ÇUKUROVA UNIVERSITY
INSTITUTE OF NATURAL AND APPLIED SCIENCES

MSc THESIS

Muhammed Mahmut AKSOY

FUZZY ITERATIVE LEARNING CONTROL WITH APPLICATION

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

ADANA-2021

ÇUKUROVA UNIVERSITY
INSTITUTE OF NATURAL AND APPLIED SCIENCES

FUZZY ITERATIVE LEARNING CONTROL WITH APPLICATION

Muhammed Mahmut AKSOY

Msc THESIS

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

We certify that the thesis titled above was reviewed and approved for the award of degree of the Master of Science by the board of jury on 18/01/2021.

………………………	………………………	……………………………………
Prof. Dr. İlyas EKER	Prof. Dr. Ahmet TEKE	Asst. Prof. Dr. Necdet Sinan ÖZBEK
SUPERVISOR	MEMBER	MEMBER

This Msc Thesis is written at the Department of Institute of Natural and Applied Sciences of Çukurova University.
Registration Number:

Prof. Dr. Mustafa GÖK
Director
Institute of Natural and Applied Sciences

This study was supported by Çukurova University Scientific Research Project Unit. Project No: FYL-2019-11773

ABSTRACT

MSc THESIS

FUZZY ITERATIVE LEARNING CONTROL WITH APPLICATION

Muhammed Mahmut AKSOY

ÇUKUROVA UNIVERSITY
INSTITUTE OF NATURAL AND APPLIED SCIENCED
DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

Supervisor : Prof. Dr. Ilyas EKER
Year: 2021 Page: 123
Jury : Prof. Dr. Ilyas EKER
: Prof. Dr. Ahmet TEKE
: Asst. Prof. Dr. Necdet Sinan ÖZBEK

Control applications have become important for accuracy and high precision in industrial systems with the developing technology. By eliminating the deficiencies of classical controllers with modern control methods, it has provided the systems to work in desired performance specifications. In this study, a fuzzy PID-type iterative learning control method was developed and a real-time experimental application of DC motor speed control was made. Alternatively, adaptive fuzzy PID-type iterative learning control has been developed. The methods are created by combining the adaptive method, fuzzy logic control and iterative learning control. The proportional, integral and derivative (K_P, K_I, K_D) gains of the PID controller are adjusted according to fuzzy logic. The fuzzy logic controller is developed according to fuzzy rules, thus ensuring the system is fundamentally robust. Fuzzy rules are used to adjust each PID parameter. Adaptive method is used for adaptation of fuzzy logic control input and one of PID parameters. There are two adaptive algorithms in this system. The algorithms helped to adjust PID parameters. In the iterative learning controller (ILC) part, a new control signal is generated by using the PID parameters generated from the fuzzy logic controller and the previously generated control signal. With this method applied on the DC motor, the transient response, tracking response and disturbance reduction situations were examined. The results show that adaptive fuzzy PID-type ILC method has better time domain characteristics and gives better DC motor performance.

Keywords: Iterative Learning Control, ILC, Fuzzy Logic, Adaptive, PID

ÖZ

YÜKSEK LİSANS TEZİ

BULANIK YİNELEMELİ ÖĞRENME KONTROLÜ İLE UYGULAMASI

Muhammed Mahmut AKSOY

ÇUKUROVA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
ELEKTRİK ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI

Danışman : Prof. Dr. Ilyas EKER
Year: 2021 Page: 123
Jüri : Prof. Dr. Ilyas EKER
: Prof. Dr. Ahmet TEKE
: Dr. Öğr. Üyesi Necdet Sinan ÖZBEK

Gelişen teknoloji ile birlikte endüstriyel sistemlerde doğruluk ve yüksek hassasiyet için kontrol uygulamaları önem kazanmıştır. Modern kontrol yöntemleriyle klasik kontrolörlerin eksikliklerini ortadan kaldırarak, sistemlerin istenilen performans özelliklerinde çalışmasını sağlamıştır. Bu çalışmada, bulanık PID tipi yinelemeli öğrenme kontrol yöntemi geliştirilmiş ve DC motor hız kontrolünün gerçek zamanlı deneysel bir uygulaması yapılmıştır. Alternatif olarak, uyarlanabilir bulanık PID tipi yinelemeli öğrenme kontrolü geliştirilmiştir. Yöntemler, uyarlamalı yöntem, bulanık mantık kontrolü ve yinelemeli öğrenme kontrolü birleştirilerek oluşturulur. PID kontrolörünün orantılı, integral ve türev (K_P, K_I, K_D) kazançları bulanık mantığa göre ayarlanır. Bulanık mantık denetleyicisi, bulanık kurallara göre geliştirilmiştir, böylece sistemin temelde sağlam olmasını sağlar. Her PID parametresini ayarlamak için bulanık kurallar kullanılır. Uyarlanabilir yöntem, bulanık mantık kontrol girişi ve PID parametrelerinden birinin uyarlanması için kullanılır. Bu sistemde iki uyarlanabilir algoritma vardır. Algoritmalar, PID parametrelerinin ayarlanmasına yardımcı oldu. Yinelemeli öğrenme denetleyicisi (ILC) bölümünde, bulanık mantık denetleyicisinden üretilen PID parametreleri ve önceden oluşturulmuş kontrol sinyali kullanılarak yeni bir kontrol sinyali üretilir. DC motora uygulanan bu yöntem ile geçici tepki, izleme tepkisi ve bozulma azaltma durumları incelenmiştir. Sonuçlar, uyarlanabilir bulanık PID tipi ILC yönteminin daha iyi zaman uzayı karakteristikleri ve daha iyi DC motor performansı sağladığını göstermektedir.

Anahtar Kelimeler: Yinelemeli Öğrenme Kontrolü, ILC, Bulanık Mantık, Uyarlama, PID

EXTENDEN ABSTRACT

In parallel with the development of technology over time, the increase in the application areas of control theory has enabled the development of modern control methods. The technological development of control applications has caused the application in many different areas today. The usage areas of modern control methods are more common than in the past. Defense systems, aircraft, autonomous vehicles, manipulators, robotic systems, power systems can be given as examples of today's usage areas. In the present study, different modern control methods are emphasized.

Control applications are used to improve the performance of the system to which it is applied. However, there are situations that determine system performance. Examples of factors affecting performance are unmodelled dynamics, disturbance and delay. Modern control methods are developed within the scope of researches and studies to improve system performance. Among modern control methods, Iterative Learning Control (ILC), fuzzy logic control and adaptive control are important developments.

PID controllers have been widely used in the industry since the past. Various tunning methods have been developed over time to adjust the parameters of PID controllers. PID controllers are preferred in applications thanks to their simple structures, but with the developing technology, alternative methods have been developed to PID controllers. Thus, due to the disadvantages of PID controllers, there has been a trend towards more successful modern control methods.

In this thesis, different control methods have been developed and an alternative to PID controllers is presented. The proposed control methods are used in many different applications recently. These control structures are designed for the system to be applied and control system structures have been created by determining control algorithms. After completing the controller structures, they are subjected to different performance tests and their responses in these tests are examined. Attention

is drawn to the positive and negative aspects of the control methods applied in this way.

Iterative learning control is one of the modern control methods recommended. The Iterative learning control structure has been extensively examined and the statistics of its use in which areas and its positive effects in these areas have been investigated. The way the Iterative leatning control algorithm works is similar to human learning. For both, the basic way to learn is to repeat. It tries to find a better solution with each repetition by learning the system in which iterative learning control is applied with repetitive work tasks. The memory block in the ILC control structure plays a role in generating the next control input signal by keeping the control input data of previous trials of the system. The closed-loop PID-type iterative learning control structure has been developed instead of the classical PID controller. The properties of the iterative learning control algorithm have been examined and its advantages and innovations against classical controllers have been explained. By making real-time experimental applications, examples of PID-type iterative learning control application to contribute to the literature are given. The closed-loop, tracking test and load test performances are satisfactory in these real-time experiments.

Fuzzy logic control is another proposed modern control method. The concept of fuzzy logic has been developed for problems that classical Aristotelian logic cannot find solutions. Fuzzy logic has brought a new perspective for solving problems. By using the fuzzy logic structure in control applications, a new alternative control method to classical PID controllers has been developed. With the development of the fuzzy logic control method, successful applications have been realized in many different areas. The information about the application areas of the fuzzy logic control structure has been compiled by reviewing the literature. In this thesis, the design and applications of the fuzzy logic control structure are explained. A fuzzy logic control structure has been developed to calculate PID parameters. The fuzzy PID logic control scheme, which successfully calculates PID parameters, has been developed. How the fuzzy logic controller changes the values of PID parameters against changing situations in practice is examined. Fuzzy rules are determined to make the calculations of PID parameters. Real-time implementation of the Fuzzy PID logic control structure is aimed to provide another example to the literature on this subject. The closed-loop system, tracking test and load test performances have been examined in real time applications and these performance values are satisfactory.

The iterative learning control and fuzzy logic control structures are combined in a single control structure. A new control scheme has been created by combining these two control structures. By combining these two control methods, it is aimed to improve performance values. PID parameter values are calculated with the fuzzy logic control structure and these parameters required for iterative learning control are provided from the fuzzy logic controller. In the iterative learning control section, a new control input is created by using the PID parameters from the fuzzy logic controller and the previous control input. It is thought to contribute to the literature by creating this control structure. With the development of the Fuzzy PID-type iterative learning control structure, it has become an alternative to classical PID controllers. The performances of the fuzzy PID-type iterative learning controller have been investigated by real-time applications. Performance results from real time applications are satisfactory.

A further contribution of the present thesis is the use of adaptive algorithms instead of scaling factors used for inputs of the fuzzy controller structure in the fuzzy PID-type iterative learning control structure. The use of adaptive algorithms has completed the adaptive fuzzy PID-type iterative learning control structure. Thus, a new control method was developed. The purpose of using adaptive algorithms is to adapt the fuzzy logic control inputs and outputs to the system. It is aimed to increase the performance of the developed control structure in applications by improving the adaptation of inputs and outputs of the fuzzy controller in the control system. The performance of the adaptive fuzzy PID-type iterative learning control system was investigated with real-time experiments.

Real-time applications of different control structures have been made for the thesis. Step test, tracking test and load test performances of control systems in real time applications were examined. Comparisons of the developed control methods in the thesis have been made. As a result of comparisons, control methods that can be used instead of classical PID controllers are proposed. The characteristic features of the proposed control methods are an important contribution to the thesis.

GENİŞLETİLMİŞ ÖZET

Zamanla teknolojinin gelişmesine paralel olarak kontrol teorisinin uygulama alanlarının artması, modern kontrol yöntemlerinin gelişmesini sağlamıştır. Kontrol uygulamalarının teknolojik gelişimi, günümüzde birçok farklı alanda uygulamaya neden olmuştur. Modern kontrol yöntemlerinin kullanım alanları geçmişe göre daha yaygındır. Savunma sistemleri, uçaklar, otonom araçlar, manipülatörler, robotik sistemler, güç sistemleri günümüz kullanım alanlarına örnek olarak verilebilir. Bu çalışmada, farklı modern kontrol yöntemleri vurgulanmaktadır.

Kontrol uygulamaları, uygulandığı sistemin performansını artırmak için kullanılır. Ancak, sistem performansını belirleyen durumlar vardır. Performansı etkileyen faktörlerin örnekleri, modellenmemiş dinamikler, rahatsızlık ve gecikmedir. Sistem performansının iyileştirilmesine yönelik araştırmalar ve çalışmalar kapsamında modern kontrol yöntemleri geliştirilmektedir. Modern kontrol yöntemleri arasında, Yinelemeli Öğrenme Kontrolü (ILC), bulanık mantık kontrolü ve uyarlamalı kontrol önemli gelişmelerdir

PID kontrolörleri geçmişten beri endüstride yaygın olarak kullanılmaktadır. PID kontrolörlerinin parametrelerini ayarlamak için zaman içinde çeşitli ayarlama yöntemleri geliştirilmiştir. Basit yapıları sayesinde uygulamalarda tercih edilmekte olan PID kontrolörler gelişen teknoloji ile birlikte PID kontrolörlere alternatif yöntemler geliştirilmiştir. Bu nedenle, PID kontrolörlerinin dezavantajları nedeniyle, daha başarılı modern kontrol yöntemlerine doğru bir eğilim olmuştur.

Bu tezde farklı kontrol yöntemleri geliştirilmiş ve PID kontrolörlere bir alternatif sunulmuştur. Önerilen kontrol yöntemleri son zamanlarda birçok farklı uygulamada kullanılmaktadır. Bu kontrol yapıları uygulanacak sistem için tasarlanmış ve kontrol algoritmaları belirlenerek kontrol sistemi yapıları oluşturulmuştur. Kontrolör yapıları tamamlandıktan sonra farklı performans testlerine tabi tutulur ve bu testlerdeki tepkileri incelenir. Bu şekilde uygulanan kontrol yöntemlerinin olumlu ve olumsuz yönlerine dikkat çekilmektedir.

Yinelemeli öğrenme kontrolü, önerilen modern kontrol yöntemlerinden biridir. Yinelemeli öğrenme kontrol yapısı kapsamlı bir şekilde incelenmiş ve hangi alanlarda kullanımının istatistikleri ve bu alanlardaki olumlu etkileri araştırılmıştır. Yinelemeli öğrenme kontrol algoritmasının çalışma şekli, insan öğrenmesine benzer. Her ikisi için de öğrenmenin temel yolu tekrar etmektir. Tekrarlayan iş görevleri ile yinelemeli öğrenme kontrolünün uygulandığı sistemi öğrenerek her tekrarla daha iyi bir çözüm bulmaya çalışır. ILC kontrol yapısındaki bellek bloğu, sistemin önceki denemelerinin kontrol giriş verilerini saklayarak bir sonraki kontrol giriş sinyalinin üretilmesinde rol oynar. Klasik PID kontrolör yerine kapalı döngü PID tipi yinelemeli öğrenme kontrol yapısı geliştirilmiştir. Yinelemeli öğrenme kontrol algoritmasının özellikleri incelenmiş ve klasik kontrolörlere göre avantajları ve yenilikleri açıklanmıştır. Gerçek zamanlı deneysel uygulamalar yapılarak, literatüre katkı sağlayacak PID tipi yinelemeli öğrenme kontrol uygulamasına örnekler verilmiştir. Bu gerçek zamanlı deneylerde kapalı döngü, izleme testi ve yük testi performansları tatmin edicidir.

Bulanık mantık kontrolü, önerilen bir başka modern kontrol yöntemidir. Bulanık mantık kavramı, klasik Aristoteles mantığının çözüm bulamadığı sorunlar için geliştirilmiştir. Bulanık mantık, problemleri çözmek için yeni bir bakış açısı getirdi. Kontrol uygulamalarında bulanık mantık yapısı kullanılarak, klasik PID kontrolörlerine yeni bir alternatif kontrol yöntemi geliştirilmiştir. Bulanık mantık kontrol yönteminin gelişmesiyle birlikte birçok farklı alanda başarılı uygulamalar gerçekleştirilmiştir. Bulanık mantık kontrol yapısının uygulama alanlarına ilişkin bilgiler literatür taranarak derlenmiştir. Bu tezde bulanık mantık kontrol yapısının tasarımı ve uygulamaları anlatılmıştır. PID parametrelerini hesaplamak için bulanık bir mantık kontrol yapısı geliştirilmiştir. PID parametrelerini başarıyla hesaplayan bulanık PID mantık kontrol şeması geliştirilmiştir. Bulanık mantık denetleyicisinin PID parametrelerinin değerlerini pratikte değişen durumlara karşı nasıl değiştirdiği incelenmiştir. PID parametrelerinin hesaplamalarını yapmak için bulanık kurallar belirlenir. Bulanık PID mantık kontrol yapısının gerçek zamanlı uygulaması, bu konudaki literatüre başka bir örnek vermek amaçlanmıştır. Kapalı döngü sistemi, izleme testi ve yük testi performansları gerçek zamanlı uygulamalarda incelenmiş ve bu performans değerleri tatmin edicidir.

Yinelemeli öğrenme kontrolü ve bulanık mantık kontrol yapıları tek bir kontrol yapısında birleştirilmiştir. Bu iki kontrol yapısının birleştirilmesiyle yeni bir kontrol şeması oluşturulmuştur. Bu iki kontrol yönteminin birleştirilmesi ile performans değerlerinin iyileştirilmesi amaçlanmaktadır. PID parametre değerleri, bulanık mantık kontrol yapısı ile hesaplanır ve yinelemeli öğrenme kontrolü için gerekli olan bu parametreler bulanık mantık denetleyicisinden sağlanır. Yinelemeli öğrenme kontrolü bölümünde, bulanık mantık denetleyicisinden gelen PID parametreleri ve önceki kontrol girişi kullanılarak yeni bir kontrol girişi oluşturulur. Bu kontrol yapısını oluşturarak literatüre katkı sağlayacağı düşünülmektedir. Bulanık PID tipi yinelemeli öğrenme kontrol yapısının geliştirilmesi ile klasik PID kontrolörlerine bir alternatif haline gelmiştir. Bulanık PID tipi yinelemeli öğrenme denetleyicisinin performansları gerçek zamanlı uygulamalarla araştırılmıştır. Gerçek zamanlı uygulamalardan elde edilen performans sonuçları tatmin edicidir.

Bu tezin bir başka katkısı, bulanık PID tipi yinelemeli öğrenme kontrol yapısında bulanık denetleyici yapısının girdileri için kullanılan ölçeklendirme faktörleri yerine uyarlanabilir algoritmaların kullanılmasıdır. Uyarlanabilir algoritmaların kullanımı, uyarlanabilir bulanık PID tipi yinelemeli öğrenme kontrol yapısını tamamlamıştır. Böylece yeni bir kontrol yöntemi geliştirildi. Uyarlanabilir algoritmaları kullanmanın amacı, bulanık mantık kontrol giriş ve çıkışlarını sisteme uyarlamaktır. Bulanık denetleyicinin giriş ve çıkışlarının kontrol sistemine adaptasyonu iyileştirilerek uygulamalarda geliştirilen kontrol yapısının performansının artırılması hedeflenmektedir. Uyarlanabilir bulanık PID tipi yinelemeli öğrenme kontrol sisteminin performansı, gerçek zamanlı deneylerle araştırıldı.

Tez için farklı kontrol yapılarının gerçek zamanlı uygulamaları yapılmıştır. Gerçek zamanlı uygulamalarda kontrol sistemlerinin adım testi, izleme testi ve yük testi performansları incelenmiştir. Tezde geliştirilen kontrol yöntemlerinin karşılaştırmaları yapılmıştır. Karşılaştırmalar sonucunda klasik PID kontrolörleri yerine kullanılabilecek kontrol yöntemleri önerilmiştir. Önerilen kontrol yöntemlerinin karakteristik özellikleri teze önemli bir katkıdır.

ACKNOWLEDGEMENTS

I would like to thank my supervisor Prof. Dr. İlyas EKER for his valuable advice and guidance during this research and preparation of this thesis.

I would also like to thank Dr. Necdet Sinan ÖZBEK for helping to make necessary preparations for laboratory work.

I would like to thank my family for their endless support and encouragements.

I would like to thank the financial support of Scientific Research Project Unit of Çukurova University for my thesis (Project Number: FYL-2019-11773).

CONTENTS **PAGE**

LIST OF TABLES **PAGE**

LIST OF FIGURES **PAGE**

XXI

LIST OF ABBREVIATONS

AFILC	: Adaptive Fuzzy Iterative Learning Control
DC	: Direct Current
FILC	: Fuzzy Iterative Learning Control
FIS	: Fuzzy Inference System
FL	: Fuzzy Logic
FLC	: Fuzzy Logic Control
I/O	: Input Output
IC	: Input Combination
ILC	: Iterative Learning Control
MFs	: Membership Functions
PC	: Personal Computer
PID	: Proportional Integral Derivative
RPM	: Revolution Per Minute
V	: Voltage

1. INTRODUCTION

1.1. Introduction

Control applications have had a large place in the industry from past and present. Control applicaitons progress in parallel with the development of technology and the historical development of technology can be classified in four main point. These milestones are as follows:

- Observing and measuring time. This time period corresponds to between 300 BC and 1200 AD.
- The industrial revolution that took place in europe. The general perception is that the industrial revolution took place in the 18th century, but its roots go back to the 1600s.
- World War I, II and the start of mass communication. This is the period between 1910 and 1945.
- The era that developed with the beginning of the space and computer age in 1957.

Significant developments occurred in the period between the industrial revolution and the world war. Control theory began to be expressed in a mathematical language. The feedback control system was first properly analyzed mathematically in 1868 by J.C. Maxwell.

Control devices began to be used with the mechanization in the industrial revolution. The developments in this process showed the necessity of automatic control and control devices were invented and used for this purpose. These devices were such as speed regulator, flow regulator, pressure regulator, temperature regulator etc. Feedback control systems were made by trial and error and intuitively in the industrial revolution. In the mid-1800s, analysis of control systems was made with mathematics for the first time, and it was shown that mathematics is the language of control systems. The period before this development can be called prehistory of control theory.

It can be called the primitive automatic control period from the 1800's to the early 1900's. The period up to the 1960s can be said to be the classical automatic control period. The period that includes the present from the 1960s can be called the modern control period.

Feedback is used to overcome the shortcomings of open loop control systems. The purpose of use of feedback in closed loop controllers is to control the states and outputs of dynamic systems. The closed loop control diagram is shown in Figure 1.1.

In the classical control method period, it provides the regulation of system behaviors with feedback signals by examining system behaviors with inputs of dynamic systems. The input signal is adjusted to make the system output converge to the desired system signal. For this, a controller is designed to monitor the output signal and compare it with the reference. The difference between the actual output value of the system and the desired signal value is called error and this error signal is added to the system via feedback, so that the system output value is aimed to approach the reference value.

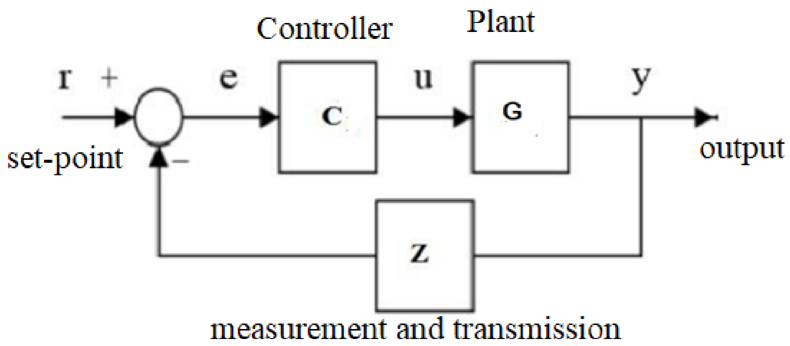


Figure 1.1. A Closed-loop Control System

The advantages of closed-loop controllers are as follows:

- Stabilizing the unstable systems
- Disturbance rejection
- Low sensitivity in parameter changes
- Have an improved tracking performance
- Ensuring guarantee performance in case of model parameters not being complete and model uncertainties

In closed-loop control system in Fig.1.1, it is seen that the system output *y* is connected to the reference *r* with feedback which contains the measurement sensor *z*. In the system, *C* and *G* are controller and plant respectively. These are called Single Input Single Output (SISO) control systems. If the number of inputs and outputs is more than one, it is called Multi Input Multi Output (MIMO).

