



MARMARA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ



SERVO SİSTEMLERİ İÇİN BİR AKILLI MAKİNA DURUM İZLEME MODELİ GELİŞTİRİLMESİ

Hayri MUTLU

YÜKSEK LİSANS TEZİ

Mekatronik Mühendisliği Dalı

DANIŞMAN

Prof. Dr. Caner AKUNER

EŞ-DANIŞMAN

Dr. Öğr. Üyesi Gazi AKGÜN

İSTANBUL, 2022



MARMARA UNIVERSITY
INSTITUTE FOR GRADUATE STUDIES
IN PURE AND APPLIED SCIENCES



**DEVELOPMENT OF THE INTELLIGENT
MACHINE CONDITION MONITORING
MODEL FOR SERVO SYSTEMS**

Hayri MUTLU

MASTER THESIS

Department of Mechatronic Engineering

Thesis Supervisor

Prof. Dr. Caner AKUNER

Thesis CO- Supervisor

Assist.Prof Dr. Gazi AKGÜN

ISTANBUL, 2022

MARMARA ÜNİVERSİTESİ FEN BİLİMLERİ ENSTİTÜSÜ

Marmara Üniversitesi Fen Bilimleri Enstitüsü Yüksek Lisans Öğrencisi Hayri MUTLU'nun “**Servo Sistemleri İçin Bir Akıllı Makine Durum İzleme Modeli Geliştirilmesi**” başlıklı tez çalışması, 06/01/2022 tarihinde savunulmuş ve jüri üyeleri tarafından başarılı bulunmuştur.

Jüri Üyeleri

Prof. Dr. M. Caner AKUNER (Danışman)

Marmara Üniversitesi (İMZA).....

Dr. Öğr. Üyesi Uğur DEMİR (Üye)

Marmara Üniversitesi (İMZA).....

Dr. Öğr. Üyesi Koray ERHAN (Üye)

Gedik Üniversitesi (İMZA).....

ONAY

Marmara Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulu'nun 12/01/2022 tarih ve 2022/01-02 sayılı kararı ile Hayri MUTLU'nun Mekatronik Mühendisliği Dalı Yüksek Lisans derecesi alması onanmıştır.

Fen Bilimleri Enstitüsü Müdürü
Prof. Dr. Bülent EKİCİ

MARMARA UNIVERSITY INSTITUTE FOR GRADUATE STUDIES IN PURE AND APPLIED SCIENCES

Hayri MUTLU, a Master of Science student of Marmara University Institute for Graduate Studies in Pure and Applied Sciences, defended his thesis entitled “Development of The Intelligent Machine Condition Monitoring Modal For Servo Systems”, on 01/06/2022 and has been found to be satisfactory by the jury members.

Jury Members

Prof.Dr. M. Caner AKUNER (Advisor)
Marmara University(SIGN)

Assist.Prof. Dr. Uğur DEMİR (Jury Member)
Marmara Üniversitesi(SIGN)

Assist.Prof. Dr. Koray ERHAN (Jury Member)
Marmara Üniversitesi(SIGN)

APPROVAL

Marmara University Institute for Graduate Studies in Pure and Applied Sciences Executive Committee approves that Hayri MUTLU be granted the degree of Master of Science in Department of Mechatronic Engineering on 12/01/2022 (Resolution no: 2022/01-02).

Director of the Institute
Prof. Dr. Bülent EKİCİ

ÖNSÖZ / TEŞEKKÜR

Bu tez çalışmasının fikir aşamasından başlayarak desteklerini esirgemeyen, yol gösteren danışmanım Sayın Prof. Dr. M. Caner AKUNER'na

Karşılaştığım her türlü sorunu aşmamı sağlayan kontrol ve YSA tasarım konularında sabırla beni yönlendiren ve destekleyen danışmanım Sayın Dr. Gazi AKGÜN'e

Tüm yüksek öğrenim hayatım boyunca beni yönlendiren ve destekleyen Sayın Prof. Dr. Nihat AKKUŞ'a

Gerekli tüm gereçlerin sağlanmasın da SIEMENS SAN. Ve TIC A.Ş ve Zeynel ERDOĞAN'a

Tüm bu süreçte benzer konularda birlikte çalıştığım öğretmen arkadaşım Hasan TAŞDEMİR'e

Ve tez çalışması süresince sabır ve desteklerini esirgemeyen aileme teşekkürü bir borç bilirim.

İÇİNDEKİLER

ÖZET	vii
ABSTRACT.....	viii
ŞEKİLLER LİSTESİ	xi
TABLO LİSTESİ.....	xii
1. GİRİŞ ve AMAÇ	1
2. METOT VE YÖNTEM	4
2.1 Servo Sistemler	4
2.1.1 Servo Motorlar	5
2.1.2 Servo Sürücüler.....	6
2.1.3 Servo Sistemin Devreye Alınması	8
2.2 Yapay Sinir Ağları	9
2.2.1 YSA Eğitilmesi	10
2.2.2 Veri Setinin oluşturulması:	12
2.2.3 Matlab ile YSA Oluşturma	14
2.2.4 Simulink Kodunun Oluşturulması	16
2.3. Endüstriyel Kontrol.....	17
2.3.1. PLC	17
2.3.2 ProfiNet.....	18
2.3.3 Endüstriyel kontrol sistemin tasarlanması ve programlanması	19
3. DENEY DÜZENİĞİ.....	22
4. SONUÇ	26
5. GELECEK ÇALIŞMALAR	28
KAYNAKLAR/REFERENCES	29
ÖZGEÇMİŞ	33



ÖZET

SERVO SİSTEMLERİ İÇİN BİR AKILLI MAKİNA DURUM İZLEME MODELİ GELİŞTİRİLMESİ

Endüstriyel servo sistemlerin devreye alınması ve parametrelerinin belirlenmesi işlemleri, devreye alan kişinin yetenek ve bilgisi ile sınırlıdır. Ayrıca devreye alınmış sistemler çevresel etkilerin veya yüklerin değişmesi halinde genellikle tekrar optimize edilmezler. Bu tez çalışmasında servo sistemler için oransal, integral ve türev (PID) parametrelerinin sürekli optimum şekilde çalışmasını hedefleyen bir yapay sinir ağı (YSA) modeli geliştirilmesi amaçlanmıştır. Bu işlem için de servo sistemlere girilen parametrelerin kontrolü ve yeniden düzenlenmesi için endüstriyel bir kontrolcü üzerinde servo sistemden elde edilecek anlık üretilen akım, tork, güç, pozisyon gibi verileri kullanan YSA tekniği ile oluşturulmuş bir makine durum izleme algoritması oluşturulmuştur. Bu oluşturulan algoritma ile servo sistemin hız ve akım PID'lerinin parametrelerinin yük karakteristiğinin değişmesi durumunda güncellenmesi sağlanmıştır. İlgili algoritmayı oluşturmak için Siemens S120 servo sürücü ile bir deney düzeneği hazırlanmış ve farklı karakteristikte yükler ile yapılan testler sonucunda eğitim verileri elde edilmiştir. Elde edilen veriler Matlab neural fitting tool box'ı kullanılarak yapay sinir ağlarının eğitiminde kullanılmıştır. Elde edilen model endüstride sıklıkla tercih edilen bir kontrolcü olan Siemens S7-1500 serisi bir programlanabilir lojik kontrolcüye (PLC) entegre edilmiştir. PLC'ye entegre edilen yapay sinir ağları algoritmasına veri beslemek için servo sürücünden veriler ProfiNET “*isochrom real time*” protokolü kullanılarak toplanmıştır, bu protokol servo sistemin içerisinden düzenli ve eş zamanlı veriler elde etmeyi garanti etmektedir. Elde edilen anlık akım, tork, güç, pozisyon gibi veriler işlendikten sonra PLC üzerinde çalışan yapay sinir ağları algoritmasını beslemek için kullanılmıştır. İlgili algoritmanın üretmiş olduğu yeni servo parametreleri yine ProfiNET protokolü ile servo sürücüye yüklenmesi sağlanmıştır. İlgili deney düzeneğinde yapay sinir ağları tekniği ile geliştirilmiş algoritmanın üretmiş olduğu yeni PID değerlerinin sürücü üzerinden elde edilen hız ve akım değerleri ile klasik Ziegler-Nichols modu kullanılarak elde edilmiş PID değerlerinin kullanılarak tekrarlanan deney sonuçları karşılaştırılmıştır.

ABSTRACT

DEVELOPMENT OF THE INTELLIGENT MACHINE CONDITION MONITORING MODEL FOR SERVO SYSTEMS

The commissioning of industrial servo systems and the determination of their parameters are limited to the skills and knowledge of the commissioner. In addition, commissioned systems are often not re-optimized if environmental influences or loads change. In this thesis, it is aimed to develop an artificial neural network model for servo systems that goals to continuously optimize the PID parameters. In order to control and rearrange the parameters entered into the servo systems in this process, a machine condition monitoring algorithm created with the artificial neural network technique using the instantaneously produced data such as current, torque, power, position to be obtained from the servo system on an industrial controller was created. With this algorithm, the parameters of the speed and current PIDs of the servo system are updated in case the load characteristic changes. In order to create the related algorithm, an experimental setup was prepared with Siemens S120 servo drive and training data were obtained as a result of tests with loads of different characteristics. The obtained data were used in the training of artificial neural networks using the Matlab neural fitting tool box. The resulting model is integrated into a Siemens S7-1500 series PLC, which is a frequently preferred in the industry. The data from the servo driver is collected using the ProfiNET isochrom real time protocol to feed data to the artificial neural networks algorithm integrated into the PLC, this protocol guarantees to obtain regular and simultaneous data from within the servo system. After the data such as instantaneous current, torque, power, position are processed, it is used to feed the artificial neural networks algorithm running on the PLC. The new servo parameters calculated by the relevant algorithm then loaded to the servo driver with the ProfiNET protocol. In the relevant experimental setup, the speed and current values obtained from the driver of the new PID values calculated by the algorithm developed with the artificial neural network technique, and the repeated experiment results using the PID values obtained using the classical Ziegler-Nichols mode were compare..

SEMBOLLER/SYMBOLS

K_d	: Türev kontrolör katsayısı
K_i	: İntegral kontrolör katsayısı
K_p	: Oransal kontrolör katsayısı
ω_f	: Açısal hız
f	: Frekans
p	: Motor kutup sayısı
T	: Tork
i	: Motor Akımı
L	: İndüktans
ψ_m	: Motor kalıcı mıknatıs manyetik akı
w	: Perceptron ağırlık çarpanı
b	: Perceptron bias değeri
I	: Birim matris
μ	: Kombinasyon katsayısı
J	: Jacobian matrisi
e	: Hata vektörü

KISALTMALAR/ABBREVIATIONS

PLC	: Programmable Logic Coontroller
PID	: Proportional Integral Derivative
IRT	: Isochronous Real Time
YSA	: Yapay Sinir Ağları
NN	: Neural Network
Z-N	: Zigler Nicholas
RMS	: Root Mean Square,
WL	: Wave Length,
IAV	: Integrated Absolute Value
MAV	: Mean Absolute Value
SSC	: Slope Sign Change
IGBT	: Insulated-Gate Bipolar Transistor
MQTT	: Message Queuing Telemetry Transport
LM	: Levenberg-Marquardt
IDE	: Integrated Development Environment
SCL	: Structured Control Language
MSE	: Mean Square Error
MLP	: Multilayer Perceptron

ŞEKİLLER LİSTESİ

Şekil 2.1 Akış Şeması	4
Şekil 2.2: Servo sürücü kontrol diyagramı	7
Şekil 2.3: Servo Sürücü güç devresi	7
Şekil 2.4 Sürücü devreye alma adımları	8
Şekil 2.5: Perceptron modeli.....	9
Şekil 2.6 YSA modeli	10
Şekil 2.7 LM metodunda Marquardt parametresinin etkisi	12
Şekil 2.8 Sürücüden okunan ham hız verisi örneği.....	13
Şekil 2.9 RMS özneliği hesaplanmış hız verisi örneği	14
Şekil 2.10 Matlab Neural Net Aracı	15
Şekil 2.11 Öğrenme performans grafiği.....	16
Şekil 2.12 Simulink programı örneği.....	17
Şekil 2.13 ProfiNet telegram konfigürasyonu	19
Şekil 2.14 Ana program stucktogramı	20
Şekil 2.15 Kesme programı stucktogramı.....	21
Şekil 3.1 Eğitim Deney düzeneği.....	22
Şekil 3.2 Dengeli yük.....	23
Şekil 3.3 Dengesiz yük	23
Şekil 3.4 Dengesiz yük değişken momentum	23
Şekil 3.5 YSA eğitim ve seçim akış şeması.....	24
Şekil 3.6 Deney düzeneği bileşenleri.....	24
Şekil 3.7 Çalışmada kullanılan deney düzeneği	25
Şekil 4.1 Karşılaştırmalı Hız-Zaman grafiği.....	26
Şekil 4.2 Karşılaştırmalı Akım-Zaman grafiği	27
Şekil 5.1 Gelecek çalışmalar için hedeflenen deney düzeneği	28

TABLO LİSTESİ

Tablo 4.1 Hız MSE karşılaştırma tablosu	27
---	----



1. GİRİŞ ve AMAÇ

Endüstride hızlı ve hassas konumlandırma özellikleri nedeni ile servo motorlar tercih edilmektedir [1]. Servo motorlar, tepkisini önemli ölçüde etkileyecek birçok ayarlanabilir kazanç parametresi içeren servo sürücüler tarafından kontrol edilirler [2]. Servo motor ve kontrolü için kullanılan sürücü ile beraber oluşturduğu bütüne servo sistem olarak nitelenmektedir. Servo sistem, zamanla değişen ve doğrusal olmayan kontrol yapılarıdır [3] [4] [5].

