

```
figure(9);plot(Vabc(1,:),Vabc(2,:),Vabc(1,:),Vabc(3,:),Vabc(1,:),Vabc(4,:),Vabc(1,:), Vabc(5,:),Vabc(1,:),Vabc(6,:),Vabc(1,:),Vabc(7,:)); xlabel('Time (sec)'); ylabel ('Voltage, (V)'); title('abc Voltages'); grid; legend ('VaPID','VbPID', 'VcPID', 'VaFUZZY', 'VbFUZZY', 'VcFUZZY');
```

```
%figure(10); plot(Wm(1,:),Wm(2,:)); xlabel('Time (sec)'); ylabel('wm, (rad/sec)'); title('THE ROTOR SPEED OF PMSM'); grid;legend('PID','FUZZY');
```

```
figure(11);plot(We(1,:),We(2,:), 'b', 'Linewidth',2); hold on; plot (We(1,:) , We(3,:) , 'r', 'Linewidth',2); plot(We(1,:),We(4,:), 'm', 'Linewidth',4); xlabel ('Time (sec)'); ylabel ('We, Weref (rad/sec)'); title('REFERENCE VE REAL SPEED OF PMSM'); grid; legend('PID', 'FUZZY','Wref'); hold off;
```



## ÖZGEÇMİŞ

30.01.1980 yılında Gaziantep'te doğdu. Ortaokul öğrenimini Ankara Gazi Osman Paşa İlköğretim okulunda, lise öğrenimini Ankara Sincan Süper Lisesi'nde tamamladı. 2000 yılında girdiği Kocaeli Üniversitesi Mühendislik Fakültesi Elektrik Mühendisliği Bölümü'nden 2005 yılında mezun oldu. Mezun olduğu yıl Kocaeli Üniversitesi Fen Bilimleri Enstitüsü Elektrik Mühendisliği'nde Yüksek Lisans'a başladı. Lisans eğitimi sırasında Çift Ana Dal programına (ÇAP) dâhil olarak Kocaeli Üniversitesi Elektronik ve Haberleşme Mühendisliği Bölümü'nden 2006 yılında mezun oldu.

Erasmus programı çerçevesinde, Nisan-Ekim 2007 tarihleri arasında Danimarka'da bulunan Aalborg Üniversitesi'nde misafir araştırmacı olarak bulundu.

2006 yılının ilk gününden itibaren kadrolu olarak Elektrik Mühendisliği Bölümü'nde Araştırma Görevlisi olarak görev yapmaktadır.