It is considered that controller (*C*), plant (*G*) and measurement sensor (*Z*) are linear and time-invariant, so their mathematically analysis is in Laplace space as:

$$Y(s) = G(s)U(s) \tag{1}$$

$$U(s) = C(s)E(s) \tag{2}$$

$$E(s) = R(s) - Z(s)Y(s) \tag{3}$$

$$Y(s) = \frac{G(s)C(s)}{1+Z(s)G(s)}R(s) = H(s)R(s) \tag{4}$$

$$H(s) = \frac{G(s)C(s)}{1+Z(s)C(s)G(s)} \tag{5}$$

The above expressions refer to the transfer function of the closed-loop control system. Also the numerator part shows the open loop gain and denominator is loop gain.

Generally, system responses are expressed in time space. In these cases the systems are represented by higher order differential expressions in time space, and the solutions of these expressions would be either very difficult or almost impossible. Laplace transformations are used in response to such problems. Expressions in time space are transformed into functions in the frequency domain.

1.2. PID Control

PID control is a feedback control system used in many areas in the industry that uses three components that are Proportional-Integral-Derivative using the error signal to generate the control signal. A PID controller adjusts the system control input to reduce the output error rate of the system. In order to get optimum efficiency from the PID controller, the system must be generic and the parameters used must be suitable for the system. The selection of parameters is made according to the characteristics of the system. The schematic diagram of the PID controller is shown in Figure 1.2 (Qiu et al., 2017).

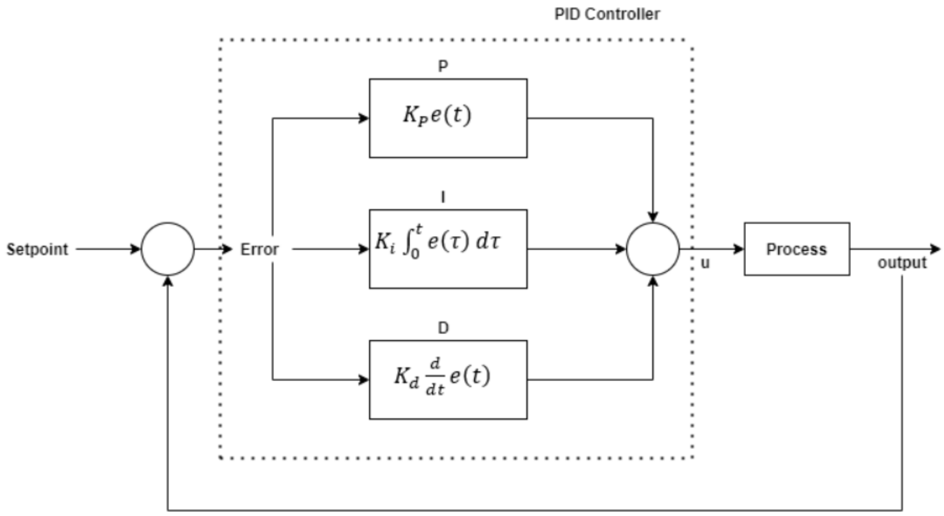


Figure 1.2. Diagram of PID controller

The error is amplified by the proportional mode. The integral term uses the trailing errors for the response. derivative uses the rate of change of the error to generate the response. The sum of the responses of these three terms is used to determine the control signal while performing many control applications (Faruq, 2008).

The general form of a PID control can be represented as:

$$u(t) = K_p e(t) + K_i e(t) + K_d \frac{d}{dt} e(t) \tag{6}$$

In Equation (6), $u(t)$ is the control signal, $e(t)= r(t) - y(t)$ tracking error, $y(t)$ denotes the measured output, $r(t)$ is the desired set point.

If Equation(6) is transformed into Laplace s domian, one obtains :

$$U(s) = K_p e(s) + K_i \frac{1}{s} e(s) + K_d s e(s) \tag{7}$$

If Equation (7) is rearranged as:

$$U(s) = (K_p + K_i \frac{1}{s} + K_d s)e(s) \tag{8}$$

PID controller transfer function is given as:

$$C(s) = K_p + K_i \frac{1}{s} + K_d s \tag{9}$$

The PID control output consists of the sum of the P, I and D outputs.

The P term causes the error output to change proportionally. Response of P term is generated by proportional gain K_p is multiplied by the system error.

$$P_{out} = K_P e(t) \tag{10}$$

The response of the term integral to the system is related to the magnitude and duration of the error. The sum of the error over time forms the natural offset of the system over time of the integral term. The integral gain K_i assigns the weight of the I term on the control.

$$I_{out} = K_i \int_0^t e(\tau)\, d\tau \tag{11}$$

The rate of change of the error is determined by multiplying the derivative gain K_d with the slope of the error over time.

$$D_{out} = K_d \frac{d}{dt} e(t) \tag{12}$$

There are several tuning methods for PID control and design. The three points are considered for PID design as follows:

- Process model identification
- Control structure design
- PID parameters tuning

Process model identification is an essential technique to design a controller. The range of control signals is determined for setting the system output to the desired output, sizing the actuators, selecting the resolution of the sensors. Performance is observed by applying a constant input to the open-loop system for model identificaiton. The following models are commonly used: first-order with time delay model, second-order with time delay model (Yang and Seested, 2013).

$$G(s) = \frac{a}{SL} e^{-sL} \tag{13}$$

$$G(s) = \frac{k_p}{Ts+1} e^{-sL} \tag{14}$$

$$G(s) = \frac{k_p}{(1+sT_2)(1+sT_1)} e^{-sL} \tag{15}$$

The control structure design is important in order to reduce the effect of load disturbances in the system, to avoid measurement noise and to ensure that it is robust against changes. For these reasons, the controller must have (Panagopoulos et al., 2002):

- Load disturbance response
- Measurement noise response

- Set point response
- Robustness

As it is known, PID controller has three parameters: proportional gain K, integral time T_i and derivative time T_D as:

$$PID = K_p\left(1 + \frac{1}{T_i s} + T_D s\right) \tag{16}$$

Many methods have been developed to determine the value of these three parameters. The method to be chosen must have the properties to meet the requirements of the process. Some of the methods used to determine pid parameters are as follows; Ziegler Nichols, Tyreus-Luben, Cohen Coon, Lambda, Chien-Hrones-Reswick PID Tuning.

Ziegler-Nichols tuning is a method used in PID applications and has significant effects. There are two different methods in Ziegler-Nichols tuning. These two methods are based on step response and frequency responce, respectively. The step response of Ziegler-Nichols method is illustrated in Figure 1.3. Calculation of PID parameters with step response method is shown in Tables 1.1.

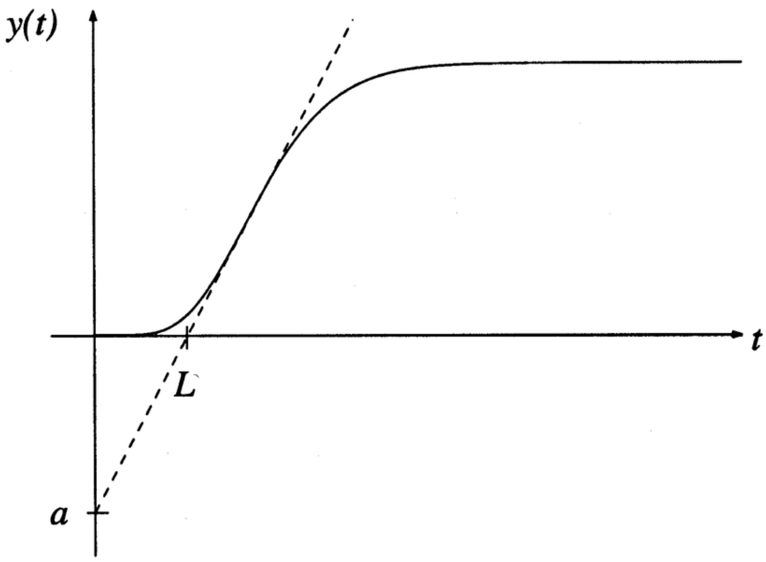


Figure 1.3. Step response of Ziegler-Nichols method

Table 1.1. PID parameter based on Z-N step response method

Parameters	K	T_i	T_d
P	1/a	—	—
PI	0.9/a	3L	—
PID	1.2/a	2L	L/2

The frequency response of Ziegler-Nichols method is illustrated in Figure 1.4. The variables of *Ku* and *Tu* denotes ultimate gain and ultimate period respectively.

Calculation of PID parameters with frequency response method is shown in Tables 1.2.

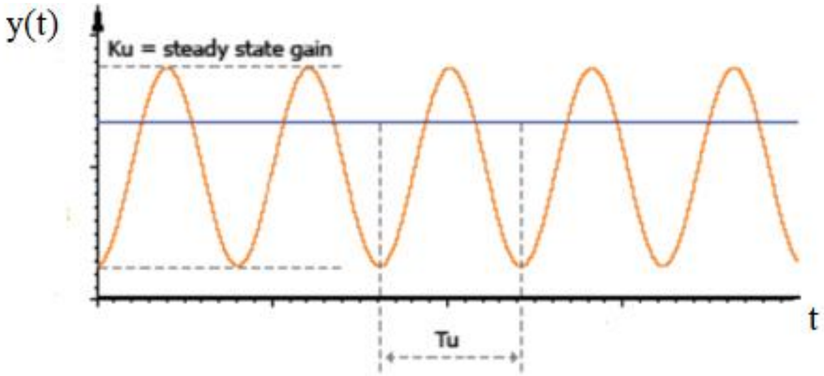


Figure 1.4. Frequency response of Ziegler-Nichols method

Table 1.2. PID parameter based on Z-N frequency response method

Parameters	K	T_i	T_d
P	0.5 Ku	—	—
PI	0.4 Ku	0.8 Tu	—
PID	0.6 Ku	0.5 Tu	0.125 Tu

There are two disadvantages for Ziegler-Nichols tuning method (Åström and Hägglund, 2004):

- Using very little information to characterize system dynamics.
- The quarter amplitude damping results poor damping and poor robustness in closed loop systems.

Classical PID controllers have started to be used with modern control methods in order to improve their performance and eliminate their deficiencies. Some of the modern control methods used in present days to improve controller performance are: fuzzy logic, neural network, iterative learning control (ILC), adaptive control, optimal control, robust control etc. These modern control methods can be applied to systems alone or in combination with others.

1.3. Fuzzy Logic Control

Fuzzy logic control is one of the modern controls used in various applications. Fuzzy control is useful in applications of nonlinear, complex mathematical modeling systems. Fuzzy controller can be applied in the control of nonlinear and time variable systems.

The fuzzy logic control mechanism has internal disciplines such as fuzzy rules and fuzzy inference. The fuzzy rules are based on experience and knowledge and can be used for complex, nonlinear systems. There are a few important points to consider when designing a fuzzy control. These are fuzzy rules, the number and shape of membership functions, the type of fuzzy. There are different types of fuzzy controllers designed with PID according to their usage areas, such as fuzzy PI, fuzzy PD. A design that will improve performance is important in the strucure of PID based fuzzy controllers. When building the fuzzy controller based on PID, the scaling factor can be used to adjust the parameters.

The adjustment of fuzzy PID type system’s parameters has been developed automatically. The PID parameters are adjusted automatically with the information received from the feedback error. In nonlinear systems, a desired modeling may not be done using linear controllers. Nonlinear systems are very difficult to control with conventional pid controllers. Fuzzy logic control is an important development for modeling nonlinear systems. The features of fuzzy control such as simplicity and easy applicability make it preferred in the control of nonlinear systems.

The fact that fuzzy controllers have better performance than conventional PID controllers does not make it perfect in every respect. Like the classical PID control method, the fuzzy control method is insufficient especially in systems that change over time, and it cannot bring the steady state error of the system to a small enough value. Therefore, iterative learnin control (ILC) is presented to eliminate the shortcomings of both classical PID controllers and fuzzy controllers (Elshazly et al., 2012).

1.4. Iterative Learning Control

Iterative learning control (ILC) is one of the modern control techniques. ILC is an effective method of uncertain dynamical systems that work repeatedly. It is run repeatedly to improve the transient response and tracking performance of dynamic systems. The ILC control scheme has an algorithm that improves the system control with the information obtained from previous trials. Repetition is the best way of learning and the ILC bases it on controlled repetitive learning it performs by following this simple rule.

The ILC performs suitably well in the operation of systems such as the periodically repeated trajectory tracking of the robotic arm. Recent ILC studies are grouped as classical ILC and adaptive ILC. Traditional ILC has a learning mechanism that corrects the control input considering the current error and input information from the previous step. Adaptive ILC adapts the system input to be controlled using the advantages of ILC and adaptive control. Control parameters are adjusted by themselves in the learning process created in the Adaptive ILC (Wei et al., 2014).

The fuzzy PID-type iterative learning control developed in the present research is about obtaining the PID parameters required for ILC with fuzzy logic and adapting the system inputs. With the combination of fuzzy logic and ILC, it is aimed to increase the system performance with the advantages of both. The parameters obtained from fuzzy logic are used in PID-type ILC controller with this combined method. In this study, set point, transient response, tracking performance and load disturbance results will be examined.

1.5. Research Objectives

The study objectives are:

- To examine the transient characteristic responses of the DC motor by overshot amplitude, steady-state error and rise time with using clasical

PID controller, fuzzy logic controller, iterative learning controller, fuzzy PID-type ILC controller

- To compare the effects of fuzzy controllers with different numbers of membership functions on DC motors.
- To compare PID-type ILC controller and fuzzy PID controllers
- To analyze the DC motor performance of the fuzzy PID-type ILC controller.
- Analyzing the results of set point, tracking performance and load disturbance experiments and comparing the results of classical PID controller, fuzzy controller, ILC controller, fuzzy PID-type ILC controller
- Combining adaptive control with fuzzy PID-type ILC controller as a scaling factor and examining the performance of the newly created adaptive fuzzy PID-type ILC controller

1.6. The Organization of Thesis

The thesis organization is as follows:

Chapter 1: Introduction

In this section, general information is given about the development of technology and the development of control applications. The operation and adjustment of PID control systems are mentioned. Fuzzy logic control and iterative learning control methods, which are modern control methods, have been introduced. Research Objectives and thesis organization are also included.

Chapter 2: Fuzzy Logic Control

The fuzzy logic theory is explained in general in this section. Fuzzy logic control application, how it works and its integration with pid is emphasized. The design of the PID-type fuzzy control application is explained and shown. Fuzzy control stability analysis was examined.

Chapter 3: Iterative Learning Control

The Iterative Learning Control (ILC) methodology is introduced at the beginning of this chapter. The operation of control systems within the framework of ILC has been examined and the design of the PID-type ILC mechanism is explained and shown. Then, the combination of ILC with fuzzy logic is explained and its application is presented. Moreover, adaptive application is included in the system as a scalar factor to improve the performance of the fuzzy PID-type ILC control system. At the end of this section, there is the stability review of the ILC as given.

Chapter 4: Experimental Setup and Preliminaries

The necessary equipment for the systems to be used in the experiments are introduced in this section. Electromechanical structure and parameters of DC motor are presented. It is mentioned about data aquisition card that provides communication between the experimental setup and the computer.

Chapter 5: Experimental Applications

In the present chapter, results of experimental applications of five different control types are presented. The controllers' performances compared with each other.

Chapter 6: Conclusions

In the final section of the thesis, information such as the main characteristics of the research and the aim of the research are presented at the end of the thesis.

2. FUZZY CONTROL

2.1. Introduction

The term of fuzzy logic first appeared in an article which was written by Dr. Lotfi A. Zadeh in 1965. Classical or Aristotelian logic was generally used to solve problems before the development of fuzzy logic. Classical Aristotelian logic does not contribute enough for the solution of problems. Due to these deficiencies for a required solution, it was possible to create more adequate solutions with the discovery of fuzzy logic theory. Fuzzy logic theory has been explained the uncertainties in the problems and showed a new way for their solutions (Zadeh, 1996).

The classical Aristotelian logic consists of a simple principle, a proposition put forward must be either true or false. There is no possibility other apart from these two results (Lear, 1986). But there can be no definite distinction in daily life activities. For example, a glass of water is either cold or hot within the framework of traditional logic. However it can be 75% cold and 25% hot. In another example, it would not be right to say that a person in his 30s is young or old. According to the fuzzy logic, their probability statements with the membership values the situations are mentioned instead of the exact statements.

2.2 Fuzzy Logic Control Theory

In conventional control design, matematical system models are used to design controllers and investigate system effectiveness. Fuzzy logic control contains fuzzy sets and fuzzy inference to set a good sample for systems which have no exact model. The main point of fuzzy logic controller is based on knowledge and experience to create control laws (Mamdani, 1974). Control actions are provided from fuzzy rule base. Fuzzy rule base is consisting of if-then conditional statement. If-then statements are continuous membership functions. The inference is the deduction of rules. The definition of memberships functions are necessary for inference. These membership functions provide definition of truth scale for every situation.

The diagram of fuzzy logic controller system is shown in Fig. 2.1. The controller sub operational parts are fuzzification, inference engine and defuzzification. The fuzzification makes the real world variables translated to fuzzy sets variables. In the section of inference engine, control actions are generated by fuzzy inference rules. Fuzzy sets with fuzzy logic operators are defined and inference methods are made with compitational terms. In the defuzzification section, computed values which are calculated by fuzzy computations and fuzzy logic operators are translated to real world values for fuzzy control action.

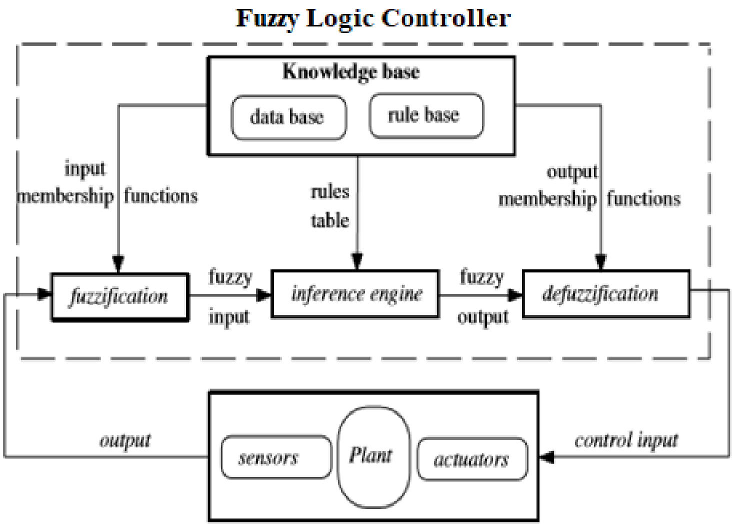


Figure 2.1. Fuzzy Control System block diagram

2.2.1. Fuzzification and Membership Functions

The first block of fuzzy logic controller is fuzzification. Fuzzification is the method of an crisp quantity into a fuzzy quantity. The purpose of the fuzzification is that choice of membership, to transform the input from a numerical value into a set of fuzzy variables (Walia et al., 2015). In the fuzzification block, the numerical input values are converted to verbal expressions and membership values ranging from 0 and 1 in response to these verbal expressions. For example, the error and the error change are considered as the input values. In the first instance, a set of rules are described such as ‘’Positive (P)’’, ‘’Zero (Z)’’, ‘’Negative (N)’’ for the error and error change values. Then, depending on the error and error change, the following verbal definitions are made.

IF the error is ‘‘Positive (P)’’ and the error change is ‘‘Negative(N)’’, Then the output is ‘‘Zero(Z)’’.

IF the error is ‘‘Positive (P)’’ and the error change is ‘‘Positive (P)’’, Then the output is ‘‘Negative(N)’’.

In the fuzzification, the membership functions have very important role in the final performance of fuzzy control and the preference of the membership functions has got a serious impact on the control law (Bağiş, 2003; Sun and Liu, 2002). Many different types of membership function are used in fuzzy controller such as gaussian function, bell function, trapezoidal function and triangular function (D’Errico, 2001). The most common used membership type is the triangular type in fuzzy controller application because of its simplicity and efficiency (Ma et al., 2012).

2.2.2. Rule Base

Fuzzy sets are used to provide symbolic knowledge information in a more human understandable or natural form, and can hold uncertainties at various levels (Mitra and Hayashi, 2000). Fuzzy rule based system to manipulate a required ‘‘IF-THEN’’ fuzzy rules, addressing the universe of division and membership. Fuzzy rules may be simply and directly developed by consultants within the style of linguistic rules (Taylan and Karagözoğlu, 2009). Fuzzy rule-based systems use linguistic variables to principle employing a series of logical rules that contain IF-THEN rules that connect antecedent(s) and consequent(s), severally. degree antecedent may be a fuzzy clause with a particular degree of membership (between zero and 1). Fuzzy rule can have several antecedants linked to AND or OR operators, where all pieces are simultaneously considered and resolved into a one number.

Consequents may also consists of numerous parts, that are then totaled into a single output of a fuzzy set (Hamdan and Garibaldi, 2010).

There are four head strategies utilized in development of a standard base (Yan et al., 1994):

- Control engineering knowledge
- Modeling the operator's behaviour
- Fuzzy modelling
- Self learning fuzzy controller

In accordance with control enginnering and modelling the operator's behaviour, rule constitution is considered to be heuristic methods. The other methods of rule base generation are defined as deterministic methods. These two separately different methods, heuristic and deterministic, were used by 'Mamdani' and 'Takagi and Sugeno' respectively. In situations that require more engineering skill and experience, the control enginnering knowledge method which is based on IF-THEN rules, is commonly used.

2.2.3. Inference Engine

Inference engine is a sub-operational block of fuzzy logic control. It is an interface that produce a new fuzzy set. Inference engine is described as the software code forms the standards, cases, objects or alternative variety of data and mastery dependent on the realities of a given circumstance. An inference engine is an scientific discipline system that systematically uses reasoning steps just like that of a human mind. For each rule inference mechanism employs the membership values and consistent with the state of the rule involves an outcome.

There are two main groups for inference mechanism in fuzzy control (Horiuchi and Kishimoto, 2002):

- The direct inferencing
- The indirect inferencing

The direct inferencing decides straight forwardly the output from knowledge base and on line information by min-max operational activities. This procedure allows simple automation of device operation using expert operator know-how. This accualy moves the knowledge of operator to the system using If-Then rules and membership functions. Otherwise, the fuzzy inference within the indirect inferencing is employed the present physiological condition supported on line knowledge first, afterwards, observational control arrangements are assigned to each state. Control values are resolved based on the inferred state. It is observed that indirect inferencing uses a few rules. This methodology is known as the Sugeno-Takagi methodology.