Servo sistemlerin tahrik ettikleri mekanik yapılar ve yük karakteristiklerine uygun olarak devreye alınması gerekmektedir. Devreye alma işleminde belirlenmesi gereken parametrelerden bazıları da hız ve akım için oransal, integral ve türev (PID) kazanç değerleridir. Bir PID denetleyicisinin arkasındaki temel fikir, bir geri besleme sinyalinden okunan değerin orantısal, integral ve türev yanıtlarını hesaplayarak bu üç bileşenin toplamı ile istenen aktüatör çıktısı için gerekli olan kontrol sinyalini hesaplamaktır [6]. Bu değerlerin belirlenmesinde kullanılan yöntemlerden biri uzman tarafından manuel olarak parametrelerin belirlenmesi bir diğeri ise destekleyen servo sürücülerin auto-tune özelliklerinden yararlanılmasıdır. Endüstriyel servo sistemlerin devreye alınması ve parametrelerinin belirlenmesi işlemleri, devreye alan kişinin yetenek ve bilgisi ile sınırlıdır. Auto-tune özeliği için de yapılan araştırmalar göstermiştir ki, farklı servo ekipmanı için kazanç parametrelerini tam olarak tasarlamaz [2]. Sadece, motora yüklenen atalete dayalı olarak nispeten güvenli ve rijit parametre değerleri atamaktadır, bu da kontrolör performansının bir kısmını feda edebilir. Bu nedenle, yanıtın, uygulamanın ihtiyaçlarını karşıladığından emin olmak için kazanç parametrelerinin yine de bir uzman aracılığıyla manuel olarak ince ayar yapılması gerekir. Genel olarak, kazanç ayarlama işleminin belirli bir yöntemi yoktur. Uzmanın deneyimine bağlıdır ve ayarlama prosedürü de zaman alıcıdır [2]. Ayrıca devreye alınmış sistemler çevresel etkilerin veya yüklerin değişmesi halinde genellikle tekrar optimize edilmezler.

Bu sistemlerin sürekli, her koşulda en verimli şekilde çalışmasını sağlayacak ve sistemlerin devreye alma adımıında uzman personel gereksinimini azaltacak, ilgili sistemin hız ve akım PID parametrelerindeki hatayı algılayarak daha doğru bir değer atamasını sağlayacak bir makine durum izleme algoritmaları geliştirilmesi hedeflenmiştir.

Kande ve arkadaşları tarafından yapılan çalışmada elektrikli makineler için geliştirilmiş, güvenliği artırılmış, düşük maliyetli bir durum izleme sisteminin gelecekteki otomasyon

sistemlerinin anahtarı olduğunu iddia etmiştir [7]. Zhaou ve arkadaşları CNC takım tezgâhları için yapmış oldukları deneysel bir çalışmada ise tezgâhın hata durumu sadece servo motorlardan alınan akım bilgisi kullanılarak yapılmaya çalışılmıştır ve CNC tezgâhının durum bilgisi ile hata durumunun teorik ve pratik bilgi ile örtüşmelerinin yanında şu önemli sonuca da ulaşmışlardır; önerdikleri yöntemin test prensibi basittir ve takım tezgâhı durumunun ağ üzerinden çevrimiçi izlenmesini kolaylaştıran yeni bir sensöre ihtiyaç yoktur. Dolayısıyla yüksek performans-maliyet oranına sahip olduğunu iddia etmişlerdir. Simülasyon ve deneylerin analiz sonuçları bunun uygulanabilir, etkili ve güvenilir olduğunu gösterdiğini belirtmişlerdir [8]. Farklı çalışmalarda ise akıllı endüstri felsefesinde, endüstrinin en önemli bileşenlerinden motorların sürekli online olarak izlenmesi öneminin vurgulandığı görülmüştür [9] [10].

Geleneksel PID ayar yöntem ve algoritmaları yerine yapay sinir ağları ile geliştirilmiş adaptif PID kontrolcünün endüstride doğrusal olmayan sistemlerde dinamik tepkilerin performansının daha iyi sonuçlar ürettiği hem simülasyon hem de deneyler ile test edilmiştir [11] [3] [12]. Bu nedenle geliştirilecek makine durum izleme algoritmasında servo sürücüdeki akım ve hız PID kazanç parametrelerinin değerlendirilip, yeniden hesaplanmasında daha yenilikçi bir model olan yapay sinir ağları ile geliştirilmiştir.

Yapay sinir ağlarını beslemek için farklı tip motorlar için genellikle akım, tork ve titreşim verileri tercih edilmektedir [13] [14]. Fakat bu çalışmamızda ilgili algoritmanın yeni bir sensör ve ekimana ihtiyaç duymaksızın var olan halihazırdaki makinelerde de kullanılabilmesi adına sadece servo sürücülerden alınabilecek veriler tercih edilmiştir.

YSA'nın eğitiminde kullanılacak eğitim verilerini elde etmek için daha önce belirtildiği gibi servo sürücü üzerinden elde edilmesini sağlayacak, farklı yük karakteristikleri uygulayacak bir deney düzeneği tasarlanmıştır. Elde edilen veriler işlendikten sonra Matlab Neural Net fitting aracı yardımı ile ve Levenberg-Marquardt metodu kullanılarak eğitim işlemi tamamlanmıştır [15] [16].

Programlanabilir Mantık Denetleyicisi (PLC), esnekliği, güvenilirliği ve karmaşık sistemlerin kontrolünde nispeten düşük maliyet avantajı nedeniyle genellikle otomasyon sistemlerinin ve saha uygulamalarının gerçek zamanlı denetleyicileri olarak tercih edilir [17] [18]. Endüstriyel otomasyon sistemlerinin vazgeçilmez bir parçası olarak çoğunlukla kontrolcü olarak hem makine ve hem de endüstriyel süreçlerin kontrolünde kullanılırlar. Geliştirilmiş algoritmanın bir bilgisayar veya farklı bir kontrolcünün üzerinde çalıştırmak

yerine PLC'lere entegre etmek var olan sistemlere donanımsal olarak yeni bir yatırım yapmaksızın uygulanabilmesi amaçlanmıştır. PLC'ler yapay zekâ yazılımları koşturmak için geliştirilmemiş olmak ile yapılan çalışmalar sonucunda da PLC'lerin YSA algoritmalarını çalıştırdığı ve performans beklentisini karşıladığı iddia edildiği görülmüştür [16] [19]. Bu neden ile geliştirilmiş YSA algoritması PLC'nin koşturacağı bir programa dönüştürülmüştür.

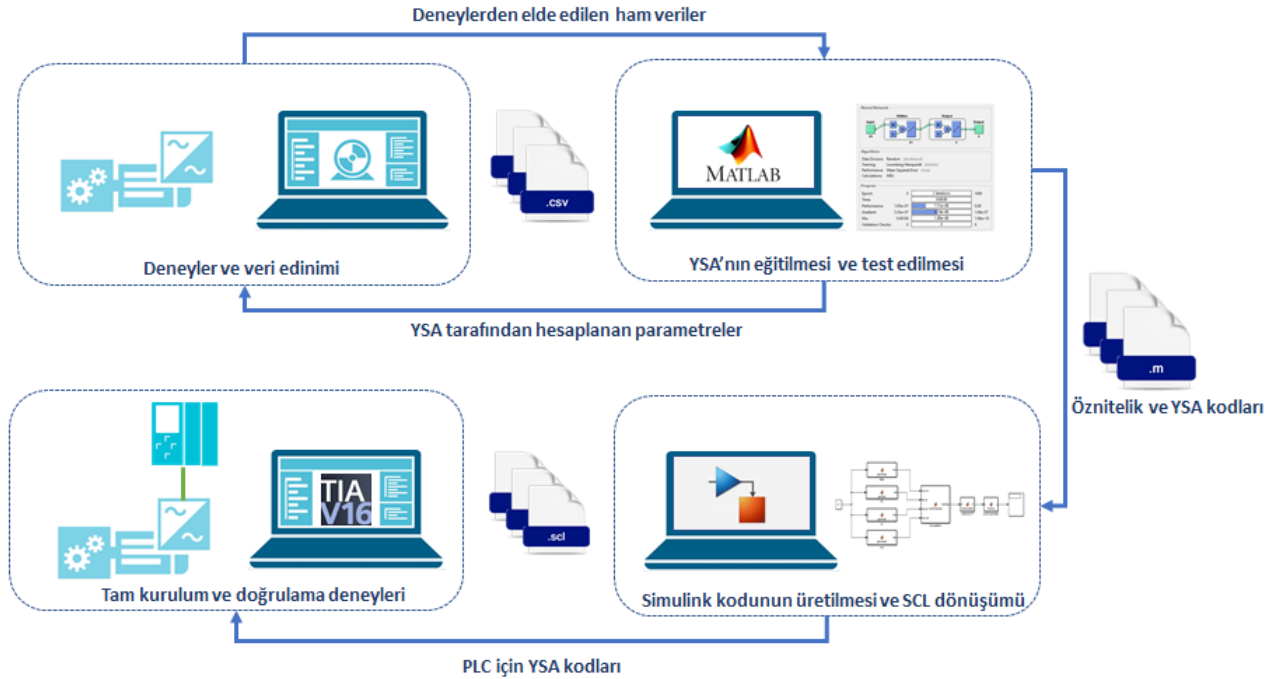
PLC üzerinde koşan bir YSA algoritması ile bu YSA'yı besleyecek verilerin ediniminde kullanılan servo sürücü ile haberleştirilmesi gerekecektir. Fakat YSA'yı beslemek için kullanılacak çapraz verilerin eşzamanlı ve deterministik zaman aralıkları ile edinilmesi gerekmektedir. Bu gereksinimi sağlayacak protokol olarak da ProfiNet IRT gerçek zamanlı haberleşme protokolü tercih edilmiştir. İncelenen çalışmalarda hareket kontrolü uygulamalarında benzeri protokoller ile karşılatıldığında özellikle düşük gecikme, deterministik yapısı sayesinde diğer protokollere göre öne çıktığının belirtildiği görülmüştür [20] [21] [22]. Tüm bu veriler ışığında çözülmesi hedeflenen problem için gerçekleştirilecek akıllı makine durum izleme algoritmasının donanım olarak ProfiNet IRT protokolünü destekleyen bir PLC ve bir servo sistem tercih edilmiştir.

Geliştirilen sistemin performansını test etmek ve karşılaştırmak içinde PID kontrolörlerini ayarlamak için endüstride yaygın olarak kullanılan zigler nicholas Z-N yöntemi tercih edilmiştir [23].

Tezin ilerleyen bölümlerinde sırası ile eğitim için kullanılacak verilerin elde edilmesi için oluşturulmuş deney düzeneği tasarımı tanıtılmış, YSA ve bunu eğitmek için kullandığımız teknikler hakkında bilgi verilmiş, Matlab algoritmasının PLC'ye entegrasyonunun prosedürü açıklanmış ve bu işlem sırasındaki kısıtlamalar belirtilmiş, haberleşme ağının kurulması ve yeniden düzenlenmiş deney düzeneği ile yapılan testlerin prosedürleri tanıtılmış ve son olarak ta yeni geliştirilen sistemin üretmiş olduğu değerler ile manuel olarak devreye alınmış bir servo sistemin çıktıları karşılaştırılmıştır.

2. METOT VE YÖNTEM

Bu çalışmada öncelikle servo sistem devreye alınarak çalışır hale getirilmiş, ardından farklı yük karakteristikleri ve farklı PID parametreleri ile deneyler yapıp veriler elde edilmiştir. Elde edilen veriler YSA eğitiminde kullanılmıştır. Eğitimi tamamlanmış YSA'ya eğitimde kullanılmayan farklı yükler ile elde edilen deneyler yüklenerek performansı test edilmiştir. YSA fonksiyonu Simulink programı olacak şekilde tekrardan tasarlanmıştır. Simulink programından PLC kodları üretilmiştir. Servo sistemden veri edinimi ve parametre yazma fonksiyonlarının üzerine YSA algoritması da bu programa eklenmiştir. Ardından eğitimde kullanılmayan yükler ile tam deney düzeneğinde testler tekrarlanarak klasik yöntemler ile karşılaştırmak ve YSA'nın performansını değerlendirebilmek üzere tekrar veri edinimi sağlanmıştır.



Şekil 2.1 Akış Şeması

2.1 Servo Sistemler

Servo sistemleri bir geri besleme alınan bir senkron motor ve bu motoru kapalı çevrim servo modunda sürmeye haiz bir elektronik güç ve kontrol ünitelerinin toplamı olarak tanımlayabiliriz.

2.1.1 Servo Motorlar

Endüstride pozisyon, hız veya tork kontrolü uygulamalarında kullanılmak üzere DC ve AC olarak iki farklı tipte ve uygulama ihtiyaçlarına göre farklı güç ve hızlarda servo motorlar üretilmektedir. Bu motorların üzerine entegre edilmiş resolver ve enkoder gibi farklı geri besleme birimleri mevcuttur. Enkoderler motorun açısal konumunu belirlemek için kullanılan her bir birim için bir darbe veya darbe dizisi üreten ekipmanlardır. Mutlak enkoderler motorun bulunduğu pozisyonu bir darbe dizi olarak kodlayarak bir üst kontrolcüye gönderebilmektedir [5]. Bir AC motorun hız denklemi 2.1'deki gibidir.

$$\omega_f = 2\pi \frac{f}{p} \quad (2.1)$$

Burada ω_f senkron dönme hızı, f besleme frekansı ve p statordaki kutup sayısı olarak belirtilmiştir. Motorda üretilen tork ise;

$$T = \frac{3p}{2} (\psi_m i_q + (L_d - L_q) i_d i_q) \quad (2.2)$$

Burada T tork, p kutup sayısı ψ_m kalıcı mıknatıs manyetik akı, L_d, L_q d ve q eksenlerinde indüktansı ve i_d, i_q d ve q eksenlerinde akımı belirtmektedir.

Bu çalışma için 6000 d/d ve 0.6Nm tork üretme gücüne sahip ve geri besleme için üzerinde 4096 pulse çözünürlükle çok turlu mutlak enkoder bulunan bir motor tercih edilmiş ve kullanılmıştır. Bu motordan geri besleme bilgisi Drive CliQ olarak bilinen protokol ile servo kontrolcüye bağlanmaktadır. Drive-CliQ Siemens tarafından enkoder verilerini kontrol cihazlarına taşımak için geliştirilmiş Ethernet teknolojisini kullanan bir haberleşme protokolü olup, standart Ethernet ara yüzden ayıran en önemli farklı 24V besleme taşıyan ayrıca 2 pininin var olmasıdır.

2.1.2 Servo Sürücüler

Motor sürücüler motorları farklı kontrol teknikleri ile kontrol etmeyi sağlayan bir kontrol ve motora hareket etmesini sağlayacak elektrik enerjisini üretilmesini sağlayacak bir güç ünitesi olmak üzere iki ana yapıdan oluşurlar.

Endüstride çoğunlukla tercih edilen v/f, vektör ve Servo kontrol karakteristikleri ile güç ünitesinin çıkışlarını kontrol eden yapılara kontrol üniteleri olarak tanımlayabiliriz. Bu kontrol üniteleri motor hareket emirlerini direkt kendi üzerinden alabileceği gibi bir üst kontrol biriminden de alabilmesini sağlayan elektriksel girişler ve/veya haberleşme arayüzlerine sahip olabilirler.

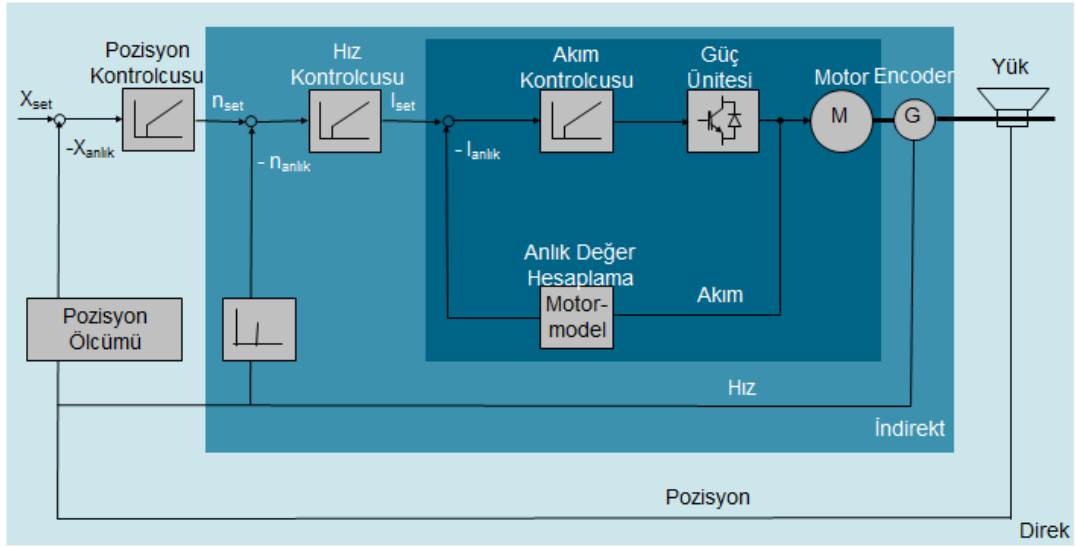
$$N = 120 \frac{f}{p} \quad (2.3)$$

Scalar veya V/f olarak adlandırılan karakteristik yukarıdaki denklemde görüldüğü üzere motor hızını sürücü çıkış frekansını değiştirerek kontrol edilmesini sağlamaktadır. Buradaki N devir f frekansı p ise motorun stator sargısının kurup sayısıdır.

$$\vec{i} = \vec{i}_p + \vec{i}_q \quad (2.4)$$

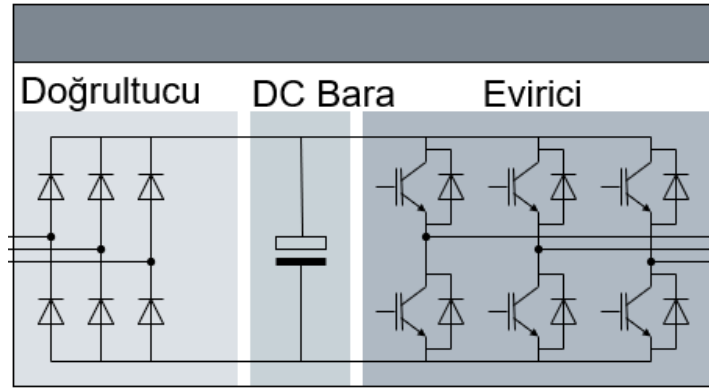
Vektör kontrol ise motor hızının yanı sıra motorun torkunun da kontrolü sağlanmaktadır. Bu kontrol tipinde yukarıdaki denklemde görüldüğü üzere motorun çektiği akım vektörü i bileşenleri olan $\vec{i}_p$ aktif akım ve $\vec{i}_q$ reaktif akım bileşenlerinin arasında 90° olması sağlanmaktadır.

Aşağıdaki şekilde görüldüğü üzere servo kontrol tipinde ise motorun pozisyon, hız ve akımını kontrol etmeyi sağlayan iç içe 3 kontrol algoritmasından oluşmaktadır. Bu kontrol işlemi için seçmiş olduğumuz servo sürücüde kaskad PI kontrol yapılarak motorun hareketi kontrol edilmektedir



Şekil 2.2: Servo sürücü kontrol diyagramı

Servo sürücülerin güç ünitelerini incelediğimizde karşımıza genellikle AC gerilimi DC gerilime çeviren bir doğrultucu, DC gerilimi stabilize etmek için kullanılan bir kapasitör bloğu ile motora iletilecek elektriği kontrol eden bir evirici gurubundan oluşmaktadır. Doğrultucu ve eviriciler için tristor veya IGBT (insulated-gate bipolar transistor) gibi farklı güç elektroniği komponentleri uygulama ihtiyaçlarına göre üreticiler tarafından tercih edilmektedir.



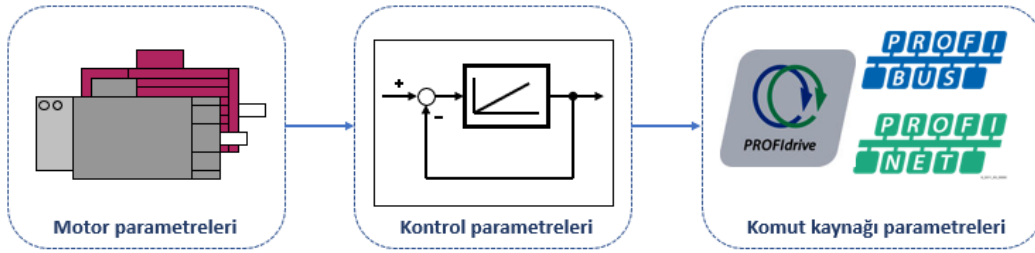
Şekil 2.3: Servo Sürücü güç devresi

Servo sürücü seçilirken hem uygulama ihtiyaçlarına uygun yapıda olması hem de kontrol edilecek motorun ihtiyacı olan gücü üretebilmesi aynı zamanda servo motor üzerinde yer alan geri bildirim ünitesinin ara yüzünü desteklemesi ve son olarak PLC gibi üst kontrol birimlerinden komut almayı ve bilgi vermeyi sağlayacak, üst kontrol birimin desteklediği bir arayüze sahip olması beklenir.

Tüm bunların ışığında bu çalışmada AC 220 V ile beslenebilen doğrultucu ünitesi diyot bloğundan oluşmuş ve evirici bloğu IGBT kullanılmış 0,37 KW güç üretebilen ve Drive-CLiQ encoder okuma ara yüzüne sahip bir güç ünitesi ile Servo modunda motor sürebilen ProfiNet IRT arayüzüne sahip Siemens S120 sürücü ailesinden bir ürün tercih edilmiştir.

2.1.3 Servo Sistemin Devreye Alınması

Bir servo sistemin devreye alınmasında üreticilere göre takip edilmesi gereken farklı prosedürler olabileceği gibi genel olarak 3 ana bölüme ayırabiliriz, ilki sürücünün var ise opsiyon donanımlarının tanımlanması, hükmedeceği motorun tanımlanması, motorun hükmedeceği mekanizmanın tanımlanması, ikinci kısımda kontrol karakteristiği, yükün tipine göre kazanç ve filtre alanlarının belirlenmesi ve son olarak da kontrol komutlarının nasıl alınacağını belirleyip ayarlanması olarak bölümlendirebiliriz.



Şekil 2.4 Sürücü devreye alma adımları

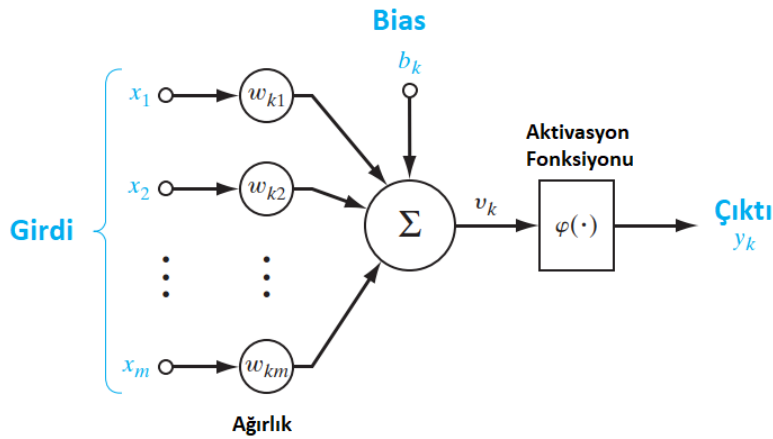
Çoğu üretici bu adımları kolaylaştırmak için sürücü devreye alma yazılımlarının içerisine, devreye alma sihirbazları eklemiştir. Seçmiş olduğumuz Siemens S120 Servo sürücülerde devreye alma sihirbazını kullanarak, donanım tanımlanması işlemi ve servo sürücüye hükmedeceği motorun tanımlanması adımıyla cihazların üzerinde bulunan Drive-CLiQ haberleşme protokolü sayesinde otomatik olarak yapılmaktadır. Sonraki adımlar yine sihirbazlar yardımı ile tanımlanmıştır. Fakat sürücü kontrolünü sağlayacak olan Profinet haberleşme ayarları ilk aşamada sistemin hata vermeden YSA'yı besleyecek deneylerin tamamlanmasına kadar devreye alınmamıştır. Bu kontrol mekanizmasının ayarlanmasında kullanılacak olan şekil 2.13'de görülebileceği gibi telegramların seçilmesi cihaz ismi atanması gibi ProfiNet işlemleri önceden belirlenmiş fakat PLC yazılımı hazırlandıktan sonra cihaza yüklemesi gerçekleştirilerek tam deney düzeneği oluşturulmuştur. İlgili deney düzeneği tezin 3. bölümünde detaylandırılmıştır.

YSA eğitim verilerinin oluşturulması için gerekli olan kontrol sinyalleri Sürücü devreye alma yazılımında var olan Function Generator bölümü kullanılarak yapılmıştır. Deney sonunda elde edilen verilerde Measuring Fonksiyonu kullanılarak dışarıya aktarılması sağlanmıştır.

2.2 Yapay Sinir Ağları

Beyinde bulunan sinirlerin çalışmalarını taklit eden algoritmalara, yapay sinir ağları adı verilmektedir. Bu ağlar, birbirlerine sinaps benzeri yapılarla bağlı yapay nöronlardan oluşmaktadır. Her bir nöron arasındaki bağlantı, o nörondan diğer nörona bilgi taşır. Bilgiyi alan nöron gerekli işlemi yaparak, bu bilgiyi bir sonraki nörona aktarır [24].

Fark ROSENBLATT tarafından geliştirilmiştir ve “perceptron” adı ile anılan yapay bir sinir dış ortamdan veya diğer hücreden alınan girdileri ağırlık yardımı ile hücreye bağlar, ardından toplam girdi toplama fonksiyonu ile hesaplanır ayrıca burada hücrenin ne kadar kolay uyarılabileceğinin ölçütü olan bias parametresi de eklenir. Elde edilen veri aktivasyon fonksiyonundan geçirilmesi ile yapay sinirin çıktısı hesaplanır [25].