2.2.4. Defuzzification

Defuzzification is basically the reverse of fuzzification. Defuzzification is the process that generates an actual value from the inference that is used as fuzzy control input. There are some methods for gettin the fuzzy control signal in the defuzzification:

- Maximum criterion
- Mean of maximum
- Center of gravity
- Bisector of area
- Leftmost maximum and rightmost maximum

These methods are implemented in the defuzzification (Amini and Nikraz, 2016). The centre of gravity is the most commonly used method among the defuzzzication methods (Kuo and Lin, 2002; Bobyr et al., 2017). The formulation of crisp value of the fuzzy control output using center of gravity method is shown by

$$U_{fz} = \frac{\sum_{i=1}^{n}\{membership(input_i)\times output_i\}}{\sum_{i=1}^{n}\{membership(input_i)\}} \tag{2.1}$$

where ***i*** is rule number**.**

2.3. Fuzzy PID Logic Control

There is some steady state error in conventional controller in which the system has no inherent integrated property. Fuzzy PID logic controller is one of the improved methods instead of conventional controller (Mirinejad et al., 2012). However, fuzzy PID control has the different structure to be more practical than conventional controllers. A typical fuzzy PID controllers have two input and three output with the error *e* and the change of error *ec* as input and *Kp*, *Ki*, *Kd* are the coefficients of PID parameters as:

$$\begin{aligned} K_p{}' &= K_p + \Delta K_p \\ K_i{}' &= K_i + \Delta K_i \\ K_d' &= K_d + \Delta K_d \end{aligned} \tag{2.2}$$

where *Kp*, *Ki*, *Kd* are nominal control parameters, $K_p{}', K_i{}', K_d{}'$ are the revised parameters, ΔK_p , ΔK_i , ΔK_d are the parameters to be calculated (Wanget al., 2017). The structure of PID fuzzy logic controller diagram is shown in Figure 2.2.

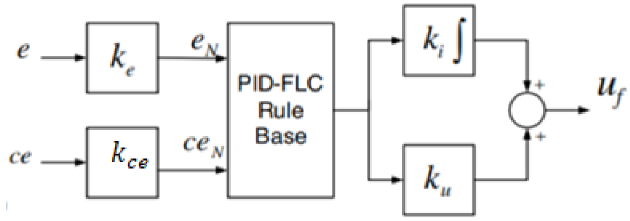
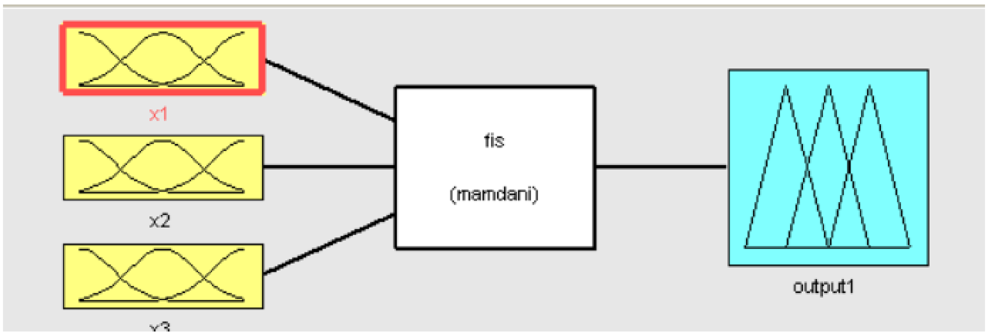
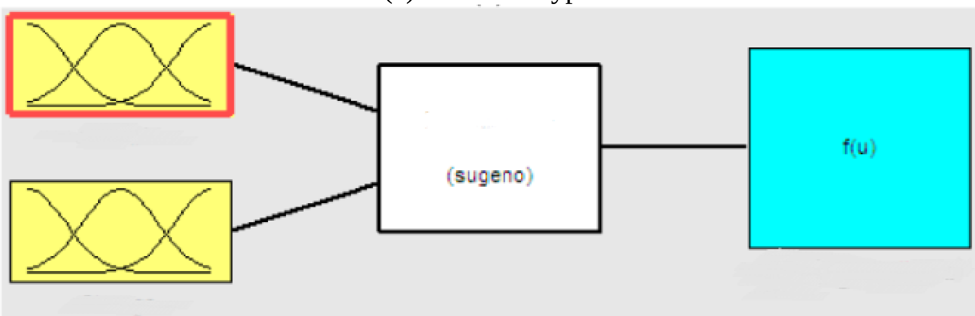


Figure 2.2. PID Fuzzy controller diagram

In the fuzzy PID controller, there are two different controller arrangements; Mamdani type and Sugeno type. These two of fuzzy systems have some differences between them. The main difference between Mamdani type and Sugeno type is the way the crisp output is generated from the fuzzy input. The defuzzification technique is used for a fuzzy output in Mamdani type system but in Sugeno type, weighted avarege is used to compute the crisp output. The another difference is that Mamdani type system has output membership functions, on the other hand Sugeno type does not have any output membership functions (Kaur and Kaur, 2012). Mamdani and Sugeno type systems are shown in Figure 2.3.



(a) Mamdani type



(b) Sugeno type

Figure 2.3. (a) Mamdani type, (b) Sugeno type

2.3.1. Design of Fuzzy PID Controller

Fuzzy PID controller modelling is one of the most used control methods (Rajagiri et al., 2019). There are several advanteges of fuzzy PID controller:

- Simplicity of design
- Cost effective
- No mathematical modelling of the system is needed
- Linguistic variables are used instead of numerical variables
- Non-linearity of the system is able to be handled without difficulty

A fundamental structure of fuzzy PID control system is shown in Figure 2.4.

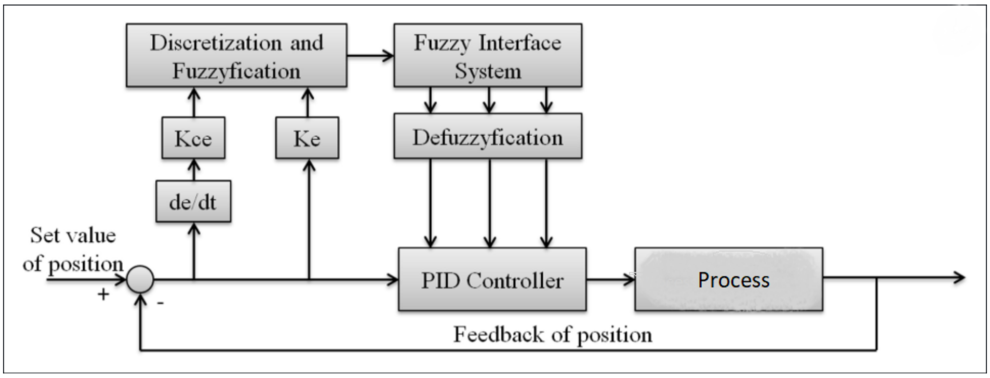


Figure 2.4. Structure of Fuzzy PID Controller System

Input variables of fuzzy control consist of two signals which are error signal and change of error signal. These two signal variables are used under the way of fuzzy control rules and the result of this operation generates the outpu variables which is determined by the defuzzification process.

There are important steps before generating the output variables. It is necessary to define design planning, parameter tunning, fuzzy logic controller operation. The first step is design planning. Essential factors of design planning:

- Determination of system input and output variables
- Computation of number of fuzzy parts
- Choice of membership function types
- Obtain fuzzy control rules
- Define inference system
- Choice of defuzzification technique

The second step is parameter tunning and this step involves following statements:

- Mapping of membership functions

- Fuzzy inference rules
- Scaling factors

The third step is fuzzy logic logic controller operation:

- Fuzzification
- Fuzzy inference
- Defuzzification

In order to define fuzzy membership function, there are generally used four types of membership functions used:

- Trapezoidal membership function
- Triangular membership function
- Gaussian membership function
- Generalized bell membership function

The error (e) and change of error (ce) are the inputs, K_p, K_i and K_d are the outputs of Fuzzy PID controller. Generally number of membership functions for each input and output variables can be 3, 5 or 7. Consider that 7 membership functions are used for each variable and these membership functions are classsified as: PB (Positive Big), PM (Positive Medium), PS (Positive Small), Z (Zero), NS (Negative Small), NM (Negative Medium), NB (Negative Big). Membership function for inputs and outputs are shown in Figure 2.5.

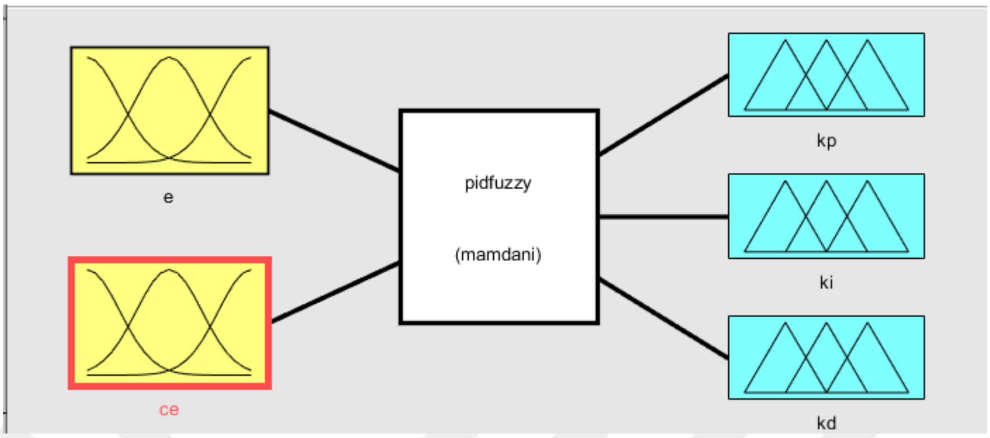


Figure 2.5. Membership function for input and output of fuzzy logic controller

Classified membership functions are shown in Fig.2.6.

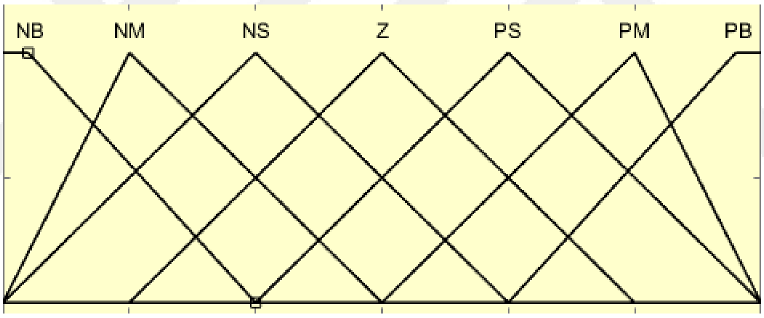


Figure 2.6. Membership functions

In the rule based, fuzzy logic control rules are derived with using error and change of error membership functions. The number of rules depends on number of membership functions of input (3x3, 5x5 and 7x7). Rules are generated by If-Then logic structure. If the number of membership functions is seven (7), the total number of rules becomes 49. The fuzzy logic rule based tables are shown in Table 2.1., Table 2.2. and Table 2.3.

Table 2.1. K_p Rule table of Fuzzy Logic Controller.

ce \ e	PB	PM	PS	Z	NS	NM	NB
PB	NB	NB	NM	NM	NM	Z	Z
PM	NB	NM	NM	NM	NS	Z	PS
PS	NM	NM	NS	NS	Z	PS	PS
Z	NM	NM	NS	Z	PS	PM	PM
NS	NS	NS	Z	PS	PM	PM	PM
NM	NS	Z	PS	PS	PM	PB	PB
NB	Z	Z	PS	PM	PM	PB	PB

Table 2.2. K_i Rule table of Fuzzy Logic Controller

e \ ce	PB	PM	PS	Z	NS	NM	NB
PB	PB	PB	PM	PM	PS	Z	Z
PM	PB	PB	PM	PS	PS	Z	Z
PS	PB	PM	PS	PS	Z	NS	NM
Z	PM	PM	PS	Z	NS	NM	NM
NS	PS	PS	Z	NS	NS	NM	NB
NM	Z	Z	NS	NS	NM	NB	NB
NB	Z	Z	NS	NM	NM	NB	NB

Table 2.3. K_d Rule table of Fuzzy Logic Controller

ce \ e	PB	PM	PS	Z	NS	NM	NB
PB	PB	PS	PS	PM	PM	PM	PB
PM	PB	PS	PS	PS	PS	PS	PB
PS	Z	Z	Z	Z	Z	Z	Z
Z	Z	NS	NS	NS	NS	NS	Z
NS	Z	NS	NS	NM	NM	NB	Z
NM	Z	NS	NM	NM	NB	NS	PS
NB	PS	NM	NB	NB	NB	NS	PS

The fuzzy logic control rules are:

1.) If (e is NB) and (ce is NB) then (K_p is PB) (K_i is NB) (K_d is PS)
2.) If (e is NB) and (ce is NM) then (K_p is PB) (K_i is NB) (K_d is NS)
3.) If (e is NB) and (ce is NS) then (K_p is PM) (K_i is NM) (K_d is NB)
4.) If (e is NB) and (ce is Z) then (K_p is PM) (K_i is NM) (K_d is NB)
5.) If (e is NB) and (ce is PS) then (K_p is PS) (K_i is NS) (K_d is NB)
6.) If (e is NB) and (ce is PM) then (K_p is Z) (K_i is Z) (K_d is NM)
7.) If (e is NB) and (ce is PB) then (K_p is Z) (K_i is Z) (K_d is PS)
8.) If (e is NM) and (ce is NB) then (K_p is PB) (K_i is NB) (K_d is PS)
9.) If (e is NM) and (ce is NM) then (K_p is PB) (K_i is NB) (K_d is NS)
10.) If (e is NM) and (ce is NS) then (K_p is PM) (K_i is NM) (K_d is NB)
11.) If (e is NM) and (ce is Z) then (K_p is PS) (K_i is NS) (K_d is NM)
12.) If (e is NM) and (ce is PS) then (K_p is PS) (K_i is NS) (K_d is NM)
13.) If (e is NM) and (ce is PM) then (K_p is Z) (K_i is Z) (K_d is NS)
14.) If (e is NM) and (ce is PB) then (K_p is NS) (K_i is Z) (K_d is Z)
15.) If (e is NS) and (ce is NB) then (K_p is PM) (K_i is NB) (K_d is Z)
16.) If (e is NS) and (ce is NM) then (K_p is PM) (K_i is NM) (K_d is NS)
17.) If (e is NS) and (ce is NS) then (K_p is PM) (K_i is NS) (K_d is NM)
18.) If (e is NS) and (ce is Z) then (K_p is PS) (K_i is NS) (K_d is NM)
19.) If (e is NS) and (ce is PS) then (K_p is Z) (K_i is Z) (K_d is NS)
20.) If (e is NS) and (ce is PM) then (K_p is NS) ($K_i$is PS) (K_d is NS)
21.) If (e is NS) and (ce is PB) then (K_p is NS) (K_i is PS) (K_d is Z)
22.) If (e is Z) and (ce is NB) then (K_p is PM) (K_i is NM) (K_d is Z)
23.) If (e is Z) and (ce is NM) then (K_p is PM) (K_i is NM) (K_d is NS)
24.) If (e is Z) and (ce is NS) then (K_p is PS) (K_i is NS) (K_d is NS)

25.) If (e is Z) and (ce is Z) then (K_p is Z) (K_i is Z) (K_d is NS)

26.) If (e is Z) and (ce is PS) then (K_p is NS) (K_i is PS) (K_d is NS)

27.) If (e is Z) and (ce is PM) then (K_p is NM) (K_i is PM) (K_d is NS)

28.) If (e is Z) and (ce is PB) then (K_p is NM) (K_i is PM) (K_d is Z)

29.) If (e is PS) and (ce is NB) then (K_p is PS) (K_i is NM) (K_d is Z)

30.) If (e is PS) and (ce is NM) then (K_p is PS) (K_i is NS) (K_d is Z)

31.) If (e is PS) and (ce is NS) then (K_p is Z) (K_i is Z) (K_d is Z)

32.) If (e is PS) and (ce is Z) then (K_p is NS) (K_i is PS) (K_d is Z)

33.) If (e is PS) and (ce is PS) then (K_p is NS) (K_i is PS) (K_d is Z)

34.) If (e is PS) and (ce is PM) then (K_p is NM) (K_i is PM) (K_d is Z)

35.) If (e is PS) and (ce is PB) then (K_p is NM) (K_i is PB) (K_d is Z)

36.) If (e is PM) and (ce is NB) then (K_p is PS) (K_i is Z) (K_d is PB)

37.) If (e is PM) and (ce is NM) then (K_p is Z) (K_i is Z) (K_d is PS)

38.) If (e is PM) and (ce is NS) then (K_p is NS) (K_i is PS) (K_d is PS)

39.) If (e is PM) and (ce is Z) then (K_p is NM) (K_i is PS) (K_d is PS)

40.) If (e is PM) and (ce is PS) then (K_p is NM) (K_i is PM) (K_d is PS)

41.) If (e is PM) and (ce is PM) then (K_p is NM) (K_i is PB) (K_d is PS)

42.) If (e is PM) and (ce is PB) then (K_p is NB) (K_i is PB) (K_d is PB)

43.) If (e is PB) and (ce is NB) then (K_p is Z) (K_i is Z) (K_d is PB)

44.) If (e is PB) and (ce is NM) then (K_p is Z) (K_i is Z) (K_d is PM)

45.) If (e is PB) and (ce is NS) then (K_p is NM) (K_i is PS) (K_d is PM)

46.) If (e is PB) and (ce is Z) then (K_p is NM) (K_i is PM) (K_d is PM)

47.) If (e is PB) and (ce is PS) then (K_p is NM) (K_i is PM) (K_d is PS)

48.) If (e is PB) and (ce is PM) then (K_p is NB) (K_i is PB) (K_d is PS)

49.) If (e is PB) and (ce is PB) then (K_p is NB) (K_i is PB) (K_d is PB)

In Figure 2.7, the structure of fuzzy PID controller system for Matlab/Simulink is shown.

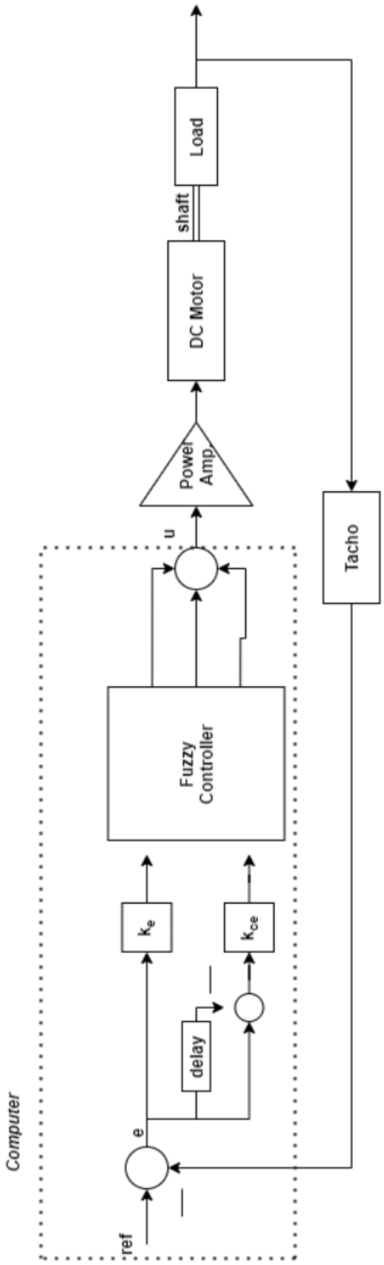


Figure 2.7. Block diagram of fuzzy PID control system using MATLAB/Simulink

Fuzzy PID control system is builded with using MATLAB/Simulink program. As it is seen, input and output scaling factors are used to have significant effect on stability and performance for fuzzy PID controller system. The surfaces of K_p, K_i and K_d are shown in Fig.2.8, Fig.2.9 and Fig.2.10 respectively.

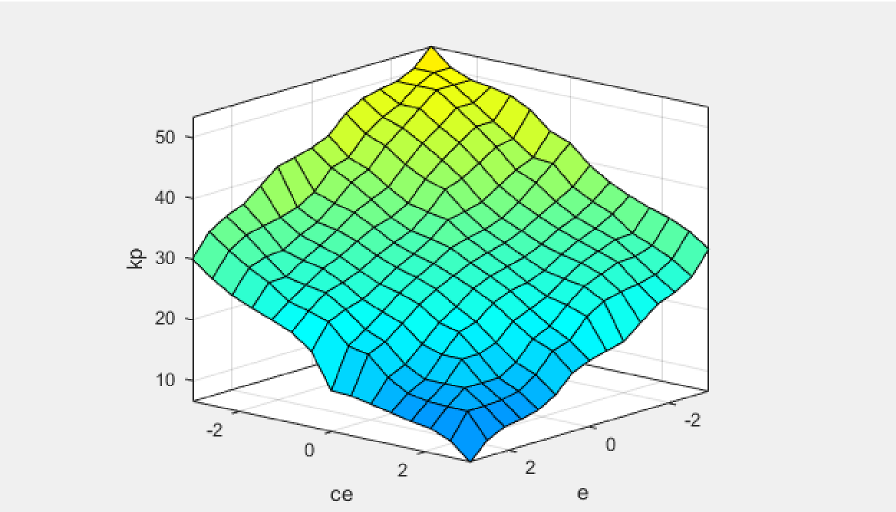


Figure 2.8. Surface view of K_p

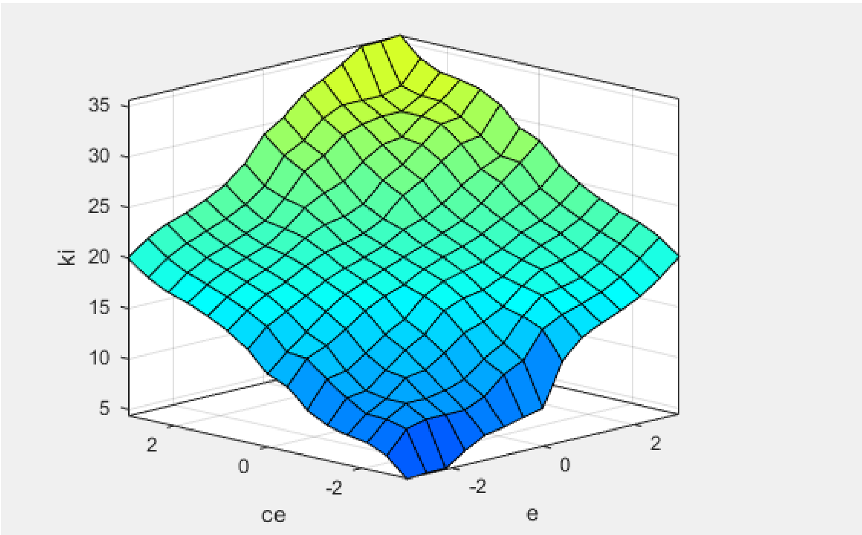


Figure 2.9. Surface view of K_i

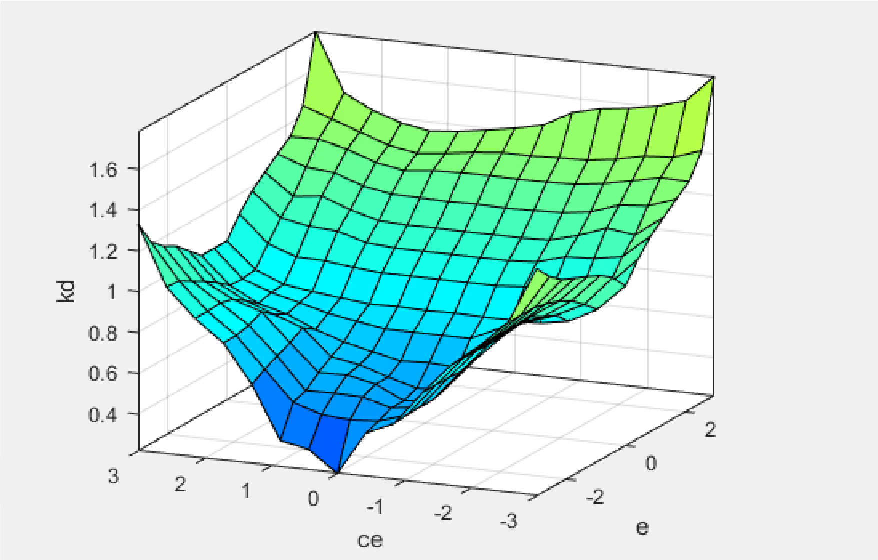


Figure 2.10. Surface view of K_d

2.4. Stability Analysis of Fuzzy PID Controller

Closed loop fuzzy PID controller and its signals are illustrated in Fig.2.11. In this way;

$y_{sp}(nT)$ → reference set point

$y(nT)$ → output variable

$e(nT)$ → error signal and $e(nT) = y_{sp}(nT) - y(nT)$

$r(nT)$ → change of error signal

$u_{PID}(nT)$→ fuzzy control singnal , where sampling time *T>0.*

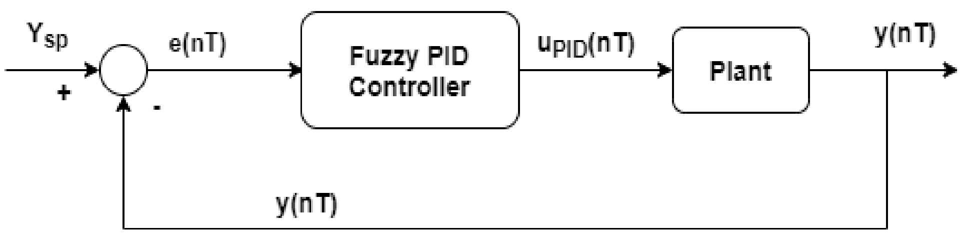


Figure 2.11. Closed-Loop Fuzzy PID Controller and Its signals

The control signal is defined by mathematically summing of fuzzy P, fuzzy I and fuzzy D simultaneously (Kumar and Mittal, 2010). The control signal is $u_{PID}(nT) = u_P(nT) + u_I(nT) + u_D(nT)$

$$= K_{UP}u_P(nT) + u_I(nT-T) + K_{UI}\Delta u_I(nT) - u_D(nT-T) + K_{UD}u_D(nT) \quad (2.3)$$

where $u_P(nT)$ is fuzzy P signal, K_{UP} is fuzzy P gain, $\Delta u_I(nT)$ is fuzzy I incremental signal, K_{UI} is fuzzy I gain, $\Delta u_D(nT)$ is fuzzy D incremental signal, K_{UD} is fuzzy D gain. $u_P(nT), \Delta u_I(nT), \Delta u_D(nT)$ are determined by defuzzification algorithm to each membership.

The small gain theorem is used for stability analysis of fuzzy PID control system as shown in Fig.2.12 (Galichet and Boukezzoula, 2008).

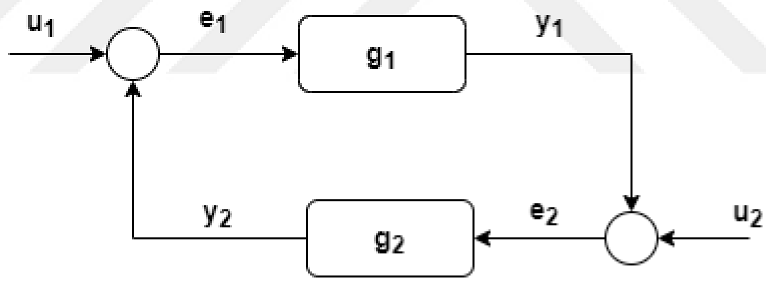


Figure 2.12. Closed-Loop Control System

There are two errors (e_1 and e_2), two inputs (u_1 and u_2) and also two outputs (y_1 and y_2).

$$\begin{aligned} e_1 &= u_1 - y_2 \\ e_2 &= u_2 + y_1 \\ y_1 &= g_1(e_1) \\ y_2 &= g_2(e_2) \end{aligned} \quad (2.4)$$

Consider that g_1 and g_2 are BIBO stable and,

$$\| \ y_1 \ \| \leq \alpha_1 \ \| \ e_1 \ \| + \delta_1 \tag{2.5}$$

$$\| \ y_2 \ \| \leq \alpha_2 \ \| \ e_2 \ \| + \delta_2 \tag{2.6}$$

$\alpha_1 = \alpha\,(\,g_1\,)$ is the gain of g_1, $\alpha_2 = \alpha\,(\,g_2\,)$ is the gain of g_2. δ_1 and δ_2 are constant, α_1 , $\alpha_2 \geq 0$.

If α_1 >1 and α_2 >1, then the system is accepted BIBO stable by these two conditions, (2.5) and (2.6) (Galichet and Boukezzoula, 2008).

The equivalent block diagram of fuzzy PID control system is shown in Fig.2.13. The plant and the fuzzy PID controller is represented by g_2 and g_1 respectively.

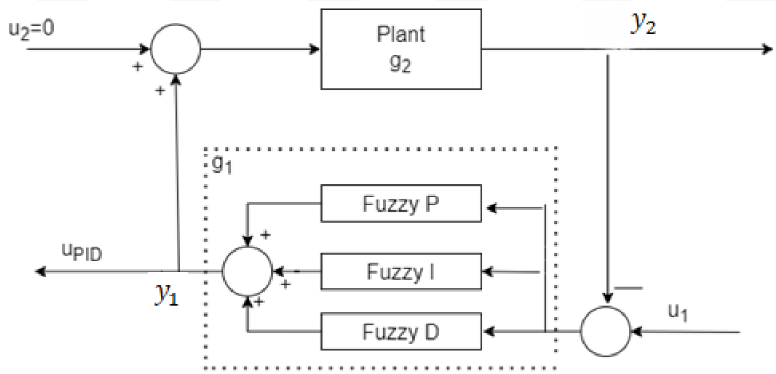


Figure 2.13. The equivalent block diagram of fuzzy PID control system

$$\| \ y_1 \ \| = \left\| g_1 \left(\begin{bmatrix} e \\ e \\ e \end{bmatrix} \right) \right\| \leq \alpha_1 \left\| \begin{bmatrix} e \\ e \\ e \end{bmatrix} \right\| + \delta_1 \tag{2.7}$$

$$\| \ y_2 \ \| = \| \ g_2\,(u_{PID}) \ \| \leq \ \alpha_2 \ \| \ u_{PID} \ \| + \delta_2 \tag{2.8}$$

where α_1 , α_2, δ_1 , δ_2 are constant.

Expanding of Equation (2.7) gives:

$$\| \ y_1 (nT) \| = \| K_{UP} u_P(nT) + K_{UI} \Delta u_I(nT) + K_{UD} u_D(nT) \tag{2.9}$$

Or

$$\| \ y_1 (nT) \| = \| K_{UP} \frac{L\left[K_P^2 \Delta e(nT) - K_P^1 e(nT)\right]}{2\left[2L - |K_P^1 e(nT)\ |\right]} + K_{UI} \frac{L\left[K_I^1 \Delta e(nT) + K_I^2 r(nT)\right]}{2\left[2L - |K_I^1 e(nT)\ |\right]} + K_{UD} \frac{L\left[K_D^2 \Delta e(nT) - K_I^1 r(nT)\right]}{2\left[2L - |K_D^1 e(nT)\ |\right]} \| \tag{2.10}$$

r is given as: $r(nT) = \frac{e(nT) - e(nT-T)}{T}$, and $\Delta e(nT) = e(nT) - \Delta e(nT - T)$

Rewriting of Equation (2.10) with using $r(nT)$ and $\Delta e(nt)$ is

$$\| \ y_1 (nT) \| \leq \left| \frac{LK_{UP}(K_P^2 - K_P^1)}{2(2L - K_P^1 M_e)} + \frac{LK_{UI}(TK_I^1 + K_I^2)}{2T(2L - K_I^1 M_e)} + \frac{LK_{UD}(TK_D^2 + K_D^1)}{2T(2L - K_D^2 M_e)} \right| |e_1 (nT)| + \left| \frac{LK_{UP} K_P^2 M_e}{2(2L - K_P^1 M_e)} + \frac{LK_{UI} K_I^2 M_e}{2T(2L - K_I^1 M_e)} - \frac{LK_{UD} K_D^1 M_e}{2T(2L - K_D^2 M_e)} \right| \tag{2.11}$$

where $M_e \coloneqq \ sup_{n\geq 0} |e(nT)| = \ sup_{n\geq 1} |e(nT - T)|$ is the maximum magnitude of error signal (Kumar et al., 2013).

Membersip functions of input and output variable for fuzzy PID control with rule based are considered to constitute the regions of fuzzy PID controller Input Combination (IC). These regions are illustrated in Fig.2.14.

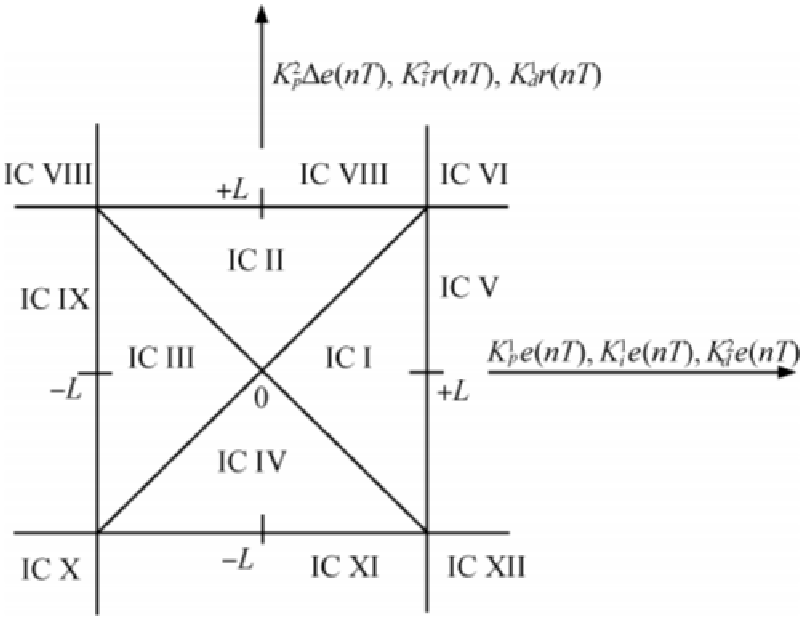


Figure 2.14. The regions of fuzzy PID controller values

The circunstance which the controlled process is nonlinear, is represented by R. Given that control operation in various IC regions is different, stability might be examined for each IC region. Assume that error and change of error are located in IC I and IC III, so that rewriting and rearranging are,

$$\alpha_1 = \left| \frac{LK_{UP}(K_P^2 - K_P^1)}{2(2L - K_P^1 M_e)} + \frac{LK_{UI}(TK_I^1 + K_I^2)}{2T(2L - K_I^1 M_e)} + \frac{LK_{UD}(TK_D^2 + K_D^1)}{2T(2L - K_D^2 M_e)} \right| \quad (2.12)$$

$$\delta_1 = \left| \frac{LK_{UP}K_P^2 M_e}{2(2L - K_P^1 M_e)} + \frac{LK_{UI}K_I^2 M_e}{2T(2L - K_I^1 M_e)} + \frac{LK_{UD}K_D^1 M_e}{2T(2L - K_D^2 M_e)} \right| \quad (2.13)$$

are the constant. $\| y_2 \| = \| R(e_1(nT)) \|$, or $\| y_2 \| \leq \| R \| \, |e_2(nT)|$ which is form of (2.6) with

$$y_2 = \| R \| < \infty \quad (2.14)$$

where $|| R ||:= sup_{v_{1 \neq v_2}\ n \geq 0} \frac{|R(v_1(nT)) - R(v_2(nT))|}{|(v_1(nT)) - (v_2(nT))|}$ is operator norm for the R. This is the gain of the non-linear process.

Accompanied by Equation (2.11) and Equation (2.14), the following cases are created by the small gain theory for BIBO stability of non-linear fuzzy PID controller in region IC I and IC III (Kumar et al., 2013; Prieto-Entenza et al., 2019).

1. $|| R ||< \infty$
2. $\left| \frac{LK_{UP}(K_P^2 - K_P^1)}{2(2L - K_P^1 M_e)} + \frac{LK_{UI}(TK_I^1 + K_I^2)}{2T(2L - K_I^1 M_e)} + \frac{LK_{UD}(TK_D^2 + K_D^1)}{2T(2L - K_D^2 M_e)} \right| \times || R ||<1$

In the similar way, the BIBO stability cases might be provided for other IC regions. The term of α_1 and the gain of g_1, fuzzy PID controller, are observed to be distinct in different IC regions. For that reason, Theorem 1 shows the explanation.

Theorem 1: The cases which are necessary for non-linear fuzzy PID controller to be stable are

- The non-linear plant that is controlled has a norm (gain) $|| R ||< \infty$
- Fuzzy PID controller parameters provide $\alpha_1\ || R ||< \infty$

2.5. Summary

In this chapter, the fuzzy logic theory is introduced and the systems of the fuzzy logic control structure are explained. Fuzzy PID logic control mechanism is designed. The dynamics of the designed fuzzy controller are explained. It is specified how to create membership functions for fuzzy rules. It is defined how Fuzzy rules will be selected for the system. Stability analysis of the Fuzzy controller is included.

3. ITERATIVE LEARNING CONTROL

3.1. Introduction

There are various intelligent control methodologies. Itertaive Learning Control (ILC) is one of the intelligent control branches and it is shown in Fig.3.1. ILC is an effective method to improve non-linear strong coupling control systems and transient response, tracking performance with repetitive motion, especially used to control moving system whose operation is repetitive and eliminate error trial by trial.

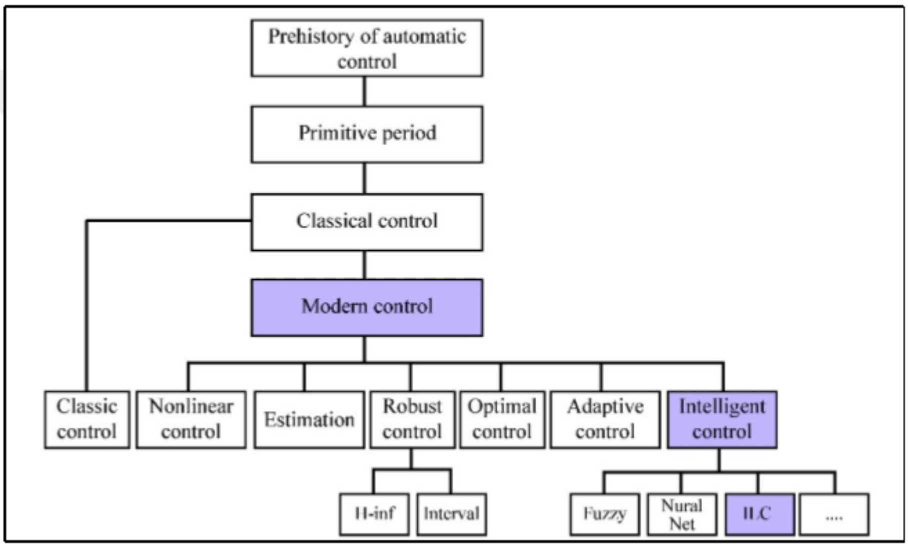


Figure 3.1. Classification of Control Application Methods

In self learning study field, it can be said that ILC has an effective control algorithm for control systems. The ILC algorithm has ability to track the required trajectory in a specified timeframe (Ashraf et al., 2008).

ILC concept is mainly similar to human learning activities, if repetition of an activity is increased, the more experiences are simply possessed and the performance levels up. Setup, uniform sampling, fixed time, repetitive desired trajectory can be accepted similar to human qualities in systems (Ahn et al., 2007).

The idea of iterative learning control was thought in the first half of 20th century, however the first publication was in 1974, and in 1978 the first paper was published in Japanese by Masaru Uchiyama (Uchiyama, 1978). The formulation of problem of iterative learning control was carried out very carefully by Arimoto in 1980s (Arimoto et al., 1984). Arimoto has researched on the convergence of ILC framework to a specific scale of speculations. The convergence of ILC for D-type and PD-type to a considerable extent was reached by Lee (Lee and Bien, 1996). Iterative tracking was achieved at the same time system initializing with system input for inital value of ILC problem by Chen (Chen et al.,1999). PD and D-type ILC sysyems' convergence problem was examined by Mingxuan when the system initial value is biased (Hao and Wang, 2017). The sudy of Xuemei has shown that input and initial value of learning simultaneously, system initial case with no desire at the beginning of learning, the results demostrate the feasibility (Hao and Wang, 2017). The concept which provided in any initial ILC algorithm under under suitable conditions for convergence was put forward for iterative learning controller in the frequency domain. The effect of accelerating the suppresion of random initial state error in ILC system was analyszed by Lu (Lu et al., 2014). The both open and closed loop ILC for nonlinear time delay systems were studied.

There are many alternative definition about ILC in the literature Commonly used of these definitions are listed as:

- The learning control concept stands for the repeatability of operating a given objective system and the possibility of improving the control input on the basis of previous actual operation data (Arimoto et al., 1986)
- It is a recursive online control method that relies on less calculation and requires less a priori knowledge about the system dynamics. The idea

is to apply a simple algorithm repetitively to an unknown plant, until perfect tracking is achieved (Bien and Huh, 1989).

- ILC is an approach to improving the transient response performance of the system that operates repetitively over a fixed time interval (Moore, 2012).
- ILC considers systems that repetitively perform the same task with a view to sequentially improving accuracy (Amann et al., 1996).
- ILC is to utilize the system repetitions as experience to improve the system control performance even under incomplete knowledge of the system to be controlled (Chen and Wen, 1999).
- The controller that learns to produce zero tracking error during repetitions of a command, or learns to eliminate the effects of a repeating disturbance on a control system output (Phan et al., 2000).
- The main idea behind ILC is to iteratively find an input sequence such that the output of the system is as close as possible to a desired output. Although ILC is directly associated with control, it is important to note that the end result is that the system has been inverted (Markusson, 2001).
- We learned that ILC is about enhancing a system's performance by means of repetition, but we did not learn how it is done. This brings us to the core activity in ILC research, which is the construction and subsequent analysis of algorithms (Verwoerd, 2004).

All of the above definitions have mentioned a certain point. The certain point of these definitions is the idea of repetition. One of the main ideas of ILC is the learning with predetermined hardware repetititon. Hardware repetition is a physical part that brings samples to the memory of the ILC through experiments. ILC has the datas of past experiment by storing them and current activity is a control force which is produced by using past experimental informations. The learning activity of ILC has got different properties if comparing with human learning. The learning activity of ILC is based on initial setup, repetitive desired trajectory, fix time point, ect. However, the human learning is based on similarities and impressions.

Iterative learning control is used for many various application fields in modern technology era. The usage fields of ILC consists of robotic applications, rotary systems, process control, actuators, power systems, ect (Ahn et al., 2007). There are areas where ILC method is used with the kalman filter (Liu et al., 2020; Hock and Schoellig, 2019). The table of these usage is shown in Figure 3.2.

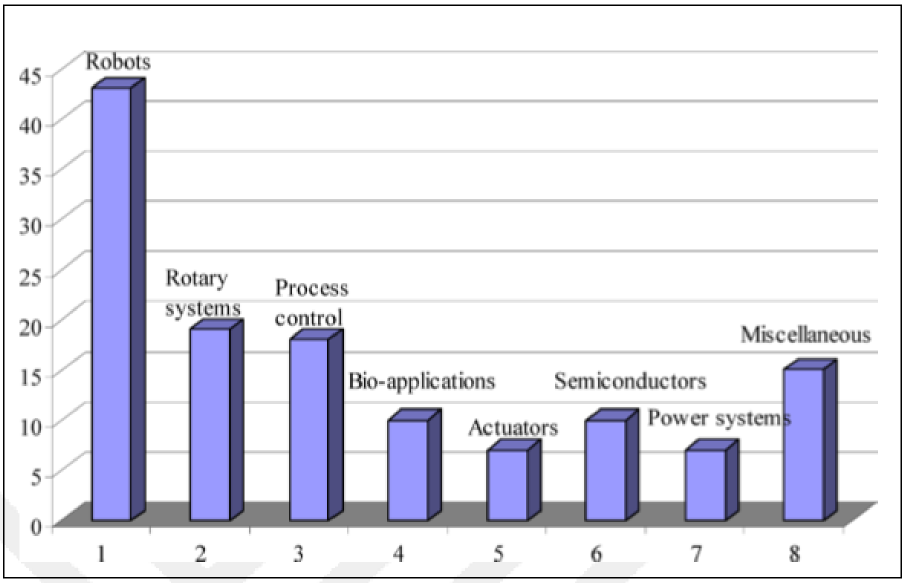


Figure 3.2. Application area of ILC

The majority of ILC applications are in the area of robotic system.

In addition, there are many publication about iterative learning control. When analyzed from past to present, there is a continuous increase in the number of publications in scientific journals and conferences (Ahn et al., 2007; Shen, 2018).

This increase in publications about ILC, a new method, shows that ILC is an effective method and will be a popular method in the future.