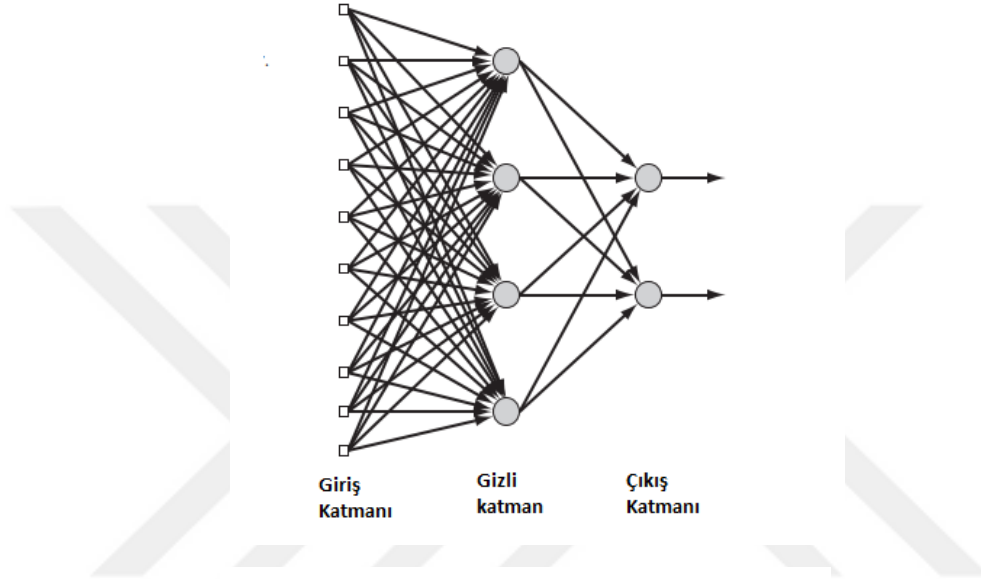
Şekil 2.5: Perceptron modeli

Ayrıca perceptron modelinin çıktı denklemi aşağıdaki denklem 2.5 gösterilmektedir.

$$y = f \left(\sum_{i=0}^n (x_i w_i) + b \right) \quad 2.5$$

Yukarıda gördüğümüz 2.5 nolu formüldeki y çıkış x giriş w ağırlık çarpanı ve b bias parametrelerini gösteren bir fonksiyon olarak tanımlanır [25].

Yapay sinir ağılar, giriş katmanı, gizli katman ve çıkış katmanı olmak üzere en basit haliyle üç katmandan oluşur. Veri setleri giriş katmanından yapay sinir ağlarına girilir. Bu veriler, gizli katmanda ağırlıklar ile çarpılarak aktivasyon fonksiyonundan geçer. Gizli katmandan geçen veriler çıkış katmanına gelerek sonuca ulaştırılır [24]. Yapay sinir ağı tek katmanlı olabileceği gibi aşağıdaki şekilde görüldüğü üzere birbirine bağlı çok katlı yapılardan da oluşabilir.



Şekil 2.6 YSA modeli

Çok katmanlı yapay sinir ağlarına multilayer perceptron (MLP) adı verilmektedir. Bu yapılar birbirlerine bağlanma şekillerine göre ayrıca sınıflandırılabilirler.

2.2.1 YSA Eğitilmesi

YSA'yı oluşturan her bir sinir ağının yapısında bulunan ağırlık değerleri, aktivasyon fonksiyonu ve bias katsayılarının belirlenmesi için yapılan işleme YSA'nın eğitilmesi denmektedir. Bu işlem için veri setinde yerel minimumun hesaplanması gerekmektedir, Yerel Minimum hesaplamada kullanılan birkaç farklı teknik vardır, bunları Levenberg-Marquardt (LM), Bayesian Regularization ve Scaled Conjugate Gradient gibi örneklendirebiliriz. Newton algoritmasının bir başka türü olan LM algoritması YSA ile birlikte ileri beslemeli geriye yayımlı algoritmalar arasında kullanılmaktadır. LM algoritması hızlı yakınsaması, kendi içerisinde az parametre muhteva etmesi ve sadece birinci dereceden kısmi türevler ile işletilmesi bakımından avantajlı bir algoritmadır [26]. Tezimizde tercih ettiğimiz LM algoritması aşağıdaki şekilde tanımlanmaktadır.

$$\Delta w = (J^T J + \mu I)^{-1} J^T e \quad (2.6)$$

Burada w ağırlık vektörü, I birim matris, μ kombinasyon katsayısıdır. e denklem 2.8 de belirtildiği gibi hata matrisi ve J denklem 2.7 detaylandırılan Jakobien matrisini ifade etmekte olup hata matrisinin ağırlıklara göre birinci derece türevlerinden oluşmaktadır [26].

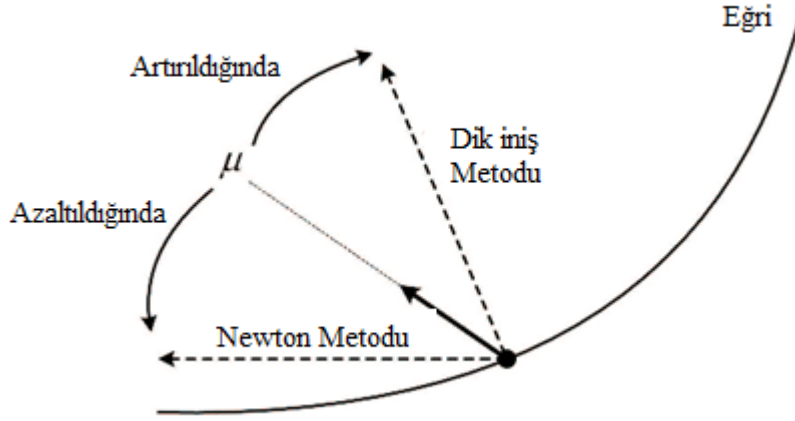
$$J = (PxM)_{xN} \quad (2.7)$$

P eğitim örnek sayısını, M çıkış sayısını ve N ağırlık sayısını (gizli katman nöron sayısı) göstermektedir.

$$e = (PxM)_{xI} \quad (2.8)$$

P eğitim örnek sayısını, M çıkış sayısını ve I birim matris olmak üzere hata vektörünü göstermektedir,

μ Marquardt parametresi olarak adlandırılır. Bu değer LM algoritmasında mütavazi bir başlangıç değeri belirlenerek kullanılmaya başlanan fakat iterasyonlar boyunca güncellenen bir parametredir. Aşağıdaki şekilde de görüldüğü üzere bu parametre çok büyükse yöntemin dik iniş metodu gibi çok küçükse Newton metodu gibi davranmasını sağlamaktadır [27].



Şekil 2.7 LM metodunda Marquardt parametresinin etkisi

2.2.2 Veri Setinin oluşturulması:

Gerçek dünyadan veya simülasyonlardan edinilmiş verilerin, etiketlenmiş ve YSA eğitimi için kullanılmaya hazır hale getirilmiş haline denir. Ayrıca çoğunlukla vektör olarak kullanılan veri setinin özel tanımlanmış bazı özelliklerini gösteren değerlere verilerin öz niteliği olarak adlandırmaktayız [28].

Daha önce belirttiğimiz gibi Siemens S120 servo drive devreye alma yazılımı olan Starter'dan dışarıya aktararak alınmış deney verileri EK 1'e görünen matlab fonksiyonu ile etiketlenmesi sağlanır. Ayrıca bu fonksiyon deneysel verilerin normalizasyonu işlemini de yapmaktadır.

Bu işlemten sonra deneysel verilerin özniteliklerini üretmek için yine Ek 1'de görünen öznitelik oluşturma kısmı kullanılmıştır. Bu kısımda ilk önce verilerin denklem 2.9-2.13'de görülen, sırasıyla root mean square (RMS), wavelength (WL), integrated absolute value (IAV), mean absolute value (MAV) ve slope sign change (SSC) fonksiyonları kullanılmıştır.

$$x_{rms} = \sqrt{\frac{\sum_{i=1}^n x_i^2}{n}}$$

(2.9)

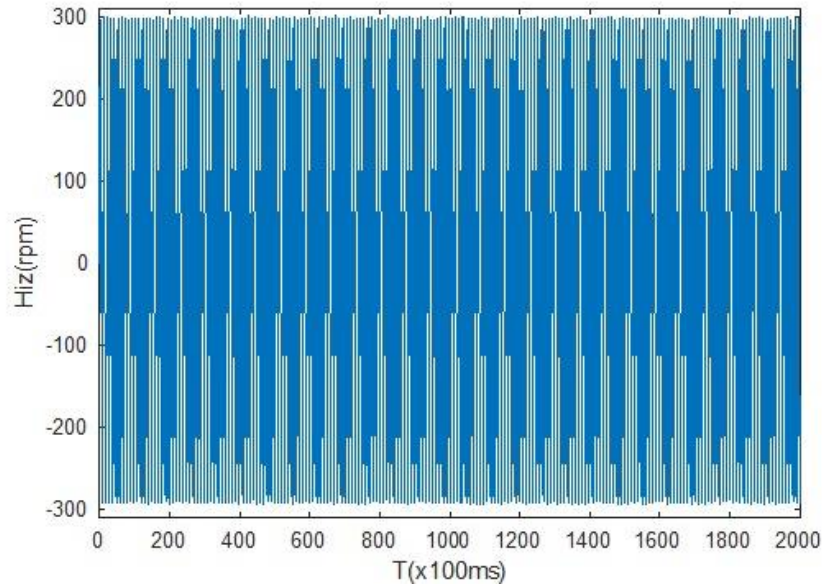
$$x_{wl} = \sum_{i=1}^n |x_i - x_{i-1}| \quad (2.10)$$

$$x_{iaw} = \sum_{i=1}^n |x_i| \quad (2.11)$$

$$x_{maw} = \frac{\sum_{i=1}^n |x_i|}{n} \quad (2.12)$$

$$x_{ssc} = \sum_{i=2}^{n-1} f[(x_i - x_{i-1})(x_i - x_{i+1})] \quad f(x) \begin{cases} 1, & \text{eğer } x \geq \text{eşik} \\ 0, & \end{cases} \quad (2.13)$$

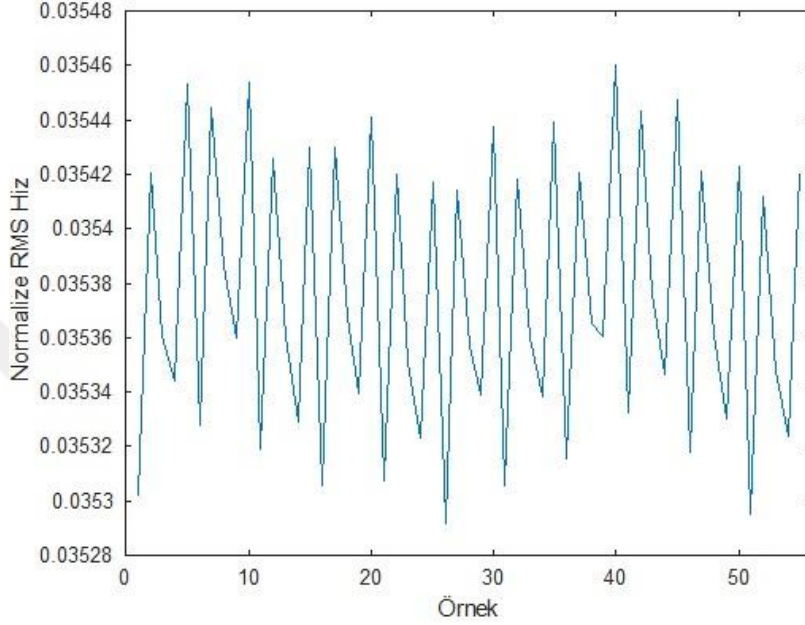
Aşağıda servo sistemin yüksüz durumunda PID değerleri ayarlandıktan sonra yapılan deneyden elde edilen verileri grafik olarak görülmektedir.



Şekil 2.8 Sürücüden okunan ham hız verisi örneği

Elde edilen ham veriler YSA'yı eğitmede kullanılmadan önce normalize edilmiş ve ardından da RMS değerleri 256 değerden oluşan bir örnekleme penceresinde ve 32 örnek

kaydırılarak hesaplanmıştır. Pencere fonksiyonları, sinyal parçalarını çarpan fonksiyonlardır. Bu fonksiyonlar yardımıyla sinyal parçalarının orta kısımlarının vurgulanması sağlanır [29]. YSA eğitiminde kullanılan bu öznitelik verisinin grafiğini de aşağıda görülmektedir.



Şekil 2.9 RMS özniteliği hesaplanmış hız verisi örneği

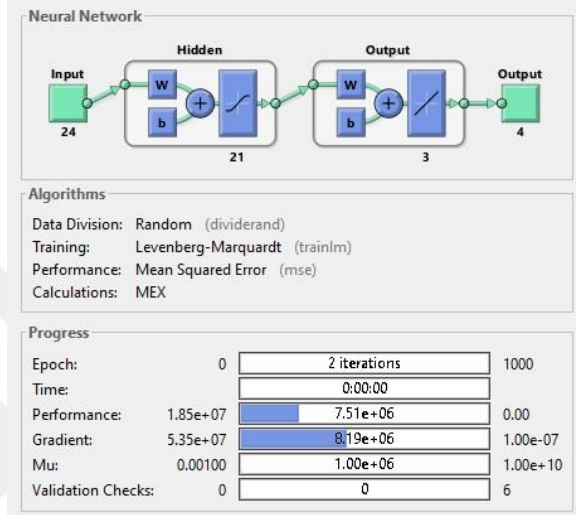
Fakat SSC öznitelik fonksiyonlarının kullanımı PLC'nin döngü süresini oldukça uzamasının sebep olduğu görülmüştür. Bu fonksiyonların işlenmesi sırasında PLC döngü süresinin 5-6 sn kadar uzadığı gözlemlenmiştir. Bu tarama süresi endüstrideki çoğu uygulama için kabul edilemez uzunlukta olduğundan bu fonksiyon öz nitelik hesaplamasından çıkarılmıştır. Bu diğer öznitelik fonksiyonların hepsi PLC döngü süresini birkaç milisaniye uzattığı gözlemlenmiştir ki bu da çoğu endüstriyel uygulama için kabul edilebilir bir seviyedir.