3.2. Theory of Iterative Learning Control

Iterative learning control is one of the well established and useful method in control theory. Although it is relatively new study concept, research and studies show that ILC is sufficiently successful in control systems. ILC, one of the intelligent control methods, has provided a way to improve the transient performance of repetitive systems over a fixed time. ILC is a control design tool that fulfills the desired system response in the shortest time by closing the gap of traditional control methods.

Diagrams are illustrated in Figure 3.3 for a better understanding of the ILC working systematic.

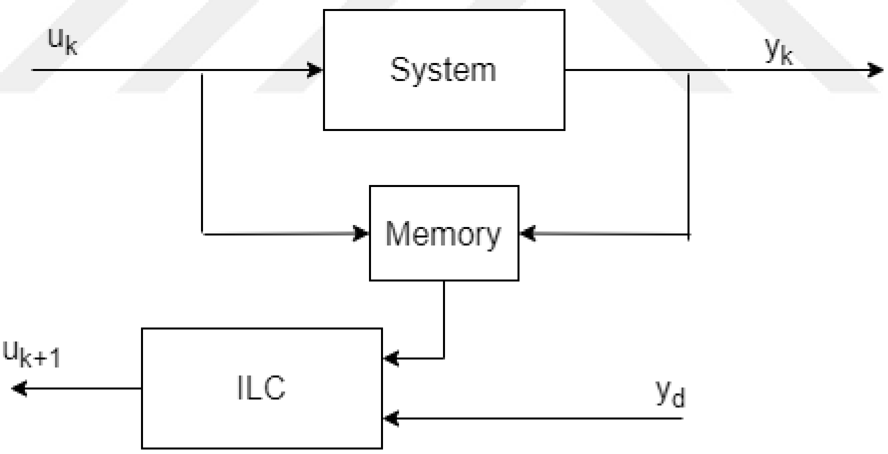


Figure 3.3. ILC configuration

The general formulation of continouos time iterative learning control system in state-space form is:

$$\dot{x}_k(t) = Ax_k(t) + Bu_k(t) \tag{3.1}$$

$$y_k(t) = Cx_k(t) \tag{3.2}$$

k is iteration, y_d is desired output, A, B and C are system matrices, u_k is control input, y_k is output signal. e_k is error signal of the system and it is calculated by difference between desired output of system $y_d(t)$ and system actual output value $y_k(t)$.

$$e_k(t) = y_d(t) - y_k(t) \tag{3.3}$$

The rules of ILC working algorithm can be listed as follows (Ahn et al., 2007).:

1. Each trial runs within a fixed time, including iteration, repetation, cycle.
2. The initial state $x_k(0)$ of system must be set to same point when the system is started for a new iteration.
3. System Dynamics must remain the same in every iteration.
4. The output measurement is done in a deterministic way.
5. The system dynamics are deterministic.

ILC is simply an intelligent control method that runs a certain task on a predefined system repeatedly over a fixed time and to get the desired output as soon as possible. While doing this, it works to realize the stable operation of the system as soon as possible with its memory and learning features. The output value observed by operating the system with keeping the initial value constant at each iteration tries to reach the desired point in every trial. The scheme of ILC theory is demonstrated in Figure 3.4.

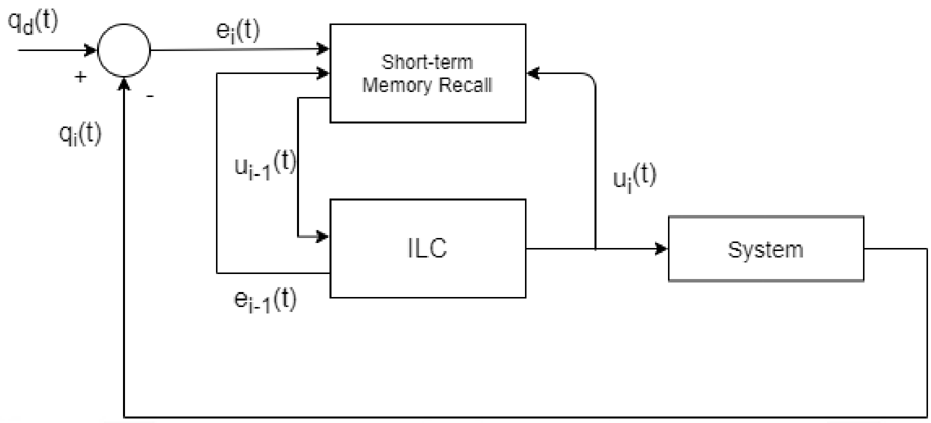


Figure 3.4. Scheme of ILC Theory

The memory of ILC stores the previous control value and error whis is difference value between input and output. Learning controller makes comparison between desired output value $y_d(t)$ and the system actual output value $y_k(t)$ and then error calculation is made, the new control signal of the sytem is created by looking at the previous values which were stored in memory. In this way, the reduction of the error value is observed as a result of each iteration. A decrease in the error value indicates that the actual output value of the system converges to the desired output value.

Tracking performance of system is improved by ILC method when system is operated repeatedly. The system is operated iteratively in a finite time interval. That iterative operation is called trial. The tracking error signal of system in the last trial is used to improve the control input signal. The generated control input signal from the last trial is used in the next trial. Eventually a better tracking performance of system is observed after each trial. This development process is shown in Figure 3.5.

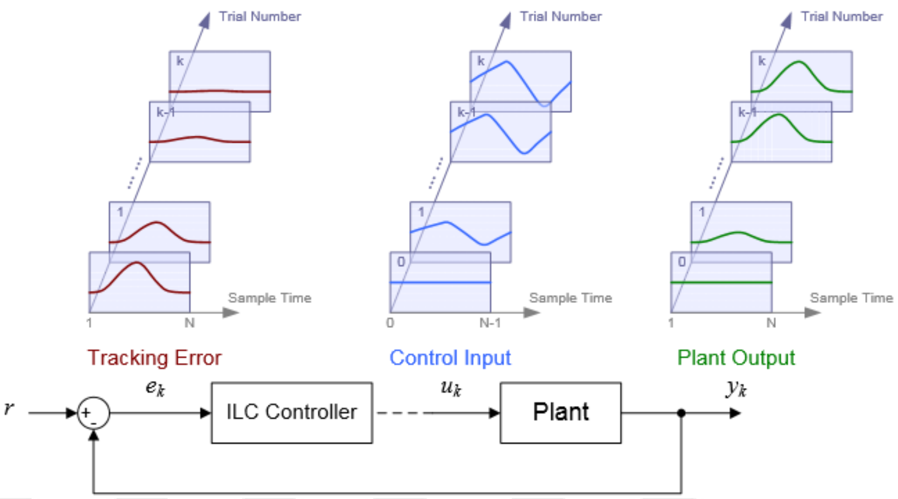


Figure 3.5. Diagram of ILC Tracking Development Process

3.3. PID-type Iterative Learning Control

ILC algorithms are used to improve conventional PID controller applications. Classical PID controllers have been used in many control studies and also combining modern control methods with PID has given better results. The main target of combination of ILC and PID is to reach a reliable conclusion by updating the PID parameters with iterative learning laws.

Iterative form of learning rules in ILC is expressed as:

$$u_{k+1} = F(u_k, e_k) \tag{3.4}$$

This expression shows that $e_k \to 0$ and $k \to \infty$.

Arimoto D-type iterative learning controller was the first PID based control structure (Chen and Moore, 2002). All developed PID-type ILC controller were created by advancing this structure. Arimoto D-type ILC structure diagram is shown in Figure 3.6.

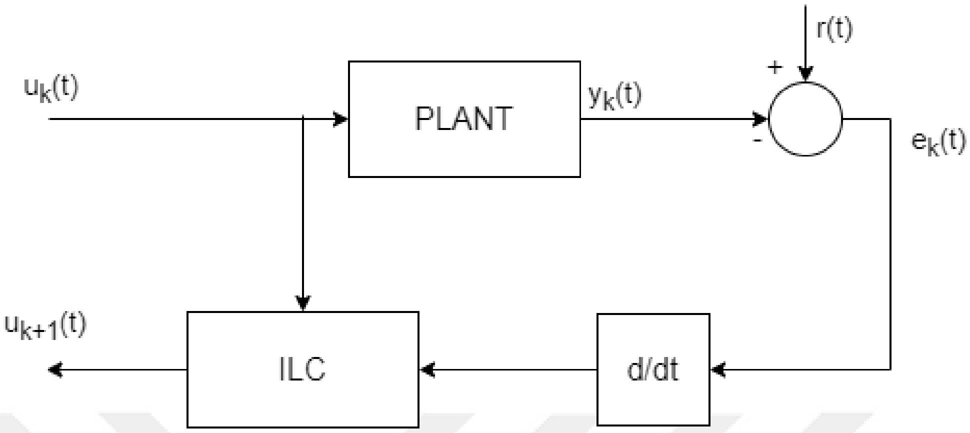


Figure 3.6. Arimoto D-type ILC

Arimoto D-type ILC scheme is formulated as:

$$u_{k+1}(t) = u_k(t) + \Gamma\dot{e}_k(t) \tag{3.5}$$

where Γ denotes gain, the more general PID-type ILC law formula has been developed by Arimoto is given as:

$$u_{k+1}(t) = u_k(t) + K_p e_k(t) + K_i \int e_k(\tau)d\tau + K_d \frac{d}{dt} e_k(t) \tag{3.6}$$

In this formula (3.6), $u_k(t)$ is the control signal, $e_k(t)$ is the tracking error which is calculated by $e_k(t) = y_d(t) - y_k(t)$ and iteration order is represented by k. K_p, K_i, K_d are of PID learning gain parameters respectively. The requirement is

$$\lim_{k\to\infty} y_k(t) \ \to y_d(t) \tag{3.7}$$

For all $t \in [0,\ T]$, if

$$\|1-\gamma h_1\| < 1 \tag{3.8}$$

where γ is learning gain, h_1 is the first Markov parameter, in the state space representation (*A*,*B*,*C*), h_1=*CB*$\neq$0 (Elshazly et al., 2012).

When using Φ, Ψ and Γ instead of K_p, K_i, K_d learning gains, Equation (3.6) is rewritten as:

$$u_{k+1} = u_k + \Phi e_k + \Psi \int e_k dt + \Gamma \dot{e}_k \tag{3.9}$$

The type of algorithm that used the data of more than one experiment before is called higher order ILC (HOILC) and its formulation is as follows (Chen and Wen, 1999):

$$u_{k+1} = \sum_{k=1}^{N}(I-\Lambda)P_K u_k + \Lambda u_0 + \sum_{k=1}^{N}\left(\Phi_k e_{i-k+1} + \Psi_k \int e_{i-k+1} dt + \Gamma_k \dot{e}_{i-k+1}\right) \tag{3.10}$$

where P_K is the learning matrix for the *k*th iteration and Λ ($0\leq \Lambda<1$) is a weighting parameter to restrain the large fluction of control input at beginning of ILC iteration.

In case of $\sum_{k=1}^{N} P_K = I$, e_k converges to zero asymptoticaly with proper selection of learning matrices.

The PID-type ILC controller design applicable for the systems is shown as a diagram in Figure 3.7.

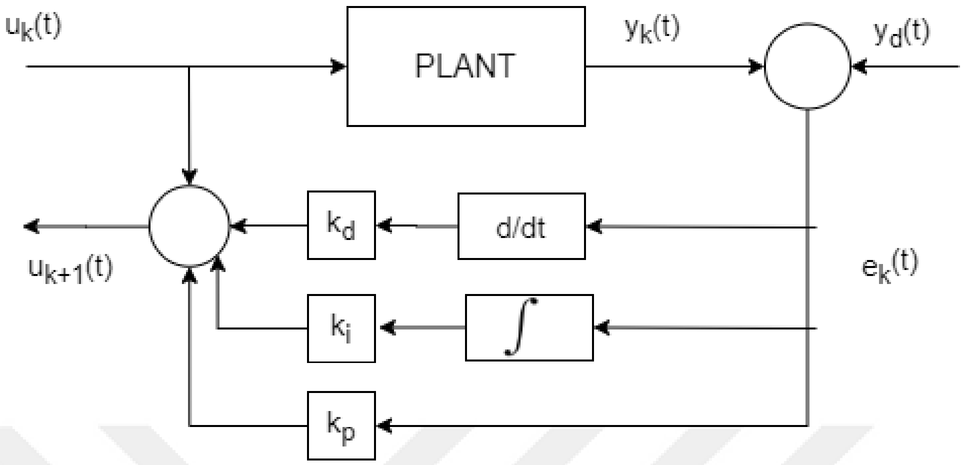


Figure 3.7. PID-type ILC

While designing PID-type ILC, the initial reference signal $r(t)$ is applied on a predefined system and the control input signal $u_k(t)$ and output signal $y_k(t)$ are observed. The ILC algorithm updates the PID parameters K_p, K_i, K_d to work stably by looking at the system error signal $e_k(t)$ repetitively. The memory block is used to keep the previous values of the system and use it in the next trials. The structure of PID-type ILC for Matlab-Simulink program is shown in Figure 3.8.

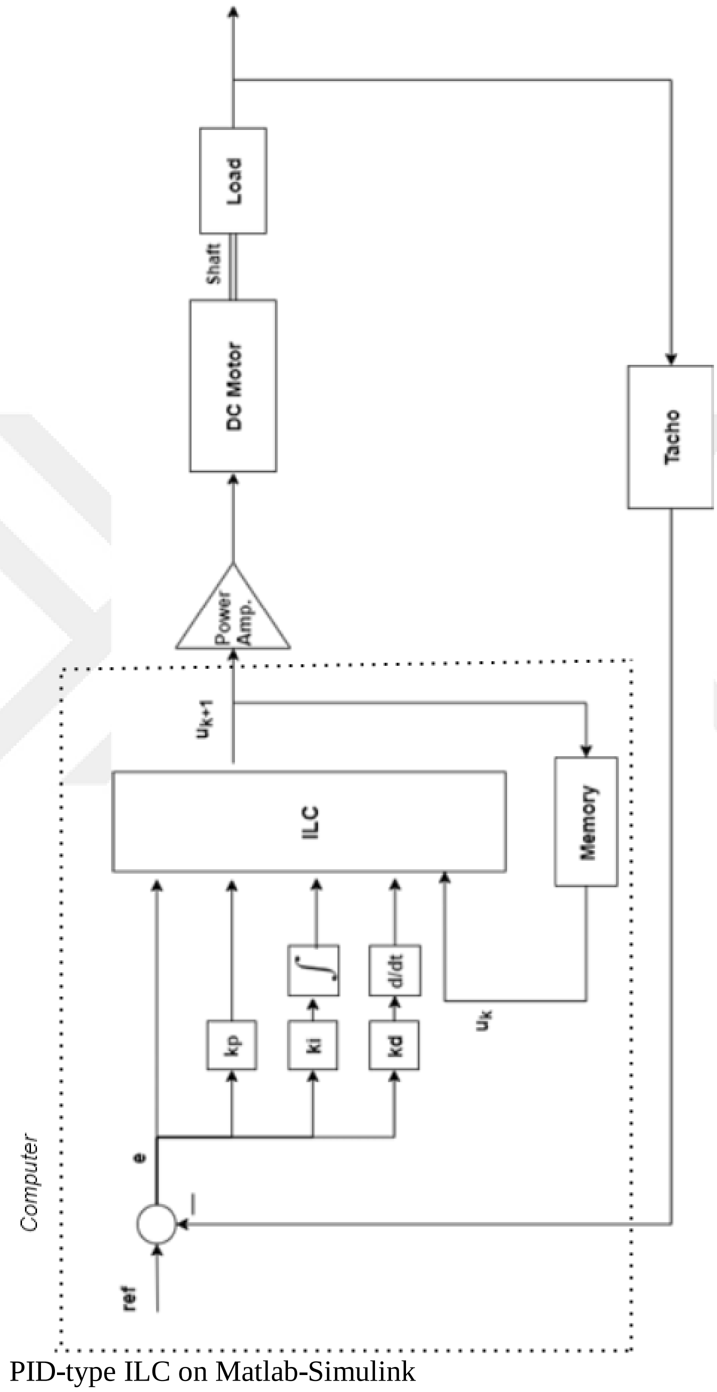


Figure 3.8. PID-type ILC on Matlab-Simulink

3.4. Fuzzy PID-type Iterative Learning Control

ILC has got a good performance for high precision tracking control. The development of this performance is improved to become more effective by several additions. Fuzzy logic control is another intelligent control method like ILC. Fuzzy control does not need a precise mathematical model of the system to be controlled (Wang et al., 2016). Fuzzy logic eliminates some uncertain effect in control system. With the conveniences provided by fuzzy logic, it provides tracking error and trajectory accuracy to reach the desired point faster. In order to improve the precision of system control, fuzzy logic is integrated into the PID-type ILC. The combination of fuzzy logic and ILC increases the effectiveness of the control on the system, and combining the ILC algorithm with the fuzzy logic creates a new control study structure.

The structure of fuzzy PID-type iterative learning controller is illustrated in Figure 3.9.

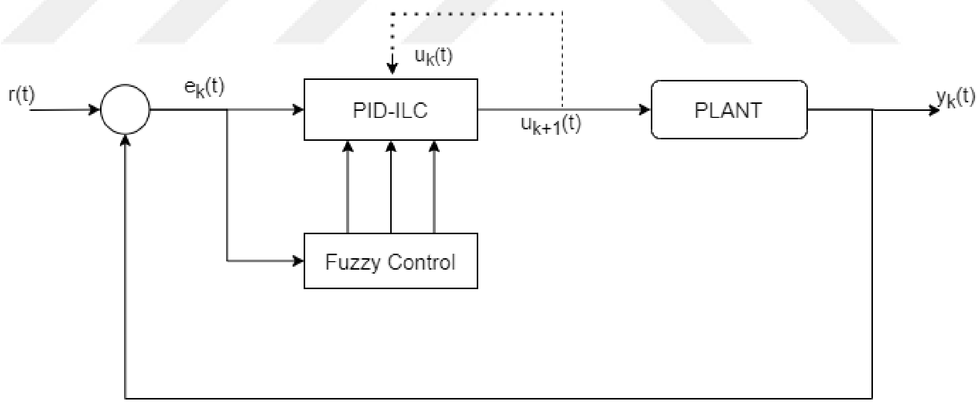


Figure 3.9. The structure of Fuzzy PID-type ILC

As it is seen in this control structure, fuzzy parameters K_p, K_i, K_d should be calculated first. In order for the fuzzy mechanism system to calculate these three parameters K_P, K_I, K_D, the system needs two other values; error e and change of error ec. The values of the parameters are determined by the calculations made in the fuzzy functions according to the values of these *e* and *ec*. Fuzzy control structure is shown in Figure 3.10.

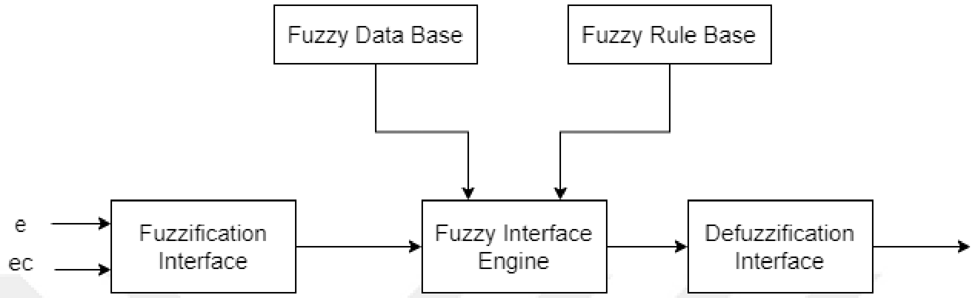


Figure 3.10. Fuzzy Control Structure

While adjusting the fuzzy parameters, traditional PID parameters are taken as reference, increasing the accuracy rate in the production of fuzzy values.

- Proportional gain $\boldsymbol{K_p}$

Proportional gain is used to increase the response speed of the system. Each increase in proportional gain affects the response speed of the system. If proportional gain is increased too much by itself, it causes overshoot in the signal of system response. This causes the system to destabilize. If proportional gain is selected small, the system response speed slows down and also slows down the system adjustment.

- Integral gain $\boldsymbol{K_i}$

Integral is an adjuster used to remove steady-state error of the system. According to the increase of the integral, the static error of the system is removed faster. Increasing the integral too much causes overshoot in the system response signal. Reducing the integral too much makes it difficult for the system to remove the steady-state error. ILC is a natural integrator, so a small value should be chosen when determining the integral value for this system.

- Differential gain $\boldsymbol{K_d}$

The effect of the derivative is to prevent a very rapid increase of the output response in the system. The derivative response is proportional to the rate of change of the process variable. It is used to control the movement of the system signal. It prevents deviation of system accuracy and prevent increase in adjustment time.

One of the most important parts when designing fuzzy PID-type ILC structure is the fuzzy control rule adjustment. To make fuzzy control rule, it is first necessary to define fuzzy inputs and fuzzy outputs. There are two fuzzy inputs and three fuzzy fuzzy outputs which are *e, ec*, K_p, K_i, K_d respectively. Error and change of error membership functions are used to derive fuzzy logic control rules. Fuzzy control rules are generated for K_p, K_i, K_d.These control rules are shown in Figure 3.11, Figure 3.12 and Figure 3.13 (Hao and Wang, 2017).