YSA'yı beslemek için kullanılacak öznitelikleri hesaplanmış verilerden oluşan dizinin boyutu da PLC'de tanımlanacak olan bir data blokta saklanacağından maksimum data blok boyutu olan 1.5MB geçmediği de kontrolde edilmiştir.

2.2.3 Matlab ile YSA Oluşturma

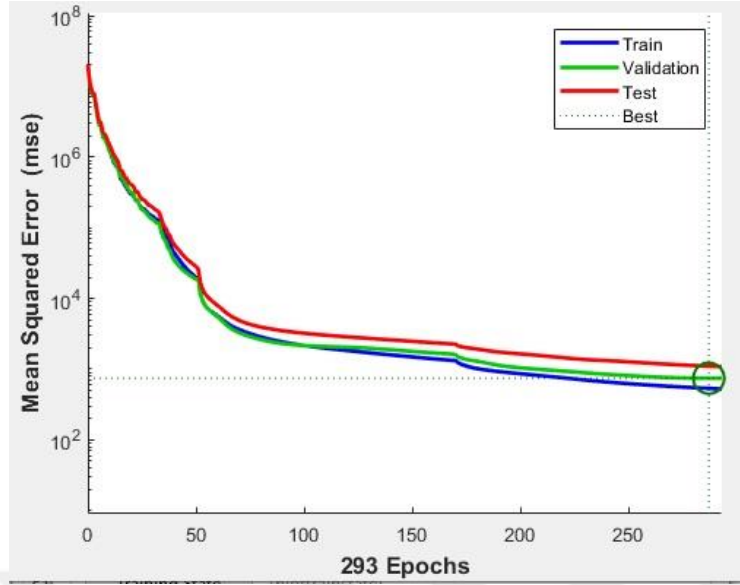
YSA'nın eğitiminde, iki katmanlı ileri beslemeli bir ağ kullanarak, veri sınıflandırma problemini çözmek için Matlab'a entegre edilmiş Neural Net Fitting aracı kullanılmıştır. Bu araç eğitim verilerini, doğrulama verilerini ve test verilerini ayırmamızı, eğitimin

seçilen algoritmaya göre yapılmasını, eğitimin doğrulanmasını, ağ mimarisini tanımlamanıza ve ağı eğitimi sağlayan bir uygulama bileşenidir. Ayrıca bu araç hataların histogramı ve regresyon analizini gibi eğitim performansını değerlendirmemizi sağlayacak birçok çıktı üretmektedir. Son olarak eğitimi tamamlanmış YSA algoritmasında bir Matlab fonksiyonu olarak üretebilmektedir. YSA eğitiminde tümleşik bir çözüm üretebilmesi YSA eğitiminde bu aracı tercih etmemizi sağlamıştır [30].



Şekil 2.10 Matlab Neural Net Aracı

Aşağıda da görebileceğimiz regresyon, performans analizi ve histogramlar kullanılarak yapılan denemeler ile eğitim doğrulama ve test verileri oranları ile gizli katman sayısı, optimize edilmiştir. YSA için tercih ettiğimiz gizli katman sayısı 21'dir.



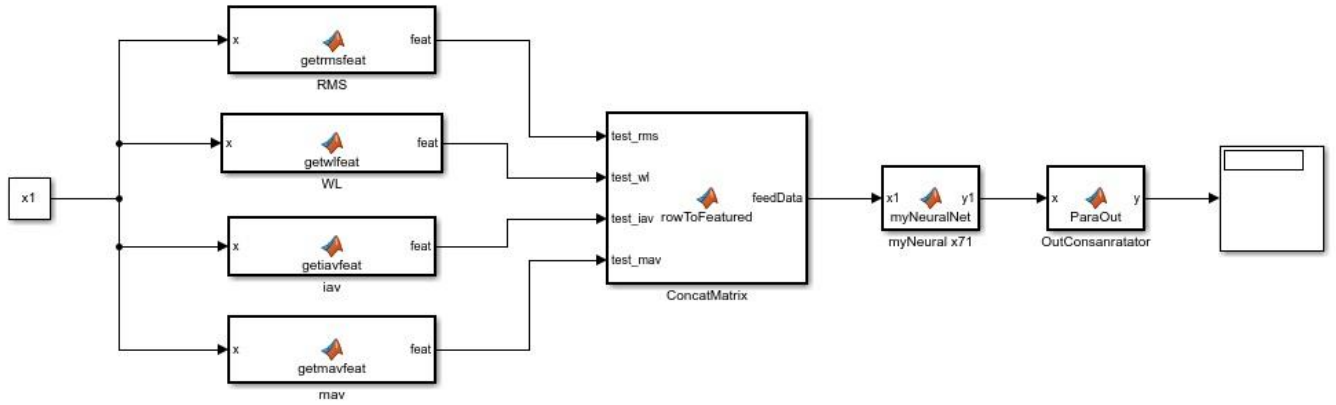
Şekil 2.11 Öğrenme performans grafiği

Ayrıca elde edilen oluşturulan YSA hiçbir şekilde eğitim verisinde kullanılmayan farklı deneyler ile de test edilerek sağlaması yapılmıştır.

2.2.4 Simulink Kodunun Oluşturulması

Elde edilen YSA yı PLC üzerinde çalıştırılmasını sağlamak için hem öznitelik hesaplama YSA besleyecek veri dizinin oluşturulması ve sonucun tekilleştirilmesi için matlab ta kullanılan algoritmaların PLC'ler tarafından desteklenen bir kodlama diline dönüştürülmesi gerekmektedir. Bu işlem için Simulinkte var olan PLC code generator fonksiyonu kullanılmıştır. Bunun içinde matlab fonksiyonları simulinke aktarılmış ve şekil 2.10'da görmüş olduğunuz blok yapısı kurulmuştur.

Bu yapıda da X1 gerçek dünyadan alınan verileri ifade etmektedir. İkinci kısımda RMS, WL, IAV ve MAV fonksiyonları ile öznitelik hesaplaması yapılmakta ardından gelen ConcatMatrix ise öznitelikleri hesaplanmış YSA besleme veri dizisi birleştirilmiştir. Üçüncü işlemde ise YSA çalıştırılmış ve yeniden hesaplanan PID değerleri bulunmuş ve bu veriler Servo sisteme yüklenecek tek bir veri grubu haline getirmek için OurConsantrator bloğuna gönderilmiştir.



Şekil 2.12 Simulink programı örneği

Oluşturulan bloğun doğrulanmasından sonra PLC’lerde değişken boyutlu dizilerin tanımlanmamasından dolayı her bir bloğun giriş ve çıkış değişkenleri sabit boyutlu diziler halinde değiştirilmiştir. Dönüştürme işlemi sırasında ise PLC Code Generator fonksiyonunda hedef IDE “TIA-Portal: Double Precision” seçilmiştir. Bu seçimin sebebi tercih etmiş olduğumuz PLC olan S7-1500’un 64 bitlik bir işlemciye sahip olmasıdır. Bu sayede hem PLC’lerin programlama aracı olan TIA-Portal’a uygun hem de PLC kodu oluşturulurken kullanılacak verilerin veri tiplerinin 64 bit Floating-point veri tipi olan ve PLC’de LREAL olarak tanımlanan tipte olması sağlanmıştır. Daha sonra her bir fonksiyon tek tek PLC programlama standartlarında belirtilen SCL diline dönüştürülmüştür.

Dönüştürülmüş kod dosyaları TIA-Portal yazılımında “external source code” bölümünden PLC projesine alınmış ve FC_NN olarak adlandırılan fonksiyonda birleştirilmiştir. Bkz. Şekil 2.10

2.3. Endüstriyel Kontrol Sistemleri

Endüstriyel ortamlarda kullanılan kontrol sistemlerinden bir tanesi olan PLC’ler ve yine zorlu endüstriyel koşullarda çalışmaya haiz endüstriyel haberleşme protokollerinden ProfiNet bu tezin giriş kısmında belirtildiği üzere tercih edilmiştir.

2.3.1. PLC

Programlanabilir mantık denetleyicileri geniş sıcaklık bandında, titreşim ve elektromanyetik gürültülerin olduğu endüstriyel ortamlarda çalışacak şekilde tasarlanmış cihazlar olup, kararlı çalışma yapıları gereği makine otomasyonundan, süreç otomasyonuna

kadar çok geniş bir kullanım alanına sahiptir. Bu cihazların döngü hızını belirleyen iki ana bileşen program hafıza birimi ve bir komut işleme hızları endüstride kullanılacak spesifik alana tasarlanarak uygulamanın ihtiyaçlarını karşılaması sağlanır. Üzerinde elektriksel giriş ve çıkış üniteleri bulunan tümleşik modelleri olabildiği gibi tamamen modüler şekilde tasarlanmış PLC'ler de mevcuttur.

Tezimizde bu ürün gurubundan Siemens marka S7-1500 serisi ürün tercih edilmiştir. Bu ürün 300KB programlama hafızasına ve 40 ns bit komut işlememe hızı ile orta üst seviye makinelerin kontrolü için tasarlanmış modüler bir PLC'dir. Bu ürünün tercih nedenlerinden ilki Matlab Simülink PLC code generator programı tarafından desteklenen ürünlerden bir tanesi olmasıdır. Bir diğer nedeni ise ProfiNet IRT özeliğini ayrıca bir donanım gerektirmeden desteklemesidir.

Endüstride kullanılan bu kontrolcüler genellikle tek çekirdekli bir CPU kullanılmaktadır. Bu tasarıma sahip PLC'ler çoklu görevleri aynı zamanda yerine getiremezler. Bu tip çoklu görev oluşturulmak istendiğinde işletim sistemlerinde önceden tanımlı kesmeler kullanılmaktadır. Eş zaman aralıkları ile bir işlem yapılması istendiği durumlarda PLC'de bunun için tasarlanmış döngüsel kesmeler tercih edilir.

2.3.2 ProfiNet

ProfiNet standart Ethernet ara yüzü kullanan ve OSI katmanlarından 2. Seviyede haberleşme gerçekleştiren, gerçek zamanlı bir saha haberleşme protokolüdür. ProfiNet uygulama ihtiyaçlarına göre özelleştirilmiş farklı alt protokollerden oluşan bir protokoller bütünü olarak da tanımlana bilir. Bunlara ProfiEnergy, ProfiSafe, Profinet IO v.b. alt kısımlar örnek verilebilir.

ProfiNet IRT ise haberleşme verilerinin önceden belirlenmiş bir senkronizasyon süresinden gönderilip almasını garanti eden gerçek zamanlı haberleşme protokolünün gelişmiş bir türü olarak tanımlana bilir. Bu özelleştirilmiş protokol özellikle hareket kontrolü işlemlerinde veri kaybını engellemesi ve hızı nedeni ile tercih edilmektedir. Bu protokolün kullanımında dikkat edilmesi gereken en önemli husus hem kontrolcünün hem kontrol edilecek cihazların hem de bu iki tip arasındaki bağlantıyı sağlayan tüm ağ elementlerinin ProfiNet IRT desteğinin olmasıdır.

Bu çalışmada haberleştiğimiz cihazların sayısının az olması ve olabilecek en iyi hızda haberleşme sağlayabilmek adına IRT senkronizasyon süresi 1ms olarak belirlenmiş ve PLC'nin network ayarlarında belirtilen şekilde konfigürasyon yapılmıştır.

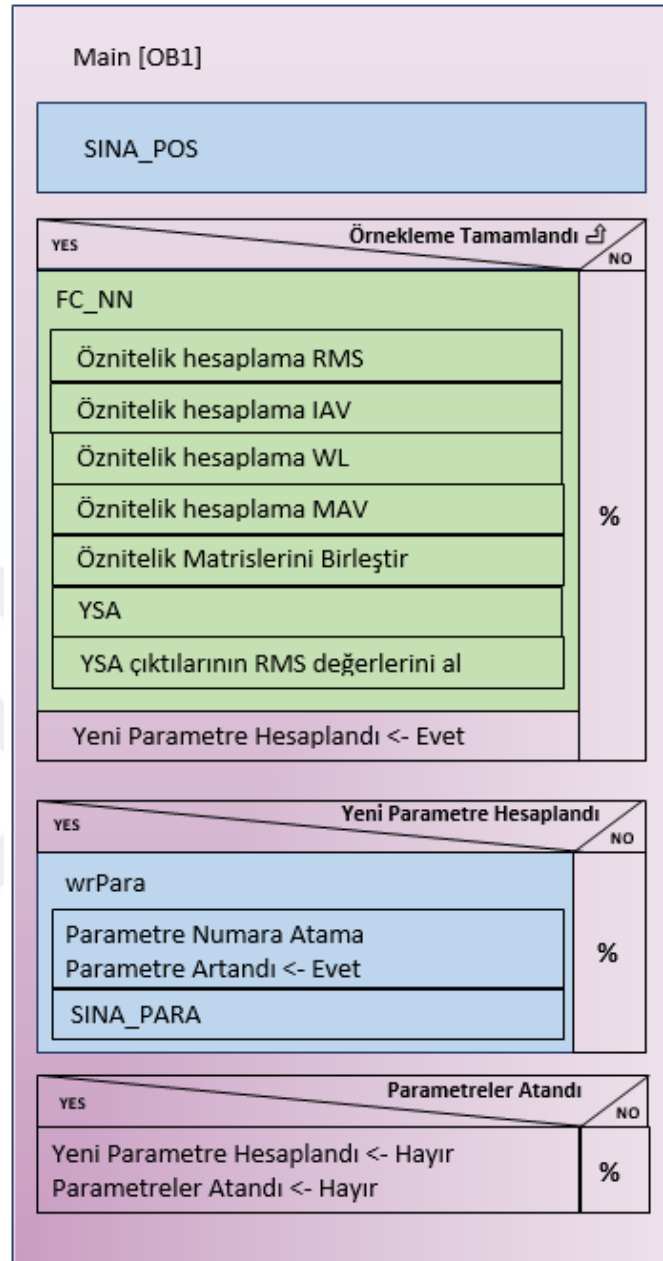
Servo sürücü ile haberleşmede kullanılacak Profinet telegramı hem motora hareket emirleri gönderilmesini sağlayacak hemde yeni hesaplanan PID değerlerini sürücüye yüklemesini sağlamak için Siemens standart telegram 111 kullanılmıştır. Fakat YSA'yı beslemek için gerekli olan anlık ölçüm değerlerini PLC'ye taşıyabilmek için de bu telegramın transmit bölümünün kuyruğuna kullanıcı tanımlı 8 word uzunluğunda bir veri alanı eklenmiştir. Aşağıdaki resimde Starter yazılımında yapılmış telegram konfigürasyonu gösterilmiştir.