	NB	*NM*	*NS*	*ZO*	*PS*	*PM*	*PB*
NB	PB	PB	PM	PM	PS	ZO	ZO
NM	PB	PB	PM	PS	PS	ZO	NS
NS	PM	PM	PM	PS	ZO	NS	NS
ZO	PM	PM	PS	ZO	NS	NM	NM
PS	PS	PS	ZO	NS	NS	NM	NM
PM	PS	ZO	NS	NM	NM	NM	NB
PB	ZO	ZO	NM	NM	NM	NB	NB

Figure 3.11. Fuzzy rules for K_p

	NB	*NM*	*NS*	*ZO*	*PS*	*PM*	*PB*
NB	NB	NB	NM	NM	NS	ZO	ZO
NM	NB	NB	NM	NS	NS	ZO	ZO
NS	NM	NM	NS	NS	ZO	PS	PS
ZO	NM	NM	NS	ZO	PS	PM	PM
PS	NS	NS	ZO	PS	PS	PM	PB
PM	ZO	ZO	PS	PS	PM	PB	PB
PB	ZO	ZO	PS	PM	PM	PB	PB

Figure 3.12. Fuzzy rules for K_i

	NB	*NM*	*NS*	*ZO*	*PS*	*PM*	*PB*
NB	PS	NS	NB	NB	NB	NM	PS
NM	PS	NS	NB	NM	NM	NS	ZO
NS	ZO	NS	NM	NM	NS	NS	ZO
ZO	ZO	NS	NS	NS	NS	NS	ZO
PS	ZO	ZO	ZO	ZO	ZO	ZO	ZO
PM	PB	NS	PS	PS	PS	PS	PB
PB	PB	PM	PM	PM	PS	PS	PB

Figure 3.13. Fuzzy rules for K_d

Fuzzy rules are created with using fuzzy membership functions, these fuzzy rules generate fuzzy output values and increase the system's accuracy in the desired trajectory. Calculation of PID parameters coefficents are expressed as:

$$\begin{aligned} K_p &= FS\,[e(t), \dot{e}(t)]\,, \\ K_i &= FS\,[e(t), \dot{e}(t)], \\ K_d &= FS\,[e(t), \dot{e}(t)] \end{aligned} \tag{3.11}$$

PID coefficients are calculated from fuzzy mechanism:

$$K_p = \frac{\sum_{i=1}^{n} \mu_{kp}(\Delta k_P)\Delta k_P}{\sum_{i=1}^{n} \mu_{kp}(\Delta k_P)} \tag{3.12}$$

$$K_i = \frac{\sum_{i=1}^{n} \mu_{ki}(\Delta k_i)\Delta k_i}{\sum_{i=1}^{n} \mu_{ki}(\Delta k_i)} \tag{3.13}$$

$$K_d = \frac{\sum_{i=1}^{n} \mu_{kd}(\Delta k_d)k_d}{\sum_{i=1}^{n} \mu_{kd}(k_d)} \tag{3.14}$$

The output values created from the fuzzy mechanism are included in the system with ILC law, and fuzzy iterative learning law occurs. The new fuzzy PID-type iterative learning algorithm created by putting these fuzzy values into the ILC algorithm is formulated as follows:

$$u_{k+1}(t) = u_k(t) + K_p e_{k+1}(t) + K_i \int e_{k+1}(\tau) d\tau + K_d \frac{d}{dt} e_k(t) \qquad (3.15)$$

The structure of fuzzy PID-type iterative learning controller for Matlab/Simulink in the Figure 3.14.

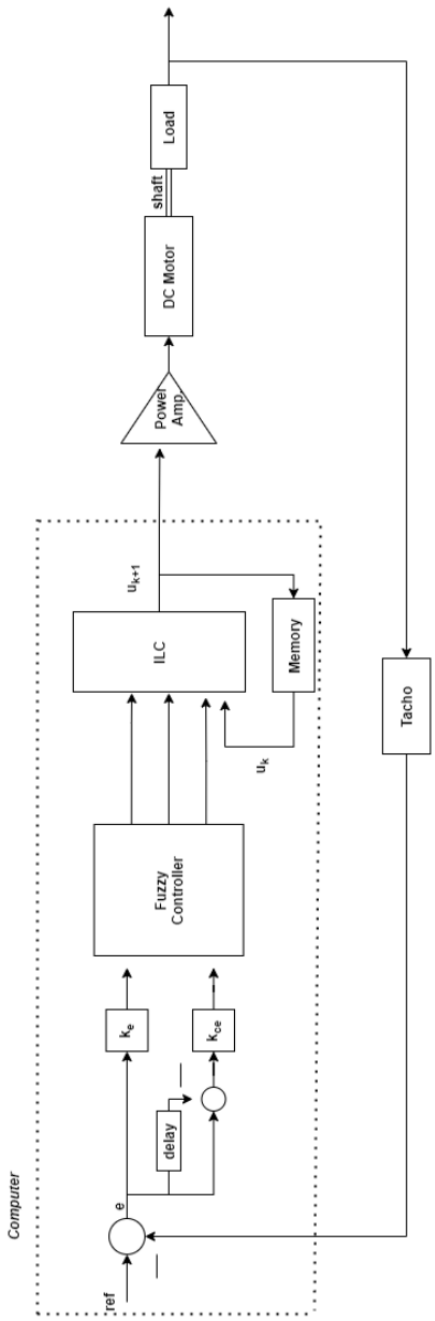


Figure 3.14. The structure of adaptive fuzzy PID-type ILC

3.5. Adaptive Fuzzy PID-type Iterative Learning Control

Adaptive fuzzy PID-type iterative learning control is basically combination of adaptive theory and fuzzy PID-type iterative learning control structures. Adaptive control is one of the modern control branches. The purpose of adaptive control is to help arranging to PID parameters of auto-tuning control architecture comparing with conventional PID control that parameters are predetermined. It is widely used in the industrial applications. Combination with other control strategies has provided advantages. Fuzzy logic controller is a frequently used method due to its simple and inexpensive control structure. On the other hand fuzzy logic control may not exhibit a perfect control performance if the system to be controlled is highly nonlinear and uncertain. There are many applications where adaptive fuzzy PID-type controller performs better than fuzzy controller (Wang, 1994; Lin and Yang, 2003; Wai et al., 2004; Jee and Koren, 2004; Woo et al., 2000; Verma and Chauhan, 2019; Tong et al., 2020). Various adaptive fuzzy control techniques are as follows (Liang et al.):

- Membership function tunning
- Input or output scaling tunning
- Linguistic rule tunning

Scale tunning is more preferred because of its ease of application to the control system mechanism and its being more effective. Combination of adaptive and fuzzy logic has helped to improve the control action for the plant operation (Wang et al., 2019; Yu et al., 2020). The adaptive fuzzy part of the control mechanism is referenced (Eker and Torun, 2006) and adaptive fuzzy PID algorithm is shown in Figure 3.15.

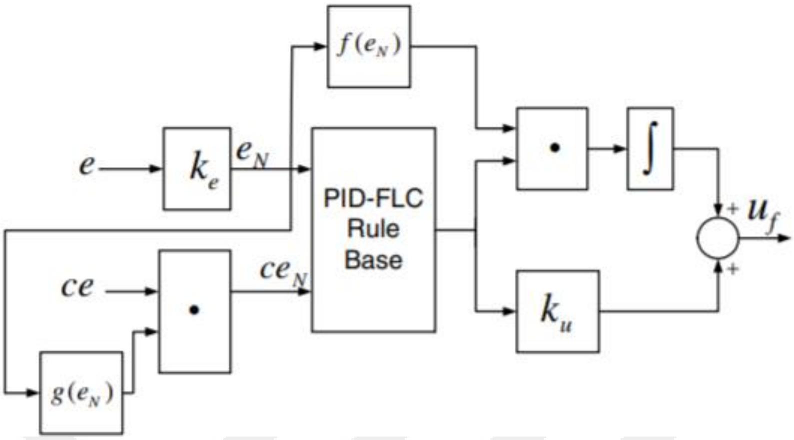


Figure 3.15. Adaptive Fuzzy PID Algorithm

The adaptive fuzzy PID algorithm is based on adaptation of scaling factors (Woo et al.,2000). The adaptation of scaling factors is made by two algorithm formulas, $f(e_N)$ and $g(e_N)$.

$$f(e_N(k)) = a_1(a_2 + a_3 abs(e_N(k))) \tag{3.12}$$

$$g(e_N(k)) = b_1(b_2 - b_3 abs(e_N(k))) \tag{3.13}$$

Scaling factors are arranged by these two adaptations formulas and integration of adaptive and fuzzy logic is completed successfully. After adaptation of scaling factors, fuzzy logic subsystems are arranged; membership functions, defuzzification, fuzzification, rules ect. PID parameters' gains K_p, K_i, K_d calculations are computed with these systems.

Iterative learning control system is based on to use the control signal which is produced in previous trial for next trial. The new control input u_{k+1} is produced by information of previous used control signal u_k. Theorically, the adaptive fuzzy PID-type ILC system is created by adding ILC part to the prepared adaptive fuzzy

PID controller. This adaptive fuzzy PID-type ILC algorithm is illustrated in Figure 3.16.

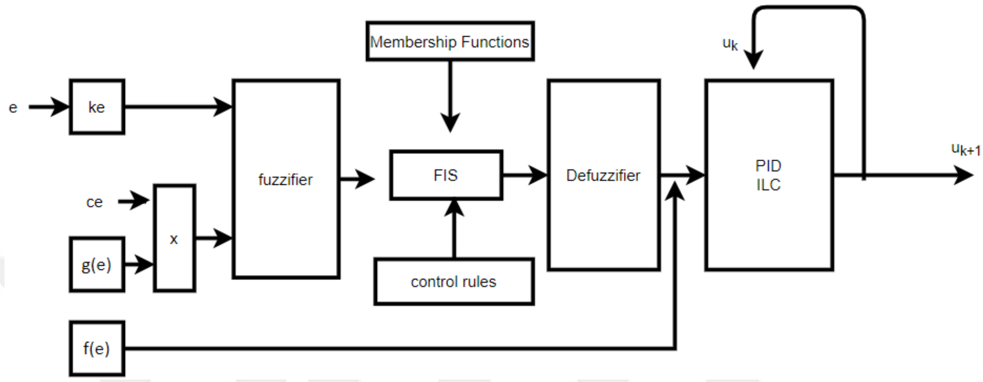


Figure 3.16. Adaptive Fuzzy PID-type ILC algorithm

As seen in the diagram above, the control signal is generated from the adaptive fuzzy PID-type ILC system and its adjustment is made for the plant to be applied. Adaptive Fuzzy PID-type ILC control system is builded with using MATLAB/Simulink program. The structure of adaptive fuzzy PID-type ILC for the Matlab / Simulink is shown in the Figure 3.17.

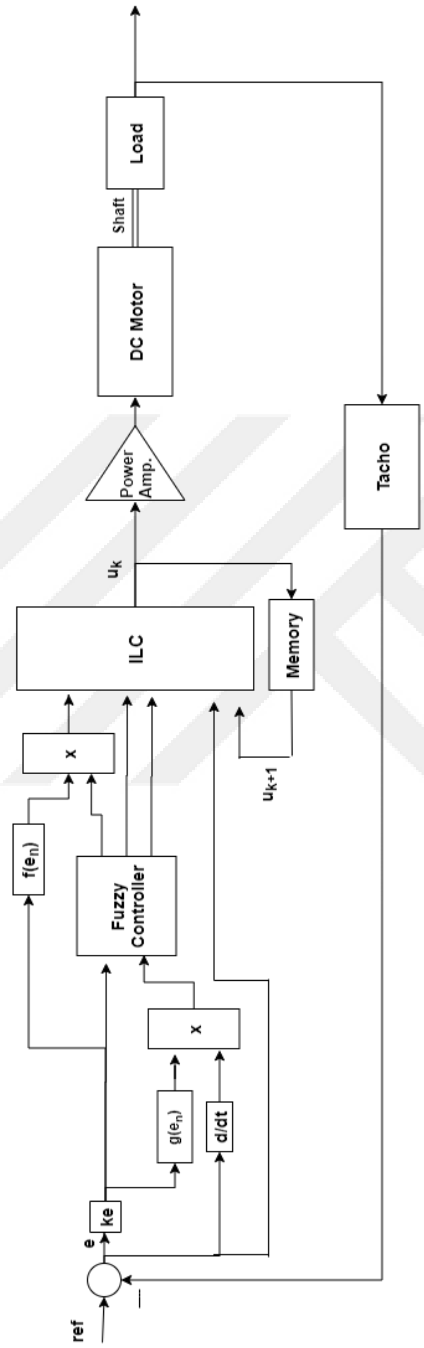


Figure 3.17. The structure of adaptive fuzzy PID-type ILC control system

3.6. Stability Analysis of Iterative Learning Control

System state space model is written as follow:

$$\dot{x}_k(t) = Ax_k(t) + Bu_k(t),$$
$$y_k(t) = Cx_k(t) \tag{3.14}$$

where $x_k(t)$ is *n x 1* state vector, $y_k(t)$ is *m x 1* output vector, $u_k(t)$ is *l x 1* control input vector and $t \in [0, \mathrm{T}]$. The error is calculated by $e_k(t) = r(t) - y_k(t)$. The control input is formulated with iteration order *k+1* as:

$$u_{k+1}(t) = \sum_{i=1}^{n} \alpha_i u_{k+1-i}(t) + \sum_{i=1}^{n} K_i e_{k+1-i}(t) + K_0 e_{k+1}(t) \tag{3.15}$$

Memory is represented by *N* and α_i is static scalar, $1 \le i \le N$, linear operator K_i.

At first, the open loop system can be expressed as follows:

$$e_k(t) = r(t) - G[u_{k+1}](t) \tag{3.16}$$

And

$$G[u_k](t) = C\int_0^t e^{A(t-\tau)} Bu(\tau)\, d\tau \tag{3.17}$$

The closed loop system error is expressed with using above expressions as:

$$e_{k+1}(t) = (I + GK_0)^{-1}\left\{\sum_{i=1}^{N} (\alpha_i I + GK_0)\,[e_{k+1-i}](t) + \left(1 - \sum_{i=1}^{N} \alpha_i\right) r(t)\right\} \tag{3.18}$$

or like that:

$$\hat{e}_{k+1} = L_T \hat{e}_k + b \tag{3.19}$$

where $\hat{e}$ denotes error and $\hat{e}_k$ is described as:

$$\hat{e}_{k+1} = [e_{k+1-N}^T(t), \ldots\ldots, e_k^T(t)] \tag{3.20}$$

and,

$$L^T = \begin{bmatrix} 0 & I & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \ldots & 0 & I \\ E_0 E_N & \ldots & E_0 E_2 & E_0 E_1 \end{bmatrix} \tag{3.21}$$

$$E_0[y](t) = (I + GK_0)^{-1}[y](t) \tag{3.22}$$

$$E_i[y](t) = (\alpha_i I + GK_i)^{-1}[y](t) \tag{3.23}$$

where $1 \leq i \leq N$, and

$$b = [0\ 0 \ldots\ (1 - \textstyle\sum_{i=1}^{n} \alpha_i) r^T(t)]^T \tag{3.24}$$

They are determined as E_T is a Banach space and $\hat{e}_k \in E_T$, W_T is a linear subspace of E_T and $b \in W_T$, $L_T \hat{e}_k$ represents the contributions of error from past trials to current trials, and b is the disturbance vector and shows the contribution of external sources on the current trial (Owens et al., 2000).

In closed loop ILC systems, iteration number k converges to infinity theorically, it provides stability of linear repetitive systems.

The first definition of stability is given below.

- Definition 1: It is considered that ILC scheme in form (3.19) is asymptotically stable whether real scalar δ> 0 and $\{\hat{e}_k\}_{k\geq 1}$ for any initial error $\hat{e}_0$

$$\hat{e}_{k+1} = (L_T + \gamma)\hat{e}_k + b \tag{3.25}$$

It converges highly to a limit error $\hat{e}_\infty \in E_T$ when $||\gamma|| \leq \delta$. The behavior of ILC scheme causes the non-zero error value to decrease and converges to zero.

When the linear repetitive process stability theory is used to point out that definition 1 is the same as the existance of finite real scalar $M_T > 0$, λ ∈ (0, 1),

$$||L_T^k|| \leq M_T \lambda_T^k \tag{3.26}$$

In this expression (3.24), $k \geq 0$ and the spectral radius $r(L_T) < 1$, and therefore the limit error is

$$\hat{e}_\infty = (I + L_T)^{-1} b \tag{3.27}$$

In the statements given, the error values of closed loop control systems are given in (3.16), and $E_0 = L_p^m[0, T]$. The spectral values of the linear operator L_T are figured out for next results. It was used to provide the asymptotic stability calculation required for the linear repetitive system (Rogers and Owens, 1992).

- Theorem 1: The first rule of convergence of closed loop ILC system is that all roots' modulus should be less than unity.

$$z^N - \alpha_1 z^{N-1} - \ldots\ldots - \alpha_{N-1} z - \alpha_N = 0 \tag{3.28}$$

In this theorem, the closed-loop ILC error converges with $L_p[0,T]$,

$$e_\infty = \left(I + GK_{eff}\right)^{-1} r \tag{3.29}$$

The term K_{eff} is called effective controller and it is expressed as:

$$K_{eff} = \frac{K}{1-\beta} \tag{3.30}$$

And

$$\beta = \sum_{i=1}^{N} \alpha_i, K = \sum_{i=0}^{N} K_i \tag{3.31}$$

To define Equation (3.29), the variables in Equation (3.18) must be replaced with their limits and they must be adjusted again.

- Theorem 2: Assuming that theorem 1 is kept and

$$\|\hat{e}_k - \hat{e}_\infty\| \leq M_1(max(\|e_\infty\|, \dots, \|e_{N-1}\|) + M_1)\lambda_e^k \tag{3.32}$$

This (3.32) makes the error sequence bounded with conditions which M_1 and M_2 are real positive scalars and $\lambda_e \in$ (max $|\mu_i|$, 1).

- Proof: The closed-loop solution of Equation (3.19) is given as:

$$\hat{e}_k = L_T^k \hat{e}_0 + \sum_{j=1}^{k} L_T^{j-1} b \tag{3.33}$$

and $\lim_{k\to\infty} L_T^k \hat{e}_0 = 0$ so that the limit error is written as:

$$\hat{e}_\infty = \sum_{j=1}^{\infty} L_T^{j-1}\, b \tag{3.34}$$

Therefore

$$\hat{e}_k - \hat{e}_\infty = L_T^k \hat{e}_0 + \sum_{j=k+1}^{\infty} L_T^{j-1}\, b \tag{3.35}$$

In addition to these, $r(L_T) = \lim_{k\to\infty} \left\|L_T^k\right\|^{\frac{1}{k}} < 1$ and $\exists M_e > 0$, $\lambda_e \in (O,1)$ so that $\left\|L_T^k\right\| \le M_e\lambda_e^k$. In this way, these cause the following expression to ocur

$$\|\hat{e}_k - \hat{e}_\infty\| \le \frac{M_e\lambda_e^k}{1-\lambda_e}\|b\| + M_e\lambda_e^k\|\hat{e}_0\| \tag{3.36}$$

In theorem 1, it can be said that the stability is intuitive independent of the facility or the controller. It can be classified as acceptable due to the fact that the trial period is limited to T and the period limitations of the outputs generated by a linear system in this period.

Theorem 1 and 2 are useful for choosing important parameters and their characteristics are as follows:

1. In order for convergence to be accepted fast, λ_e must be small and geometrically converged as λ_e^k.
2. Although the limit error value is not zero, the effective control K_{eff} feedback system defined in Equation (3.28) and Equation (3.29)

indicates that this is useful. If max$|\mu_i|$ converges to zero from right hand side, the limit error is actually the first step of learning iteration.

3. If and only if b=0 and $\sum_{i=1}^{N} \alpha_i = 0$, it is possible for the limit error to be zero. However, this is not possible in case of $r(L_T) < 1$, it is possible in case of $r(L_T) = 1$. This circumstence is similar to the classical control method which integral operation is included. This causes a zero steady state limit error in response to constant reference signals.

Looking at the results given, there is an impact between the system and control structure from both a theoretical and a practical perspective. Especially, in the case of $k_i = 0$, $0 \leq i \leq N$. This GK_0 must show stable properties under these conditions.

3.7. Stability Analysis of Fuzzy Iterative Learning Control

The evaluation of nonlinear function structures of fuzzy controllers with Dirichlet's condition is as follows (Oppenheim et al.,1997):

$$u = N(e) = \sum_{\lambda=0}^{\infty} a_{\lambda+1} \sin[(\lambda+1)\frac{\pi}{\theta} e] \tag{3.37}$$

where e is control error, λ is iteration number and a is a linear part of fuzzy tunning parameters.

Writing Equation (3.37) in matrix form:

$$\begin{aligned} &z = \sigma b, b = [b_1\ b_2\ \dots\ b_{\lambda+1}]^T, z = [z_1\ z_2\ \dots\ z_{\lambda+1}]^T, \\ &\sigma = \left[\sin(2i+1)\frac{\pi}{\theta} e_j\right]_{1,j=\overline{1,\lambda+1}}, \\ &e = j.h_e, j = \overline{1, \lambda+1} \end{aligned} \tag{3.38}$$

where h_e > 0, σ is regular matrix, also b and z represent specific vector of mean nonlinearity and nonlinear vector respectively in these equations. The following equations are for the cases fuzzy controller expressions are accepted:

$$I_R b = n, I_R = \left[2J_1\{(2j-1)\frac{\pi}{\theta}A_i\}/A_i\right]_{1,j=\overline{1,\lambda+1}},$$
$$n = [n(A_1)\ n(A_2) \dots\ n(A_{\lambda+1})]^T, e = A_i \sin(\omega_t), i = \overline{1,\lambda+1} \tag{3.39}$$

where n, J_1 and A_i represent gain vector, the Bessel function and magnitude of input respectively. These statements constitute:

$$Hz = n, H = I_R\sigma^{-1} \tag{3.40}$$

Equation (3.40) explains to be linear using the regular matrix H. In this way, it may be expressed as convenient linearization. The matrix plane approximation of the closed loop fuzzy control system is expressed in (3.41).

$$a_n s^n + a_{n-1}s^{n-1} + \cdots + a_1 s + a_0 = 0 \tag{3.41}$$

The expression given in (3.41) is acceptable and the coefficients in the expression are calculated according to the parameters of the linear part and the equivalent gains of the non-linear part. The matrix expression in (3.42) was created based on frequencies:

$$\Omega = [(\sin(k\pi/2) + (\cos(k\pi/2))w_r^k]_{r=\overline{1,n},k=\overline{0,p}}$$

$$P_0 = \begin{bmatrix} a_0(n(A_1)) & 0 & a_2(n(A_1)) & 0 & a_4(n(A_1)) & \dots \\ a_0(n(A_2)) & 0 & a_2(n(A_2)) & 0 & a_4(n(A_2)) & \dots \\ \dots & \dots & \dots & \dots & \dots & 0 \end{bmatrix}^T,$$

$$Q_0 = \begin{bmatrix} 0 & a_1(n(A_1)) & 0 & a_3(n(A_1)) & 0 & \dots \\ 0 & a_1(n(A_2)) & 0 & a_3(n(A_2)) & 0 & \dots \\ \dots & \dots & \dots & \dots & \dots & 0 \end{bmatrix}^T, \tag{3.42}$$

$$P_\sigma = \begin{bmatrix} \pi_0 & -\pi_1' & \pi_2 & -\pi_3' & \pi_4 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & 0 \end{bmatrix}^T,$$

$$Q_\sigma = \begin{bmatrix} \pi_0' & \pi_1 & \pi_2' & \pi_3 & \pi_4' & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & 0 \end{bmatrix}^T,$$

where P_0, Q_0, P_σ and Q_σ represent steady state regime and transients respectively. Also π_k and $\pi_k{}'$ in expressions are polynomials and $\sigma < 0$, from $s = \sigma + iw$. M matrix describes the matrix plane with double elements:

$$M = \begin{bmatrix} c_{11} & c_{12} & \dots & c_{1q} \\ d_{11} & d_{12} & \dots & d_{1q} \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \\ c_{m1} & c_{m2} & \dots & c_{mq} \\ d_{m1} & d_{m2} & \dots & d_{mq} \end{bmatrix} \tag{3.43}$$

The *ΩP* and *ΩQ* are valid in steady-state and transient cases as:

$$\Omega P = \left[c_{ij}\right]_{i=\overline{1,m},j=\overline{1,q}}, \Omega P = \left[d_{ij}\right]_{i=\overline{1,m},j=\overline{1,q}}. \tag{3.44}$$

In addition, the expressions of two step-type curve are given as:

$$c_{\rho\eta} < c < c_{(\rho+1)\varepsilon},\ \ d_{\lambda\beta} < d < d_{(\lambda+1)\beta},\ \ \rho,\gamma = \overline{1, m-1},\ \ \varepsilon,\beta = \overline{1, q-1} \tag{3.45}$$

In the case of $c = d = 0$, the intersection curves are known as the coincidence point in the matrix plane.