PROFINET drive	Variable	Value	Unit	Index
r2089[0]: CO: Send binector	ZSW1	0000	hex	[1]
r2089[3]: CO: Send binector	POS_ZSW1	0000	hex	[2]
r2089[4]: CO: Send binector	POS_ZSW2	0000	hex	[3]
r2089[1]: CO: Send binector	ZSW2	0000	hex	[4]
r2089[2]: CO: Send binector	MELDW	0000	hex	[5]
r2521[0]: CO: LR position ac	XIST_A	0000_0000	hex	[6]
r63: CO: Actual speed smoc	NIST_B	0000_0000	hex	[8]
r2131: CO: Actual fault code	FAULT_CODE	0000	hex	[10]
r2132: CO: Actual alarm code	WARN_CODE	0000	hex	[11]
r68: CO: Absolute current ac	user	0000	hex	[12]
r61[0]: CO: Actual speed un	user-defined	0000	hex	[13]
r70: CO: Actual DC link volte	user-defined	0000	hex	[14]
r72: CO: Output voltage	user-defined	0000	hex	[15]
r80: CO: Torque actual valu	user-defined	0000	hex	[16]
r82[0]: CO: Active power act	user-defined	0000	hex	[17]
r84: CO: Flux actual value	user-defined	0000	hex	[18]
r482[0]: CO: Encoder actual	user-defined	0000_0000	hex	[19]

Şekil 2.13 ProfiNet telegram konfigürasyonu

2.3.3 Endüstriyel kontrol sistemin tasarlanması ve programlanması

Bu çalışmada PLC programı dört ana yapıdan oluşmaktadır. Bunlardan ilki servo sistemi hareket komutlarını gönderilmesini sağlayan ve üretici tarafından temin edilen “DriveLib” kütüphanesinde bulunan Sina_Pos bloğunun çalıştırılmasını sağlayan bölümdür. Bir diğeri 2.y. bölümün de belirtilen öznitelik hesaplama ve YSA programlarını içeren Matlab Simulink tarafından üretilmiş programlar. Bu kısmı şekil 2.11’ deki stucktogramında belirtilen FC_NN isimli function boğun içerisinde toplanmıştır. Üçüncü kısım YSA tarafından düzenlenen yeni parametrelerin yazılmasını sağlayan üretici tarafından sağlanan Sina_Para fonksiyonunun da çağırıldığı wrPara fonksiyonudur. Bu üç parçayı aşağıda belirtilen stucktogramda görebilirsiniz [31].

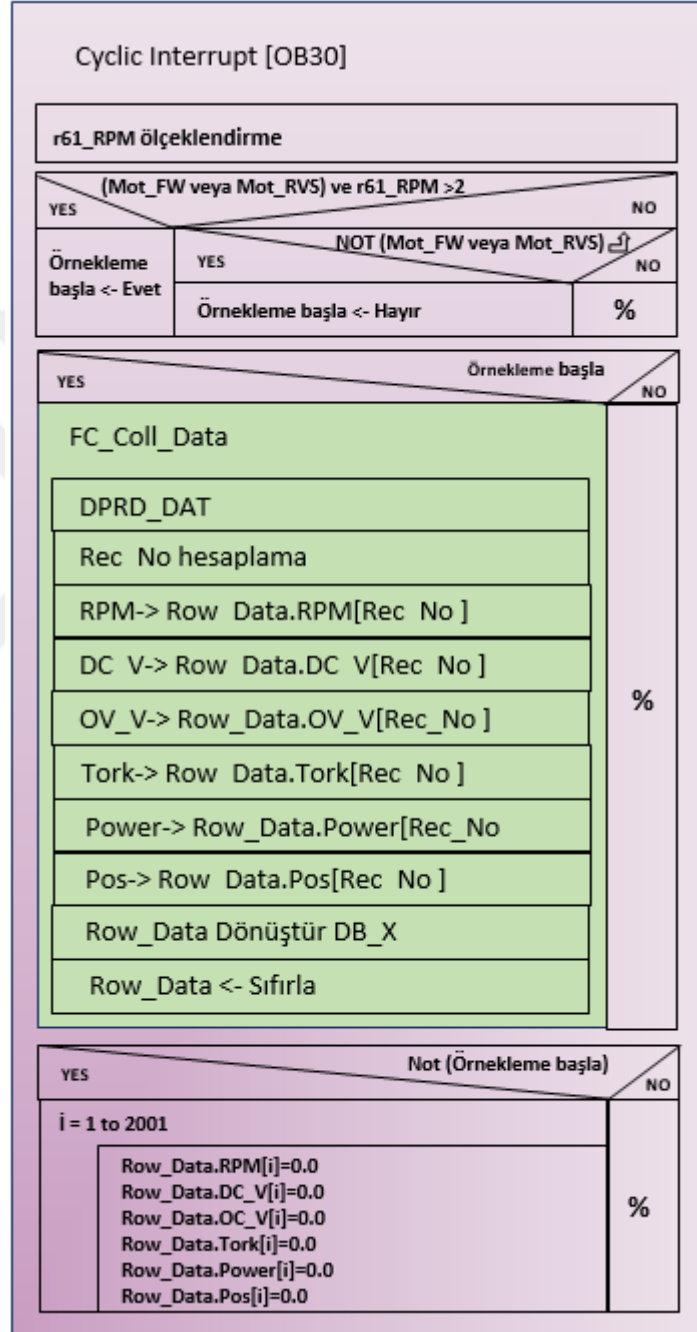


Şekil 2.14 Ana program stuckogramı

Programın son parçası olan Servo sistemden anlık ve çapraz olarak verilerin okunmasını, bunların bir data blokta istiflenmesini ve öznitelik hesaplama fonksiyonlarına girdi olacak şekilde tek bir dizi elemanına dönüştürülmesini içeren bölüm yer almaktadır. Bu bölüm okunacak verinin aynı zaman aralıkları ile okunmasını sağlamak için döngüsel kesme programının içerisine yazılmıştır. Bu sayede veriler eş zaman aralıkları ile okunarak data blokların içerisine istiflenmesi sağlanmıştır. Sürücünden okunan verilerin çapraz olarak eş zamanlı okunmasını sağlamak içinde ProfiNet’ protokolünde bir telegramın bölünmeden okunup yazılmasını sağlayan tümleşik veri transfer bloğu olan “DPRD_DAT”

kullanılmaktadır. Bu blok üretici tarafından gelişmiş haberleşme kütüphanesinde sunulmaktadır.

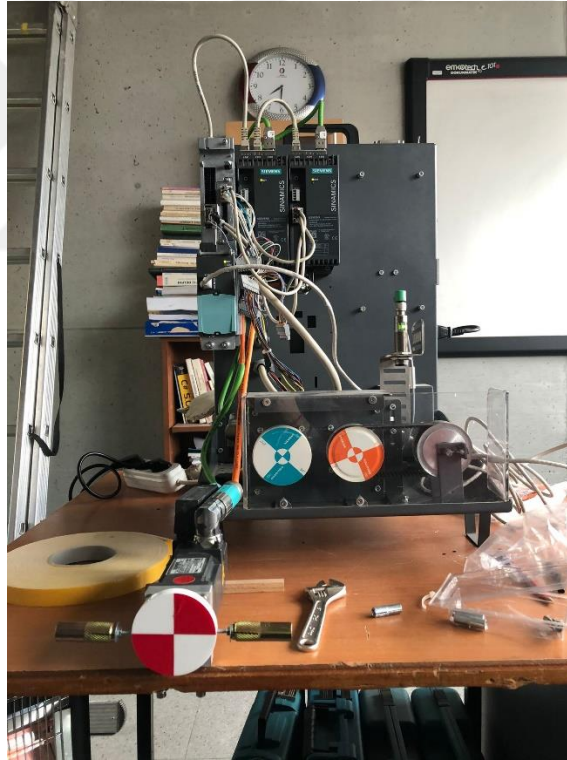
Haberleşme telegramından elde edilen ham veriler öznitelik bloğuna atanmadan önce ölçeklenerek normalize edilir ardından da tek bir dizinin içerisine kopyalanmaktadır.



Şekil 2.15 Kesme programı stucktogramı

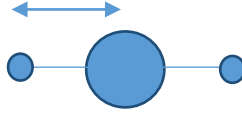
3. DENEY DÜZENEGİ

Öneride bulunulan YSA'yı eğitmek için farklı yük karakteristikleri uygulanabildiği bir deney düzeneği tasarlanmıştır. Bunun için motor miline eklediğimiz bir kompozit çarkın üzerine yük ekleyebileceğimiz iki vida eklenmiş bu vidaların üzerine de farklı ağırlıkta ağırlıklar yerleştirilerek deneyler gerçekleştirilmiştir. Gerçekleştirilen deneyler sırası ile motorun tamamen yüksüz olduğu durum ile başlanmıştır. Motor bu şartlarda tune işlemi tamamlanmıştır. Bu işlemin ardından aynı fiziksel şartlar ile önceden belirlemiş olduğumuz zamana bağlı sinüzoidal bir hız-pozisyon fonksiyonu uygulanarak 3.3 saniye uzunluğunda bir veri kaydı gerçekleştirilmiştir. Bu veri kaydında YSA'yı eğitmek için kullanılacak 8 değer, 100 milisaniye zaman aralıkları ile çapraz bir şekilde örneklenmesi sağlanmıştır. Bu adımın ardından sırası ile hız ve akım PID değerleri değiştirilerek ilgili deney tekrarlanmıştır.



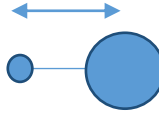
Şekil 3.1 Eğitim Deney düzeneği

Ardından deneyler şekil 3.2 de görüldüğü üzere 7cm uzunlukta ve 45.75 gr dengeli bir yük yerleşimi ile devam edilmiştir. Aynı deney prosedürü yeni fiziksel şartlar için uygulanmış ve tekrarlanmıştır.



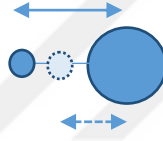
Şekil 3.2 Dengeli yük

Bir sonraki adımda 7cm uzunlukta ve 45.75 gr olarak 0.0314Nm tork oluşturmak için tek bir ağırlık takılarak dengesiz bir yük oluşturulmuştur ve aynı deney prosedürleri tekrarlanarak ilgili örnekler alınmıştır.



Şekil 3.3 Dengesiz yük

Son olarak momentumu değişken bir momentum oluşturmak adına 45.75 gr yük 7cm ile 5cm arasında hareketli olacak şekilde yerleştirilmiş ve deney prosedürleri tekrarlanarak bu durum içinde YSA eğitilecek datalar toplanarak kaydedilmiştir.

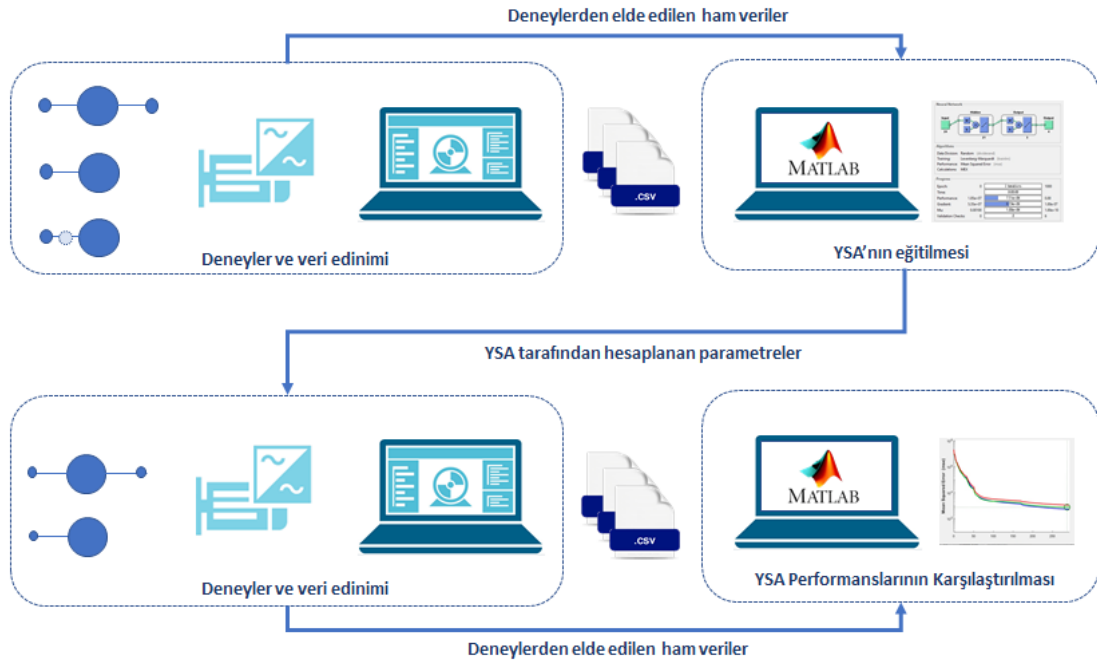


Şekil 3.4 Dengesiz yük değişken

Ayrıca YSA'yı daha iyi besleyebilmek adına deneylerde elde edilen tune değerleri farklı fiziksel deney düzeneklerinde test edilerek çapraz şartlar içinde eğitim verileri elde edilmiştir.