To sum up, stability analysis of iterative learning based fuzzy control systems is expressed as follows (Precup et al., 2008; Preitl et al., 2008):

- Determining the expression of the linearized characteristics of the fuzzy control system.
- Calculation of matrix M given in (3.43) and determination of step-type curves given in (3.45)
- Adjusting limit cycles by calculating the intersection of step type curves. If there are no limit cycles, the control system is already stable. But if there is cycle, continue with the next statement
- In case of having limit cycle, the periodic solution of control system is considered and solution approaches $e = A_0\, sin(w_0 t)$. Stability of limit cycle is possible by placing a small enough σ value of the coincidence point on the matrix plane with transient magnitude larger than the A_0 which is value of the limit cycle magnitude.

3.8. Summary

In this chapter, iterative learning control theory is explained and its areas of use are specified. PID-type iterative learning contol structure is designed. Fuzzy PID-type iterative learning control structure was developed by combining fuzzy logic control and iterative learning control structures. Adaptive algorithms are included in the fuzzy PID-type iterative learning control structure. Stability analysis is mentioned.

4. EXPERIMENTAL SETUP AND PRELIMINARIES

This section introduces the necessary experimental setup for real-time implementation of the proposed and previously described control methods. The experimental setup components are shown in Figure 4.1.

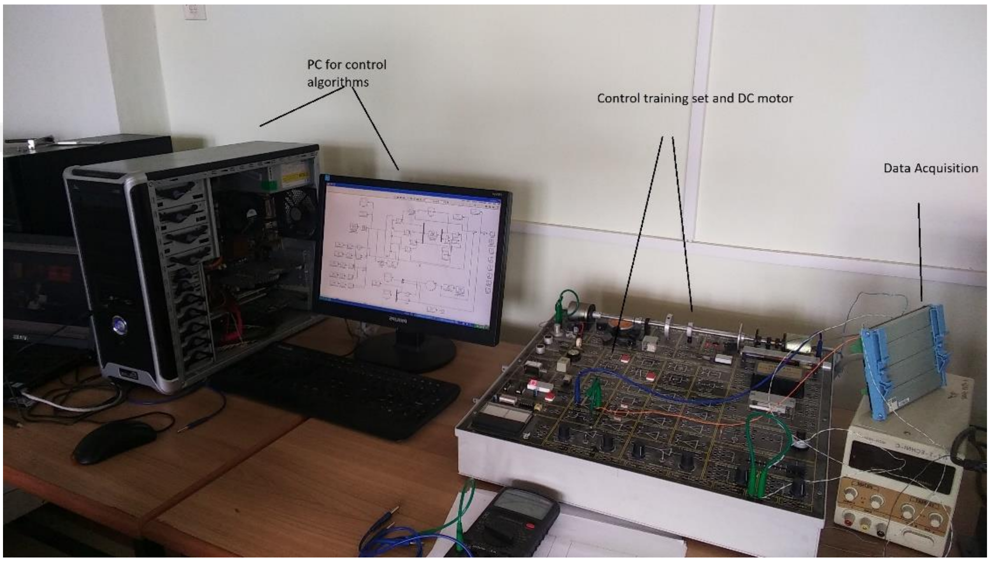


Figure 4.1. A scene from experimental set-up

4.1. Data Acquisition for Electromechanical System

The purpose of data acquisition is to read the data in the system such as voltage, current, pressure, temperature. Advantech PCI-1716 DAQ card model was used in the present study. The properties of Advantech PCI-1716 DAQ card are given in Table 4.1.

Table 4.1. The properties of Advantech PCI-1716 DAQ card

Analog Input	Channels	16 SE/Diff
	Sampling Rates	250 kS/s
	Resolution	16 bits
	Unipolar Input (V)	0~10,0~5,0~0.25,0~1.25
	Bipolar Inputs	±10,5,2.5,1.25,0.65
Analog Output	Channels	2
	Resolution	16 bits
Digital I/O	Digital Input Channels	16
	Digital Output Channels	16
General	Bus	PCI
Timer/Counter	Resolution	16 bits
	Time Base	10 Mhz
	Channels	1

The experimental setup is connected to a computer via DAQ (data aquisition) card for real time applications. There are 16 analogue inputs, 4 analogue outputs, 48 digital inputs and ouputs, 250Ks/s-16 bit maximum sampling rate and resolution in data aquisition (DAQ) card properties. Real-time window is used on Matlab/Simulink sofware programme for experiment. This mechanism is shown as a diagram in Figure 4.2.

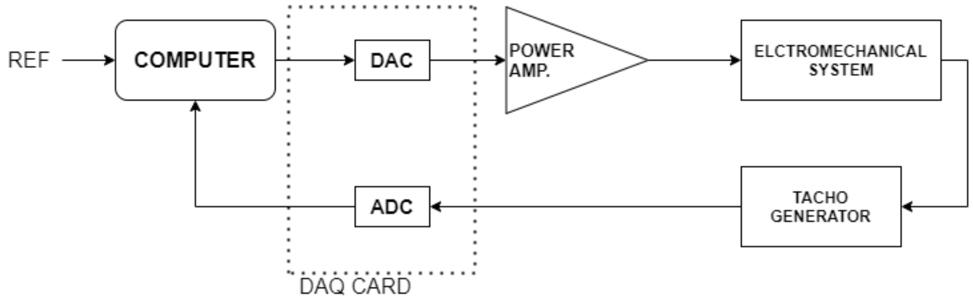


Figure 4.2. Diagram of experimental setup

4.2. Electromechanical System

Electromechanical system modeling and controls have an important place because of their wide application areas. The use of direct current (DC) electric motors is involved in many different industrial applications. There are many advantages of DC motors which are its simple structure, speed regulation, easy maintenance etc. Because of these reasons, many academic researches and studies are carried out on DC motors such as; control application for velocity and position, tracking performance, set point, energy efficiency, robustness.

Generally DC motor control applications focus on speed and angular position issues. Therefore, DC motorsa are designed to measusre velocity containing high frequency noice. The low-pass filter is used into feedback system to eliminate the noices. In this way, the use of filter can cause an increase in energy consumption, complexity and computation of control system mechanism.

The DC motor electromechnical system is illustrated in Figure 4.3.

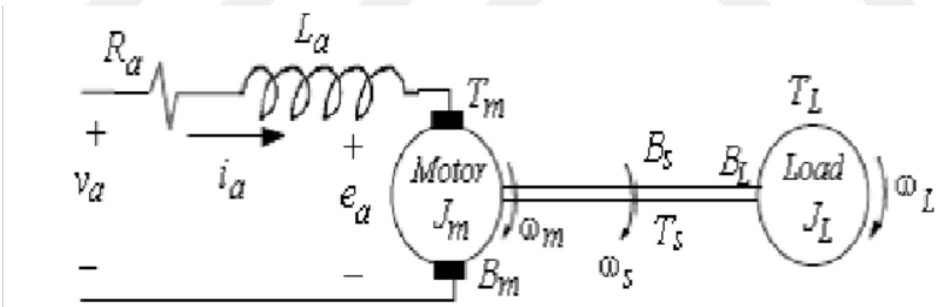


Figure 4.3. Diagram of DC motor electromechanical system

The dynamical equations of the electromechanical system are given in (Eker, 2008; Eker, 2004; Furat and Eker, 2014). The DC motor electromechanical equations are described as:

$$V_a(t) = L_a \frac{d}{dt} i_a(t) + R_a i_a(t) + K_m \omega_m(t) \quad (4.1)$$

$$J_m\left(\frac{d}{dt}\omega_m(t)\right) = T_m(t) - T_s(t) = R_m\omega_m(t) - T_f(\omega_m) \tag{4.2}$$

$$J_L\left(\frac{d}{dt}\omega_L(t)\right) = T_s(t) - R_L\omega_L(t) - T_d(t) - T_f(\omega_L) \tag{4.3}$$

$$T_s(t) = K_s\big(\theta_m(t) - \theta_L(t)\big) - B_m\big(\omega_m(t) - \omega_L(t)\big) \tag{4.4}$$

$$\frac{d}{dt}\theta_m(t) = \omega_m(t),\ \frac{d}{dt}\theta_L(t) = \omega_L(t) \tag{4.5}$$

All parameters of DC motor which are used in expressions are explained in Table 4.2.

Table 4.2. DC Motor Parameters

$\boldsymbol{V_a}$	Armature Voltage	$\boldsymbol{R_m, R_L}$	Vicous Friction Coefficient
$\boldsymbol{L_a}$	Armature Inductance	$\boldsymbol{K_m}$	Torque Coefficient
$\boldsymbol{R_a}$	Armature Resistance	$\boldsymbol{T_m}$	General Motor Torque
$\boldsymbol{i_a}$	Armature Current	$\boldsymbol{T_d}$	External Load Disturbance
$\boldsymbol{T_s}$	Nonlinear Friction	$\boldsymbol{T_f}$	Transmitted Shift Torque
$\boldsymbol{J_m, J_l}$	Moments of Inertia	$\boldsymbol{\omega_M, \omega_L}$	Rotational Speed

The block diagram of DC motor electromechanical system is shown in Figure 4.4. The DC motor consists of varios parts which have different duties in operation. There are several disks in motor shaft and these disks operate different kinds of transducers. Slotted-opto transducer is used to achieve shaft speed. The tachogenerator is used for speed transducer in control application. This tachogenerator operation is the voltage production which is proportional to the shaft speed.

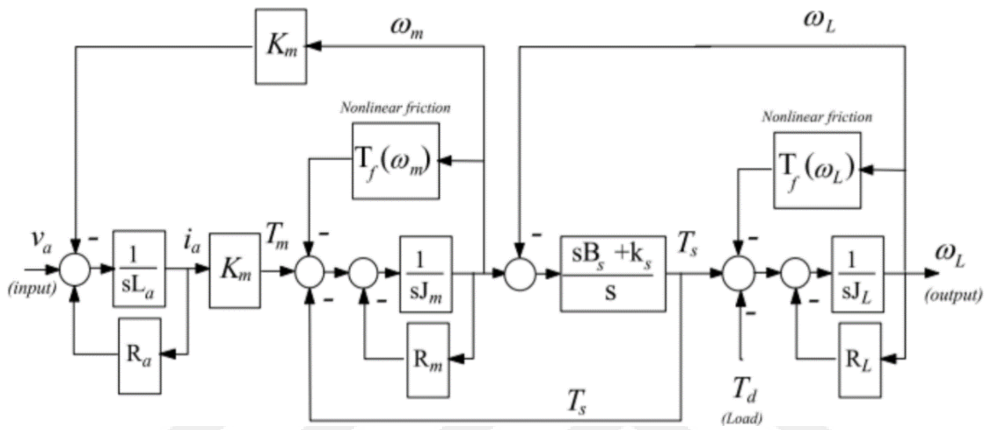


Figure 4.4. DC motor electromechanical system

It is difficult to obtain precise numerical values of parameters when modeling. Due to these difficulties, the input and output values of dynamic structures are observed for system identification and parameter values are found.

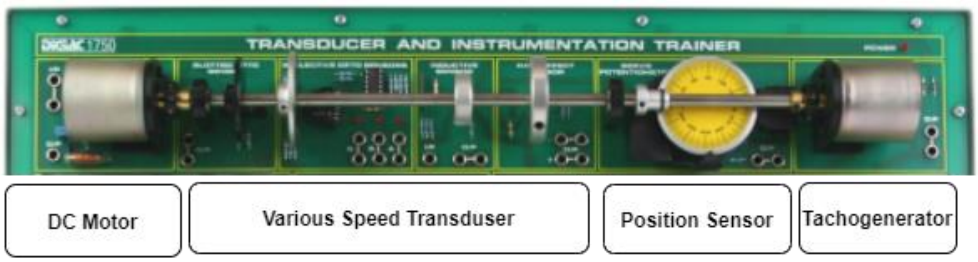


Figure 4.5. Electromechanical system

The DC motor used in the experiment is located on the DIGIAC 1750 control training set and it is shown in Figure 4.5. The slotted-opto transducer is an element that measures the shaft speed during the test period. The tachogenerator produces voltage in proportion to the shaft speed.

The voltage values corresponding to the different shaft speeds of the DC motor in the tachogenerator are as follows:

- 600 rpm, 2.37 V
- 800 rpm, 3.03 V
- 900 rpm, 3.39 V
- 1000 rpm, 3.7 V
- 1200 rpm, 4.48 V
- 1500 rpm, 5.4 V
- 2000 rpm, 7 V
- 2500 rpm, 7.8 V

4.2.1. Process Reaction Curves

The process reaction curve is a widely used method for determining the characteristics of a system. With this method, the rise time settling time and time constant of dynamic systems are determined (Eker, 2004). In system modeling, the armature of the dc motor is induced with step input. The speed of the motor shaft is measured in revolution per minute (rpm). When the output voltage generated from the tachogenerator is measured, 4.48 volts is obtained and this measured voltage corresponds to a speed of 1200 rpm at the motor shaft. The second order system modelling is as (Furat and Eker, 2015; Furat and Eker, 2014):

$$G(s) = \frac{K}{(1+T_d s)(1+T_p s)} = \frac{c}{s^2+as+b} \tag{4.6}$$

The plant coefficient are obtained by calculating from system output and $K = 0.823$, $T_d = 0.0087$, $T_p = 0.1418$ (Ozbek and Eker, 2019). The second order model expression is as:

$$\ddot{y}_m(t) = -a\dot{y}_m(t) - by_m(t) + cu(t) + d(t) \tag{4.7}$$

The parameters are nominal system parameters and disturbance: $a = a_n$, $b = b_n$, $c = c_n$ and $d(t)$. The parameters of nominal system are calculated as $a_n = 118.1763$, $b_n = 783.5862$, $c_n = 663.4968$ (Eker, 2006; Furat and Eker, 2014; Ozbek and Eker, 2019; Ozbek and Eker, 2015).

Fuzzy PID-type ILC controller implementation applied to the electromechanical system that is DC motor. Fuzzy PID-type ILC is implemented in Matlab because Matlab/Simulink allows computer based control applications. The error $e(k)$ and change of error $ce(k)$ are used as inputs fort he fuzzy logic mechanism. PID parameters coefficients are calculated by fuzy logic system. The produced signals from fuzzy logic are used into the ILC part. The control signal u_{k+1} is generated by knowledge of previous control input u_k. The controller signal is applied to the DC motor and the speed of DC motor shaft is measured. Fuzzy PID-type ILC computer based controller is shown as diagram in Figure 4.6.

4.3. Summary

In this chapter, the equipments for experimental applications are introduced. Electromechanical properties of DC motor are given. The features of the DAQ card are mentioned and the preparations of the experimental setup are explained.

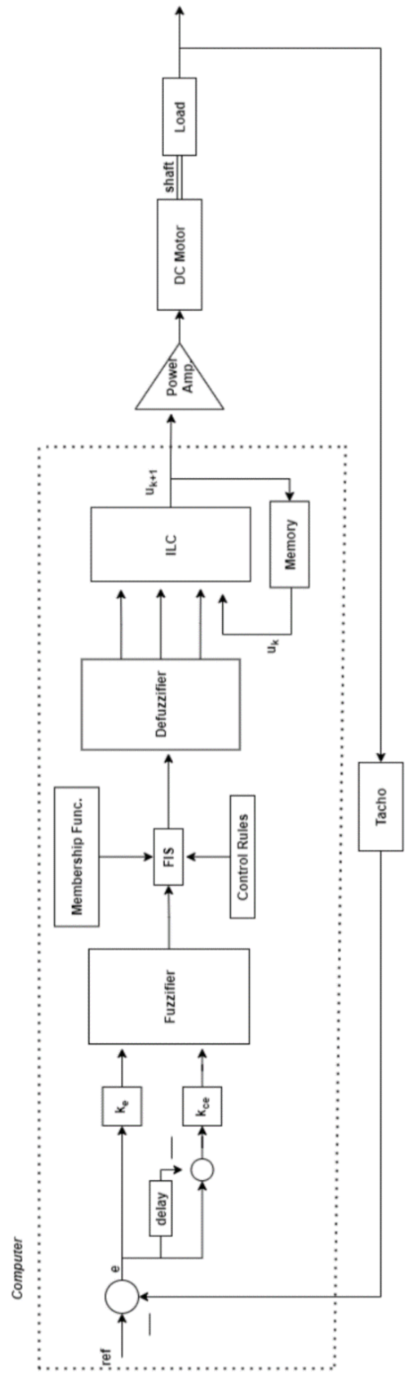


Figure 4.6. Diagram of Fuzzy PID-type ILC Implementation

5. EXPERIMENTAL APPLICATIONS AND RESULTS

This chapter contains information about real time experimental applications and the results of these applications. There are five different control methods applied to the system to be controlled in the experiment work and these are respectively:

- PID controller
- Iterative learning controller
- Fuzzy logic controller
- Fuzzy PID-type iterative learning controller
- Adaptive fuzzy PID-type iterative learning controller

In analyzing the performance of five different control methods given above, basically three main performance response values are observed and these are as follows:

- Step response test
- Tracking test
- Load test

In addition, values such as rise time, settling time and overshoot were examined in the applied control methods.

5.1. PID Controller Application and Result

PID controller has been applied to the system to be controlled first. PID controllers have three parameters as described earlier and they are proportional, integral and derivative. The parameter gains K_p, K_i and K_d were determined before the system was run.

5.1.1. PID Controller Step Test

Using closed-loop Ziegler Nichols method, PID controller parameters are calculated to be K_p=23.5493, K_i=28.5714, K_d=0.0088. The transient performance graph of PID controller is shown in Figure 5.1 and the control signal corresponding to the transient performance is shown in Figure 5.2. The error signal graph of the transient performance of PID controller is shown in Fig.5.3.

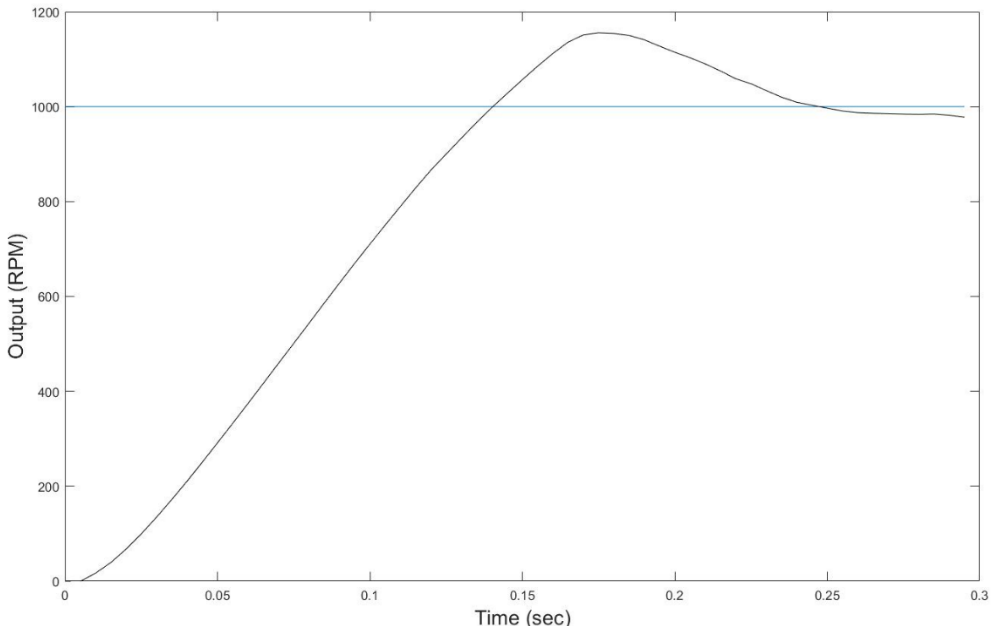


Figure 5.1. The transient performance of PID controller

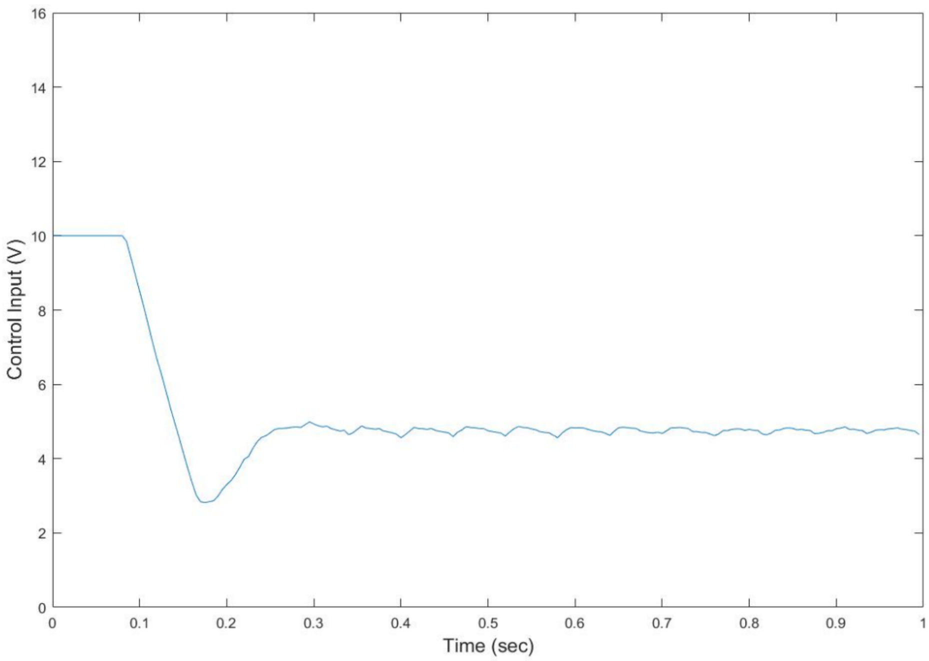


Figure 5.2. Control signal graph of PID controller

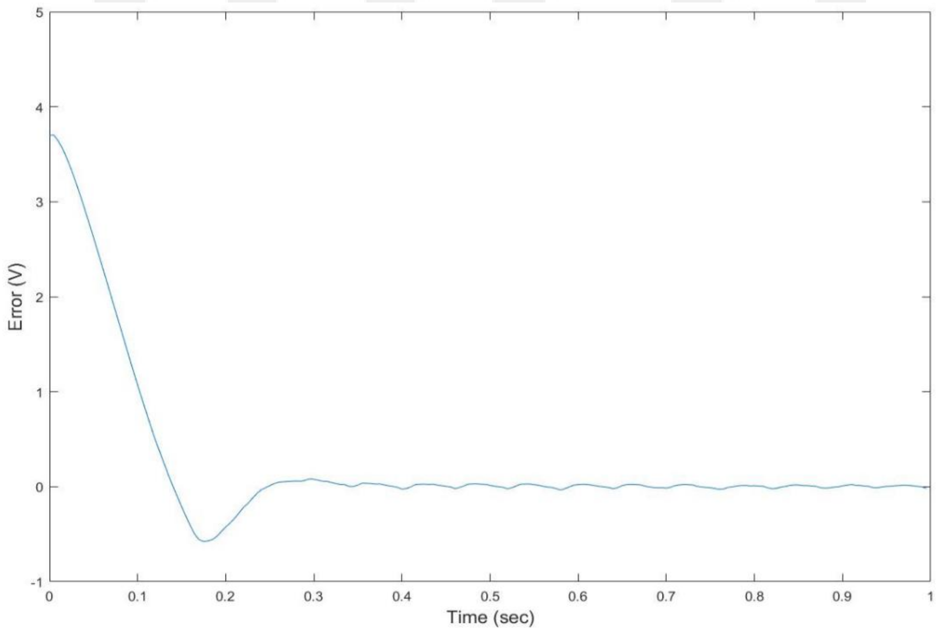


Figure 5.3. Error signal graph of transient performance

An overshoot is observed on the system output signal of the applied PID controller according to the set-point transient response information. In addition, the rise time and settling time values of the system are obtained in the output signal analysis. These values are given in Table 5.1.