İlgili verilerin elde edilmesinin ardından bu veriler YSA'yı eğitmek için kullanılmıştır.

Aşağıda belirtilen akış şemasında da görülebileceği üzere YSA'nın eğitilmesi esnasında testler için bağımsız bir deney verisi oluşturmak için 45.75 gr olan deney ağırlığı 14.5 grama düşürülerek hem dengeli yük için iki ağırlık takılarak hem de dengesiz yük için tek ağırlık takılarak deneyler tekrarlanmıştır. YSA'dan elde edilen değerler bu değerler ile karşılaştırılarak YSA'nın performansına karar verilmiş ve en uygun bulunan 3 YSA tasarımı için deneyler ağırlığı azaltılmış gerçek deney düzeneğinde test edilerek en uygun YSA tasarımına karar verilmiştir. Bu esnada motorun çekiş akımı ile hız verilerindeki takip hatası kullanılmış ve en az hata veren YSA endüstriyel kontrolcünün içerisine gömülme için tercih edilmiştir.



Şekil 3.5 YSA eğitim ve seçim akış şeması

Önerilen sistemin test edilmesi için son olarak deney düzeneğimizde servo sistem PLC üzerinden kontrol edilebilmesi için ProfiNET IRT ile haberleştirilecek hem PLC'den Servo sisteme komut gönderilmesi sağlanmış hem de YSA'nın karar almasında kullanacağı anlık tork vs değerleri okunarak yine YSA'nın ürettiği sonuçları ilgili parametrelere yazılmasını sağlayacak 2.3.3 bölümde açıklanan programın yüklenerek deney düzeneği tekrar düzenlenmiştir.

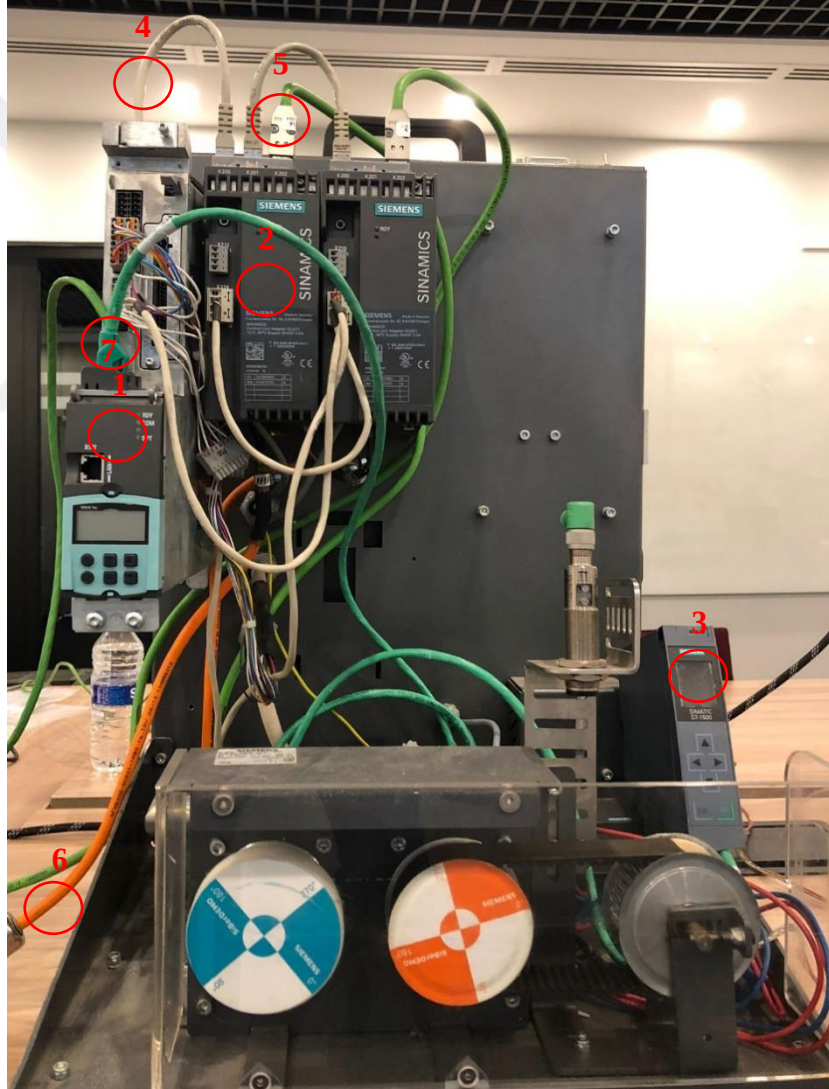


Şekil 3.6 Deney düzeneği bileşenleri

Yeni düzenlenmiş deney setinde YSA eğitimlerinde kullanılmayan 14.5 gr ağırlıklı 5 cm uzunlukta dengeli yüklü bir fiziksel ortam hazırlanarak tam deney düzeneğinde örnekler

alınarak daha önce yapılan deney sonuçları işle bu çalışmanın bir sonraki sonuç kısmında karşılaştırılmıştır.

Aşağıdaki resimde görünen son deney düzeneğinde 1 numara Siemens S120 servo sürücü kontrol ünitesi 2 numara Siemens Sinamics servo sürücü güç ünitesini, 3 Siemens S7-1500 PLC, 4 numara Servo kontrol ünitesi ile güç ünitesi arasında haberleşmenin sağlandığı drive cliq haberleşme bağlantısı, 5 numara motor ile sürücü arasındaki drive cliq haberleşme kablosu, 6 numara motor güç kablosu ve son olarak da 7 numara ile gösterilen bağlantı servo sürücü kontrol ünitesi ile PLC arasındaki ProfiNet haberleşme bağlantısı gösterilmektedir.

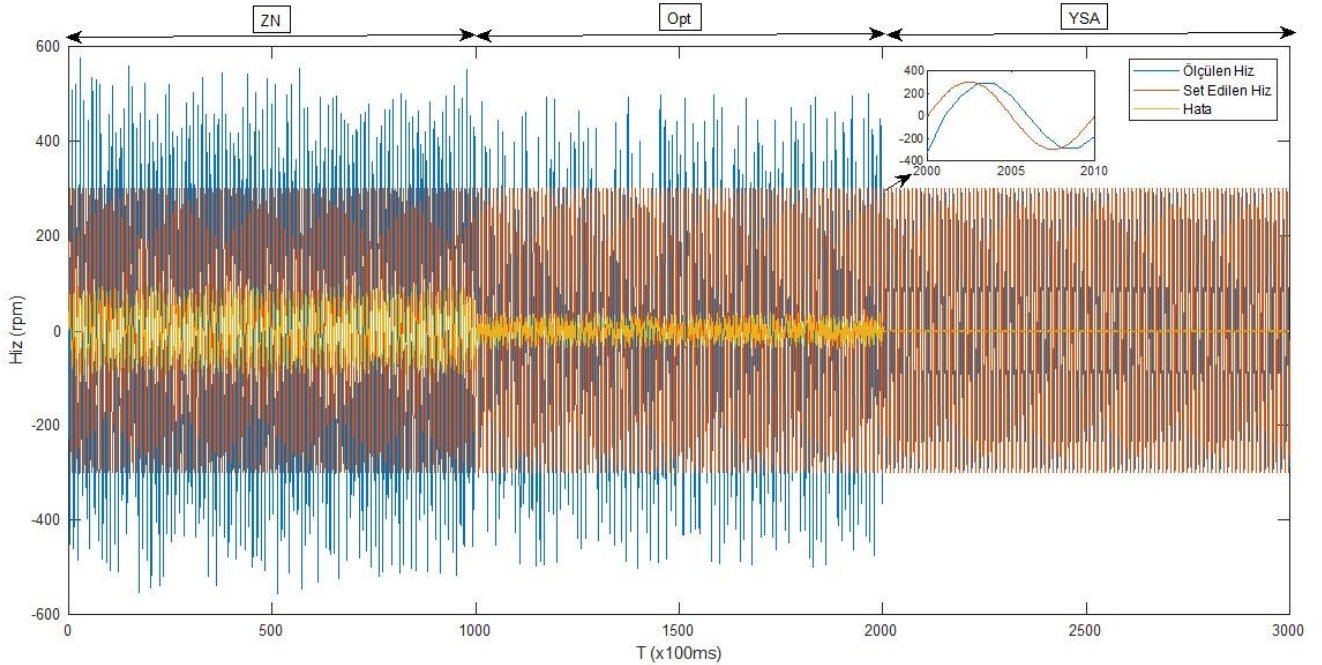


Şekil 3.7 Çalışmada kullanılan deney düzeneği

4. SONUÇ

Kurmuş olduğumuz son deney düzeneği ile yapmış olduğumuz deneyler ve sonuçlarını karşılaştırmak için endüstride sıklıkla kullanılan Z-N tunning teknikleri ile deneyler tekrarlanmıştır. İlk olarak belirtmek gerekir ki Z-N tekniği deneysel bir teknik olup gözleme dayalı veri elde etmek ile başlar, ardından da optimizasyon için sistemi devreye alan kişinin bilgi ve yetenekleri ile sınırlıdır. Bu yüzden Z-N işleminin ardından online olarak akım ve hız değerleri izlenmiş ve manuel optimizasyon teknikleri kullanılarak sürücü tekrar ayarlanmış ve deney tekrarlanmıştır. Bu sayede Şekil 4.1’de görüldüğü gibi verileri karşılaştırabileceğimiz üç farklı sonuç elde edilmiştir.

İlk olarak hız kontrol için verilen set değeri ile elde edilen geri besleme verisi ve üç sistem içinde oluşan mean squared error (MSE) verilerini Şekil 4.1’de kontrol ettiğimizde görebiliyoruz ki YSA tekniği ile elde edilen değerlerdeki takip hataları çok daha azdır. Buda göstermektedir ki YSA ile hesaplanan bu değerler ve bunların sürekli kontrolü ve düzenlenmesi ile daha iyi sonuçlar devreye alan kişinin yeteneklerinden tamamen bağımsız olarak elde edilebilmektedir.

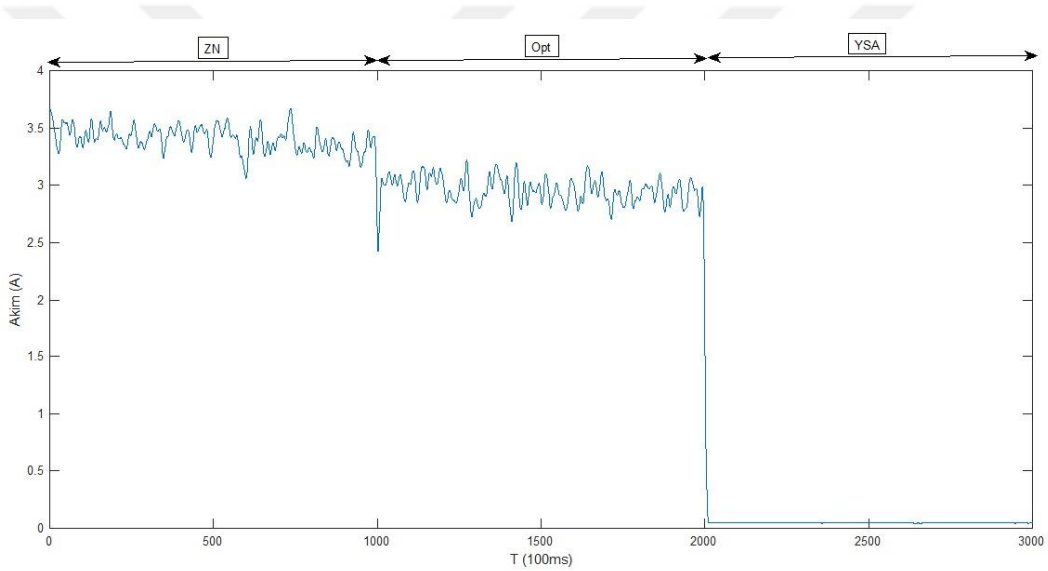


Şekil 4.1 Karşılaştırmalı Hız-Zaman grafiği

Tablo 4.1 Hız MSE karşılaştırma tablosu

	Z-N	Optimize Z-N	YSA
MSE Hız	171978	23549	2010

Bir diğer kontrol datamız da motorların çekmiş olduğu akımlardır. Şekil 4.2’de görebildiğimiz gibi aynı hareket karakteristiğinde çok daha az ve kararlı şekilde akım ihtiyacı duymuştur. Buda YSA ile hesaplanan değerlerde kontrol edilen motorların daha verimli çalıştığını ilgili deney sonuçlarına bakarak söyleyebiliriz.



Şekil 4.2 Karşılaştırmalı Akım-Zaman grafiği

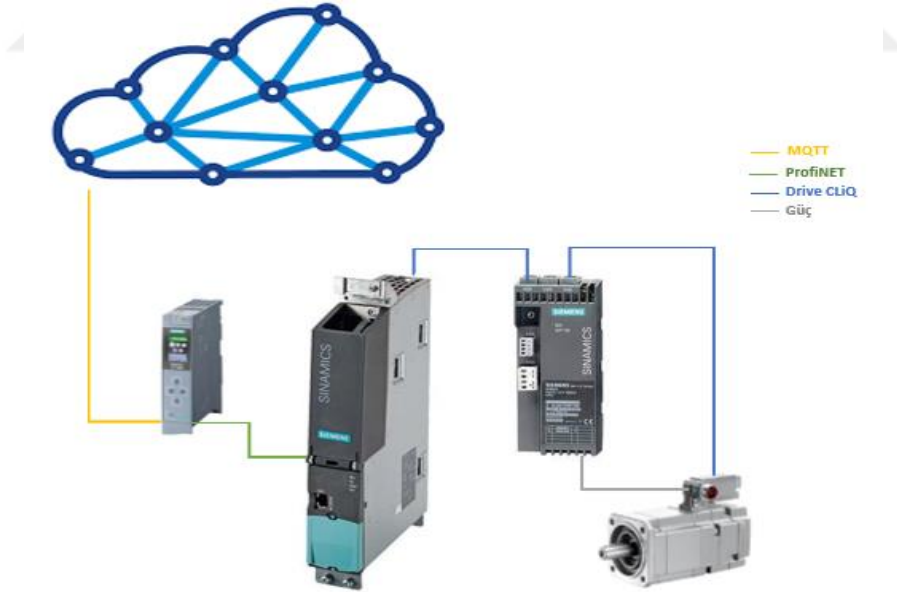
Tüm bu verilerin ışığında servo sistemlerin PID katsayılarının hesaplanmasında YSA tekniğinin de kullanılabileceği ve endüstride hali hazırda sıklıkla kullanılan PLC olarak adlandırdığımız kontrol objelerinde bu YSA algoritmalarını belirli limitler dahilinde başarılı bir şekilde işletebildiğini söyleyebiliriz.

Ayrıca bu sistemin en önemli özelliklerinden bir tanesi de her 3.33 sn örnekleme aralıkları ile sistemden veri edinerek gözlemlediği yeni duruma göre tekrar ilgili parametreleri YSA algoritmamızın belirlediği yeni değer ile güncellenmesidir. Bu da sistemin adaptif / uyarlanabilir bir yapıda olduğunu ve özellikle güç ve tork talebinin zamanla değişebildiği sistemlerde optimum kontrol parametrelerinin önerilen akıllı yöntem ile belirlenebileceği görülmüştür.

5. GELECEK ÇALIŞMALAR

Deney düzeneği olarak tercih edilmiş yapı endüstride kullanılan sistemlerin bir benzeri olması sayesinde bu teknik ek bir donanım gerektirmeksizin hali hazırda kullanılan pek çok makineden veri toplama işlemini yapabilecektir. Elde edilen verilerin büyüklüğü ve farklı lokasyonlardan veri alınmasını sağlamak için yeni verileri “*message queuing telemetry transport*” (MQTT) gibi güvenli bir veri toplama [32] veya muadili bir protokol kullanılarak bir bulut sunucuya aktarılması sağlanabilir (Şekil 5.1). Ayrıca bu ortamda öğrenme işlemi sürekli ve tekrarlı hale getirilerek YSA’nın zamanla değişen parametrelere uyum sağlayarak daha başarılı sonuçlar üretmesini sağlayabilir [33]. Yeni YSA verileri de tekrardan ilgili servo sistemlere geri yüklenerek makine performansları daha da iyileştirilebilir.

Tezimizde veri seti oluşturmak için tek bir servo sistem kullanılmıştır fakat bu öneri ile farklı büyüklükte ve farklı markalarda da öğrenme işlemi yapılabilir ve daha genel bir YSA modeli oluşturulması sağlanabilir.



Şekil 5.1 Gelecek çalışmalar için hedeflenen deney düzeneği

KAYNAKLAR/REFERENCES

- [1] Jaen-Cuellar, A. Y., de J. Romero-Troncoso, R., Morales-Velazquez, L., & Osornio-Rios, R. A. (2013). PID-controller tuning optimization with genetic algorithms in servo systems. *International Journal of Advanced Robotic Systems*, 10(9), 324.
- [2] Chen, C. W., Chang, L. K., Liao, Y. T., Chung, C. H., Su, W. C., Chen, K. S., & Tsai, M. C. (2020, November). Tuning of Servo Drive Controller Based on Boosted Tree Model and Particle Swarm Optimization. In 2020 23rd International Conference on Electrical.
- [3] Lin, Q. S., Yao, Y. F., & Wang, J. X. (2010, November). Simulation and application of neural network PID auto-tuning controller in servo-system. In 2010 2nd International Workshop on Database Technology and Applications (pp. 1-4). IEEE.
- [4] Lin, C. M., & Hsu, C. F. (2002). Neural-network-based adaptive control for induction servomotor drive system. *IEEE Transactions on Industrial Electronics*, 49(1), 115-123.
- [5] Firoozian, R. (2014). *Servo motors and industrial control theory*. Springer.
- [6] PID Theory Explained, 17 Mart 2020. [Çevrimiçi]. Available: <https://www.ni.com/en-il/innovations/white-papers/06/pid-theory-explained.html>. [Erişildi: 2021].
- [7] Bhinge, R., Biswas, N., Dornfeld, D., Park, J., Law, K. H., Helu, M., & Rachuri, S. (2014, October). An intelligent machine monitoring system for energy prediction using a Gaussian Process regression. In 2014 IEEE International Conference on Big Data (Big.
- [8] Zhou, Y., Mei, X., Zhang, Y., Jiang, G., & Sun, N. (2009, May). Current-based feed axis condition monitoring and fault diagnosis. In 2009 4th IEEE Conference on

Industrial Electronics and Applications (pp. 1191-1195). IEEE.

- [9] Kande, M., Isaksson, A. J., Thottappillil, R., & Taylor, N. (2017). Rotating electrical machine condition monitoring automation—A review. *Machines*, 5(4), 24.
- [10] Dinardo, G., Fabbiano, L., & Vacca, G. (2018). A smart and intuitive machine condition monitoring in the Industry 4.0 scenario. *Measurement*, 126, 1-12.
- [11] Jingjing, X., & Jiaoyu, L. (2013, May). Neural network PID controller auto-tuning design and application. In 2013 25th Chinese Control and Decision Conference (CCDC) (pp. 1370-1375). IEEE.
- [12] Aftab, M. S., & Shafiq, M. (2015, February). Adaptive PID controller based on Lyapunov function neural network for time delay temperature control. In 2015 IEEE 8th GCC Conference & Exhibition (pp. 1-6). IEEE.
- [13] Kumar, S., Mukherjee, D., Guchhait, P. K., Banerjee, R., Srivastava, A. K., Vishwakarma, D. N., & Saket, R. K. (2019). A comprehensive review of condition based prognostic maintenance (CBPM) for induction motor. *Ieee Access*, 7, 90690-90704.
- [14] Samhour, M., Al-Ghandour, A., Ali, S. A., Hinti, I., & Massad, W. (2009). An intelligent machine condition monitoring system using time-based analysis: neuro-fuzzy versus neural network. *Jordan Journal of Mechanical and Industrial Engineering*, 3(4), 294.
- [15] Watson, G. A. (Ed.). (2006). Numerical analysis: proceedings of the Biennial Conference held at Dundee, June 28-July 1, 1977 (Vol. 630). Springer..
- [16] Duymazlar, O., Engin, M., & Engin, D. (2020, June). Embedded Artificial Neural Network on PLCs to Predict Nonlinear System Responses. In 2020 9th Mediterranean Conference on Embedded Computing (MECO) (pp. 1-4). IEEE.
- [17] Jung, I. S., Mulman, B. M., Thapa, D., Koo, L. J., Bae, J. H., Hong, S. H., ... & Wang, G. N. (2008, October). PLC control logic error monitoring and prediction using

Neural Network. In 2008 Fourth International Conference on Natural Computation (Vol. 2).

- [18] Canedo, A., Ludwig, H., & Al Faruque, M. A. (2014). High communication throughput and low scan cycle time with multi/many-core programmable logic controllers. *IEEE Embedded Systems Letters*, 6(2), 21-24.
- [19] Li, J., & Gómez-Espinosa, A. (2018, November). Improving PID control based on neural network. In 2018 International Conference on Mechatronics, Electronics and Automotive Engineering (ICMEAE) (pp. 186-191). IEEE.
- [20] Dias, A. L., Sestito, G. S., & Brandao, D. (2017). Performance analysis of profibus dp and profinet in a motion control application. *Journal of Control, Automation and Electrical Systems*, 28(1), 86-93.
- [21] Fontanelli, D., Macii, D., Rinaldi, S., Ferrari, P., & Flammini, A. (2013, May). Performance analysis of a clock state estimator for PROFINET IO IRT synchronization. In 2013 IEEE International Instrumentation and Measurement Technology Conference (I2MTC).
- [22] Wu, X., Xie, L., & Lim, F. (2014, October). Network delay analysis of EtherCAT and PROFINET IRT protocols. In *IECON 2014-40th Annual Conference of the IEEE Industrial Electronics Society* (pp. 2597-2603). IEEE..
- [23] Liu, L., Shan, L., Yan, J., Liu, C., & Dai, Y. (2018, June). An improved BFO algorithm for optimising the PID parameters of servo system. In 2018 Chinese Control And Decision Conference (CCDC) (pp. 3831-3836). IEEE.
- [24] Alan, M. (2020)Derin öğrenme ile biyosinyal sınıflandırma ve hastalık tahmini. Yüksek Lisans Tezi, Marmara Üniversitesi Fen Bilimleri Enstitüsü, İstanbul, Türkiye.
- [25] Haykin, S. S. (2009). *Neural networks and learning machines*/Simon Haykin.
- [26] OKKAN, U., SERBEŞ, Z. A., & GEDİK, N. (2018). MATLAB ile Levenberg-Marquardt algoritması tabanlı YSA uygulaması: Aylık yağış-akış modellemesi. *Dicle*

Üniversitesi Mühendislik Fakültesi Mühendislik Dergisi, 9(1), 351-362.

- [27] Çavuşlu, M. A., Becerikli, Y., & Karakuzu, C. (2012). Levenberg-Marquardt algoritması ile YSA eğitiminin donanımsal gerçekleştirilmesi. Türkiye Bilişim Vakfı Bilgisayar Bilimleri ve Mühendisliği Dergisi, 5(1).
- [28] Akgün, G. (2015) El rehabilitasyonu için EMG kontrollü giyilebilir iskelet destek sistemi tasarımı. Yüksek Lisans Tezi, Marmara Üniversitesi Fen Bilimleri Enstitüsü, İstanbul, Türkiye.
- [29] Pencere-Fonksiyonlari, [Çevrimiçi]. Available: <http://www.atasoyweb.net/Spektral-Sizinti-Ve-Pencere-Fonksiyonlari>. [Erişildi: 7 Ağut 2021].
- [30] Neural Net Fitting, Mathworks, [Çevrimiçi]. Available: <https://www.mathworks.com/help/deeplearning/ref/neuralnetfitting-app.html>.
- [31] Hering, E. (1989). Struktogramme nach Nassi/Shneiderman (DIN 66 261) und Pseudocode. In Software-Engineering (pp. 30-40). Vieweg+ Teubner Verlag, Wiesbaden.
- [32] Pampapathi, B. M., Guptha, M. N., & Hema, M. S. (2021). Data distribution and secure data transmission using IANFIS and MECC in IoT. Journal of Ambient Intelligence and Humanized Computing, 1-14.
- [33] Xue, S., Xiong, L., Yang, S., & Zhao, L. (2016). A self-adaptive multi-view framework for multi-source information service in cloud ITS. Journal of Ambient Intelligence and Humanized Computing, 7(2), 205-220.

ÖZGEÇMİŞ

KİŞİSEL BİLGİLER

Hayri MUTLU

EĞİTİM BİLGİLERİ

2018 - 2019	Marmara Üniversitesi Teknoloji Fakültesi, İstanbul Lisans, Mekatronik Mühendisliği ABD
2003 - 2007	Marmara Üniversitesi Teknik Eğitim Fakültesi, İstanbul Lisans, Mekatronik Eğitimi, Mekatronik Öğretmenliği ABD
1999 – 2003	Tekirdağ Anadolu Meslek Lisesi (Elektronik Bölümü)

İŞ TECRÜBESİ

2016 -	Siemens San. Tic. A.Ş. Dijital Endüstriler Departmanı, Teknik Eğitimci
2012 - 2016	Kon-Tek Otomasyon A.Ş. Pazarlama Departmanı, Otomasyon Ürün Yöneticisi
2011 - 2012	Wartsila Empa A.Ş. Elektrik Üretim Gurubu, Otomasyon Proje Mühendisi
2007 - 2011	Kon-Tek Otomasyon A.Ş. Proje Departmanı, Otomasyon Proje Mühendisi