Table 5.1. The valeus of rise time, settling time and overshoot

	Rise Time	Settling Time	Overshoot	Output Deviation (rpm)
PID Controller	130 ms	345 ms	15%	± 6.2

5.1.2. PID Controller Tracking Test

The tracking performance is the ability of the system to adapt to this situation in changing the set point value during operation. In this study, the system reference value is changed as 900 rpm and 1000 rpm at 1 second intervals and the tracking performance of the system is observed against these changes. The graphes of system tracking performace and its control signal are shown in Figure 5.4 and Figure 5.5 respectively.

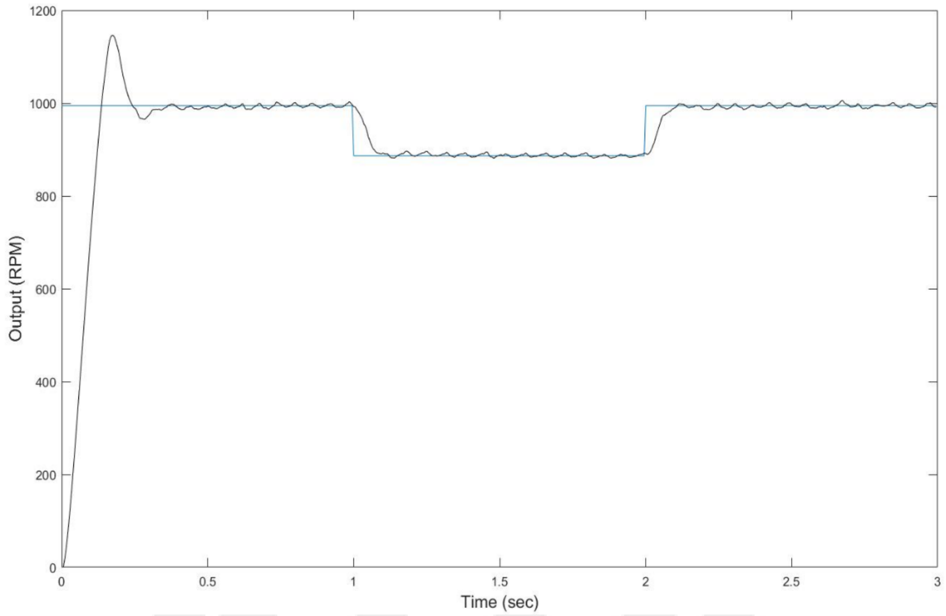


Figure 5.4. The tracking performance of PID controller

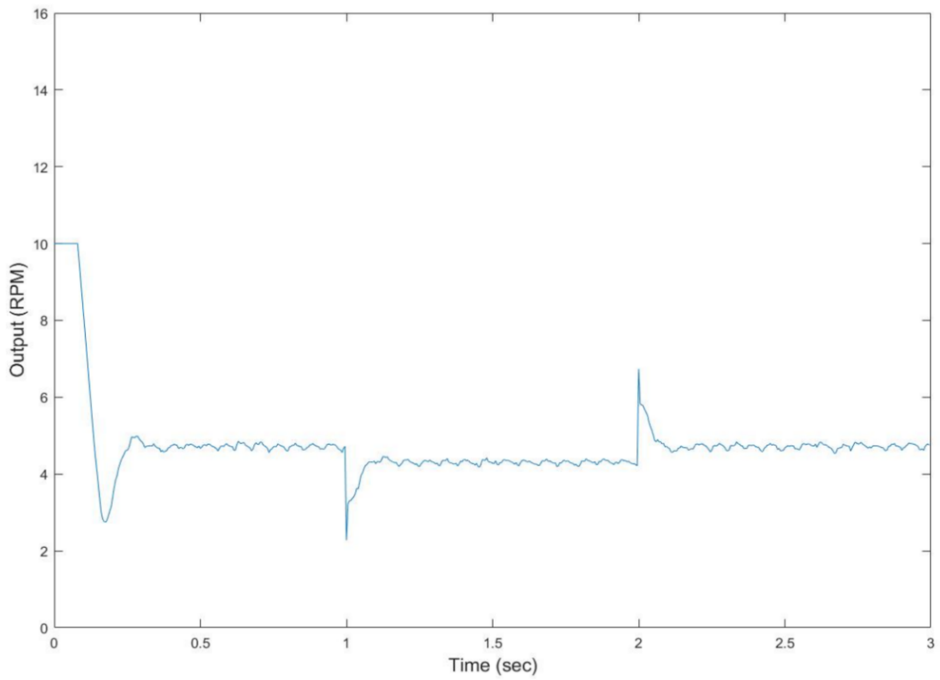


Figure 5.5 The control signal of PID controller tracking performance

5.1.3. PID Controller Load Test

Disturbance rejection performance is a response to an external disturbance while the system is running. Disturbance is applied to the system with the position sensor in the control setup. In this study, a disturbing force is applied to the system between the 3rd and 4th seconds. The disturbance rejection performance of PID controller and its control signal are demonstrated in Figure 5.6 and Figure 5.7 respectively.

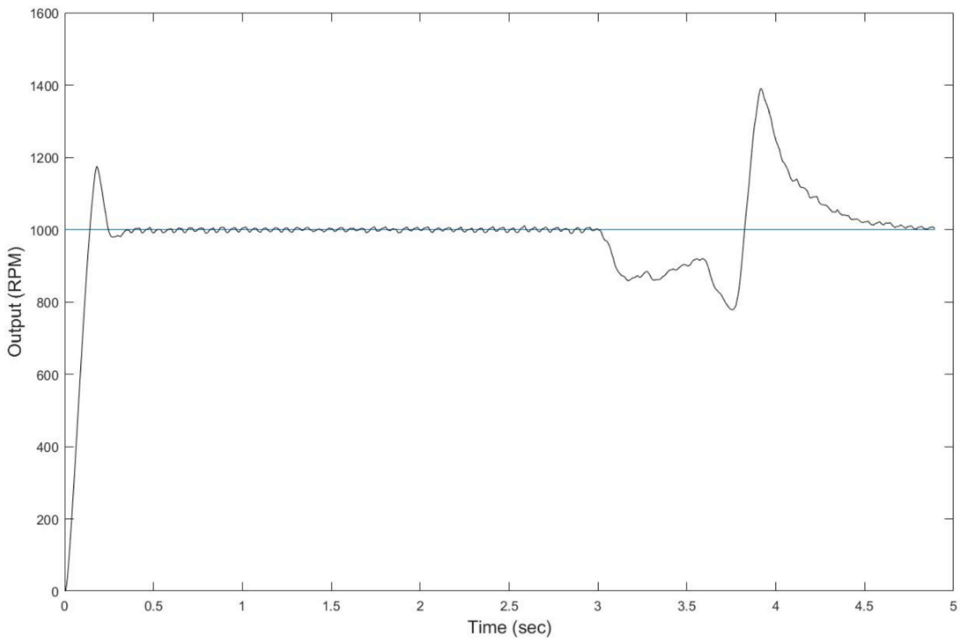


Figure 5.6. The disturbance rejection performance of PID controller

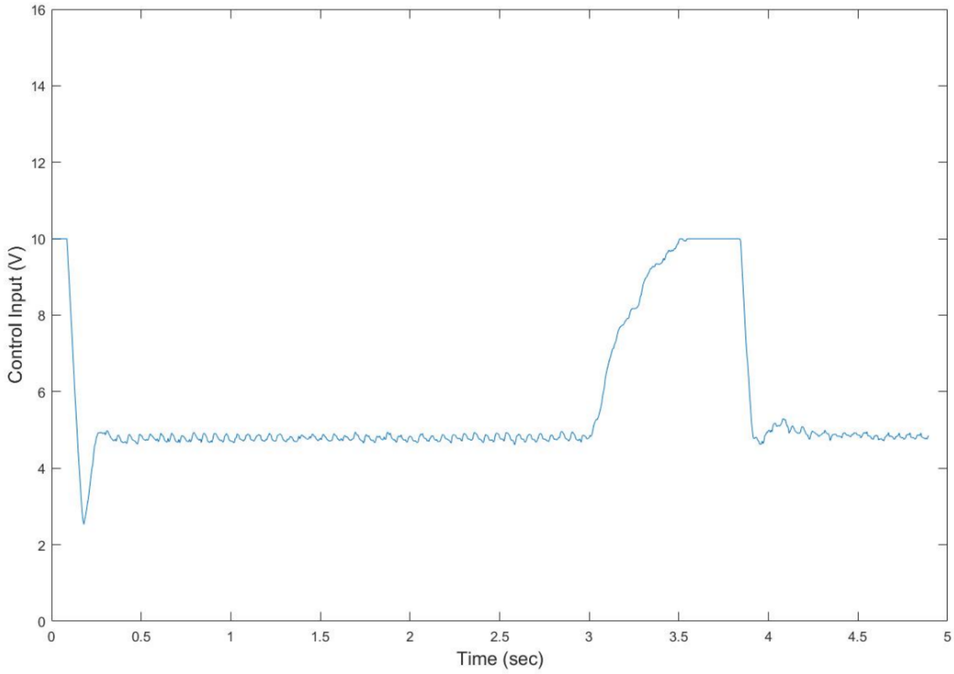


Figure 5.7. The control signal of PID controller disturbance rejection performance

5.2. Iterative Learning Controller Application and Results

Iterative learning controller (ILC) is a method that uses the control signal generated in the previous trial while generating the system control signal as explained earlier. The developed ILC was applied to the system and transient, tracking and disturbance rejection performances were examined.

5.2.1. Iterative Learning Controller Step Test

The transient performance graph of ILC controller is demostrated in Figure 5.8 and the control signal corresponding to the transient performance is shown in Figure 5.9.

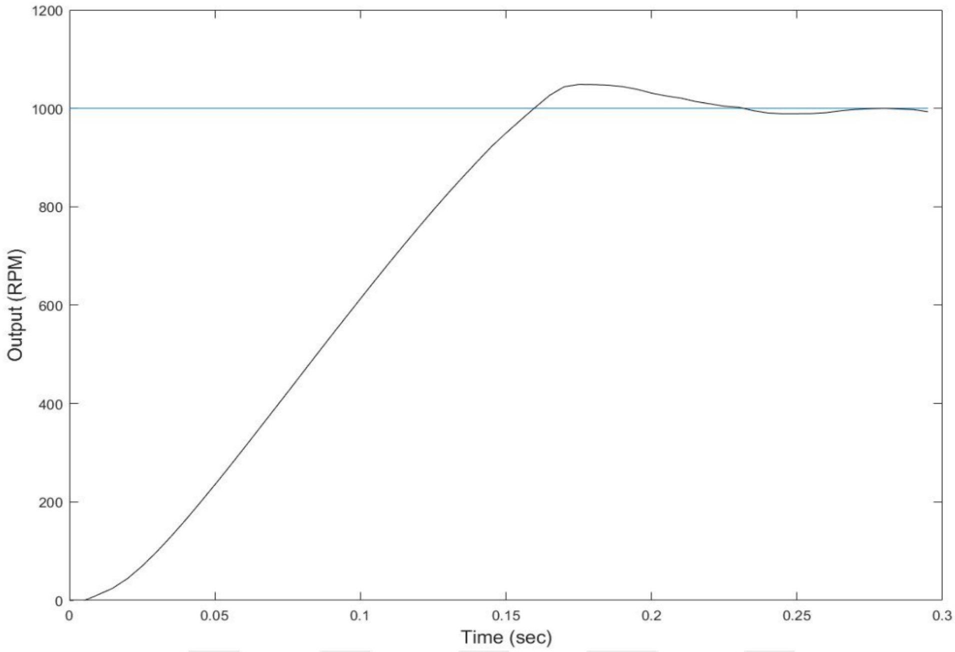


Figure 5.8. Transient performance of ILC

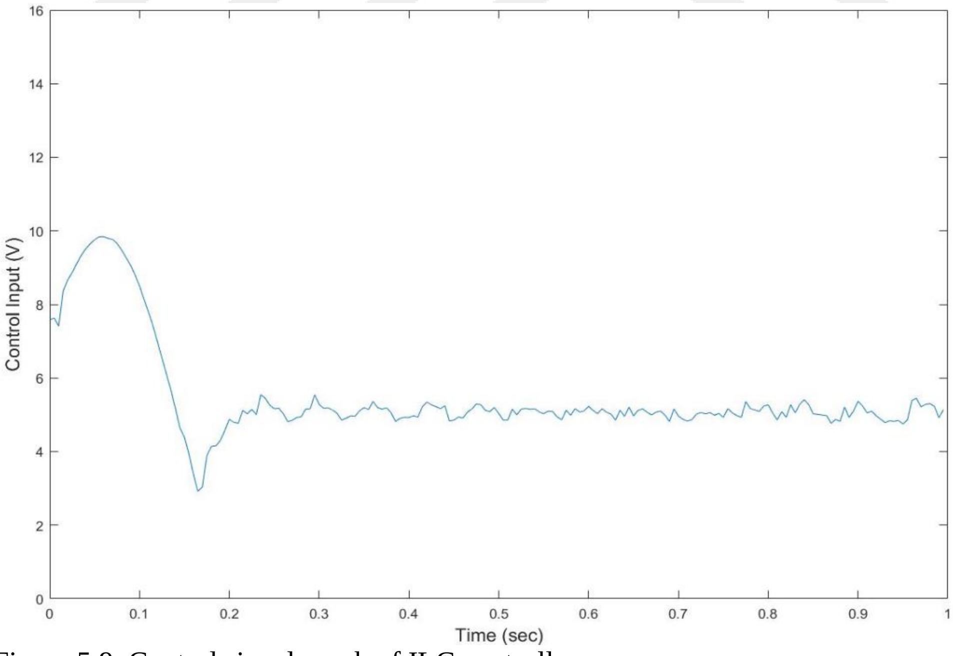


Figure 5.9. Control signal graph of ILC controller

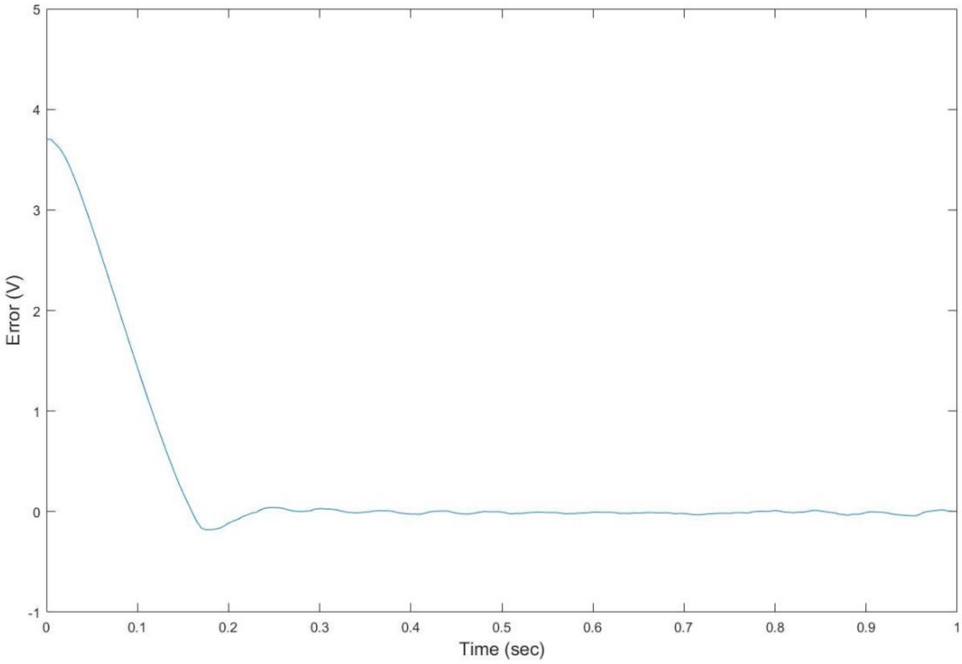


Figure 5.10. Error signal of ILC transient performance

The error signal graph of the transient performance of ILC controller is shown in Fig.5.10.

According to the transient performance graphes, a small overshoot is observed in this experiment results. The rise time, settling time and overshoot values are given in Table 5.2.

Table 5.2. The valeus of rise time, settling time and overshoot

	Rise Time	Settling Time	Overshoot	Output Deviation (rpm)
ILC Controller	140 ms	230 ms	4.8%	±5.3

5.2.2. ILC Tracking Test

The tracking performance of the ILC method applied to the system has been examined. The reference value of the system is changed to 900 rpm and 1000 rpm at

1 second intervals. The tracking performance of ILC is observed. The graphes of tracking performance and its control signal are demonstrated in Figure 5.11 and Figure 5.12 respectively.

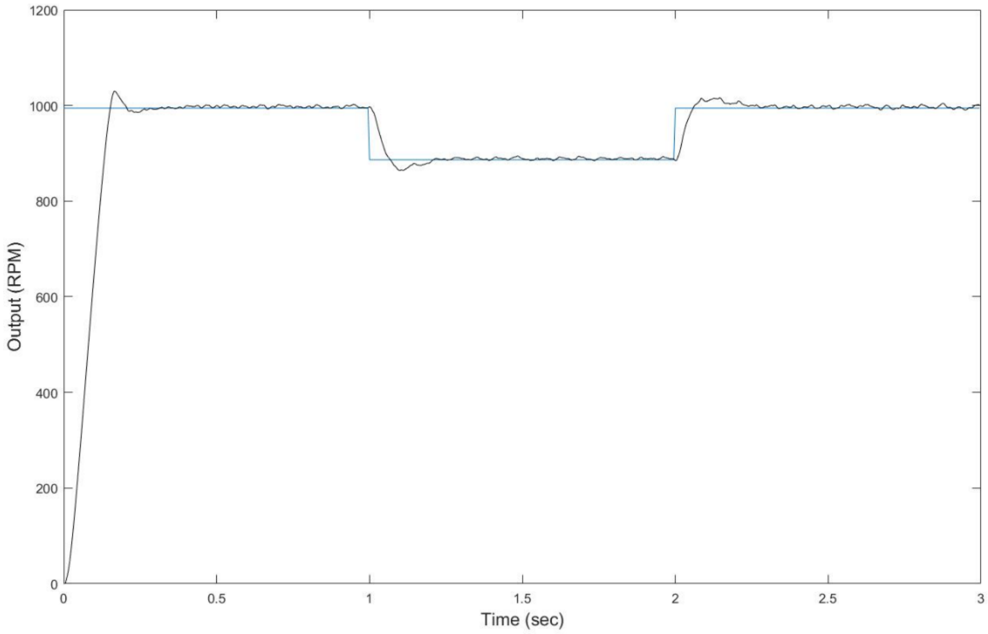


Figure 5.11. The tracking performance of ILC

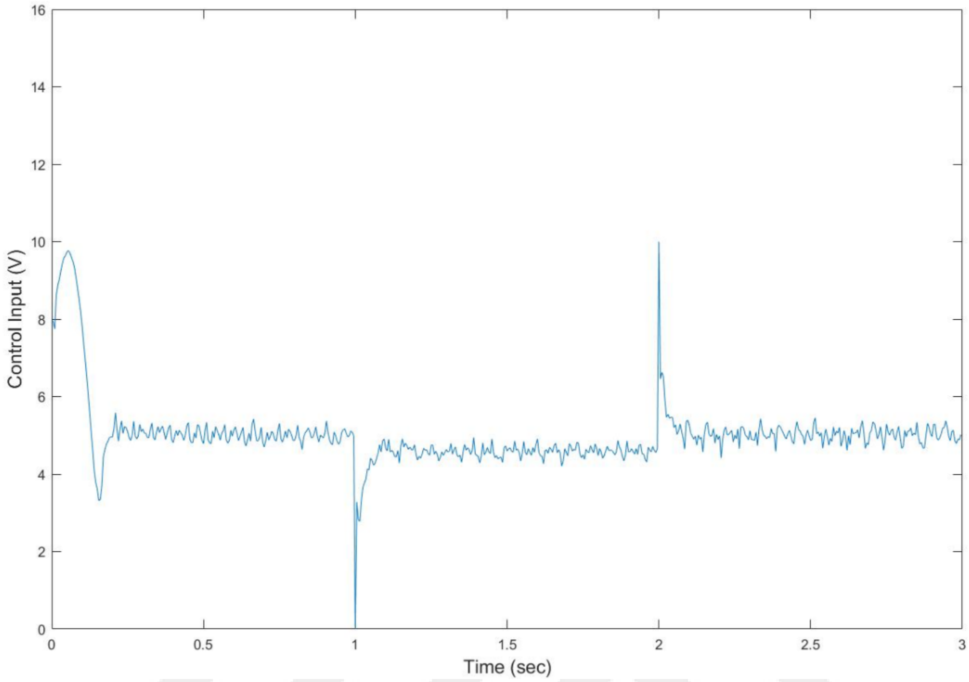


Figure 5.12. The control signal of ILC tracking performance

5.2.3. ILC Load Test

A disturbing force between the 3rd and 4th seconds is applied to the system from outside. The disturbance rejection performance of the ILC against this disrupting force was examined. The graphes of disturbance rejection performance and its control signal are demonstrated in Figure 5.13 and Figure 5.14 respectively.

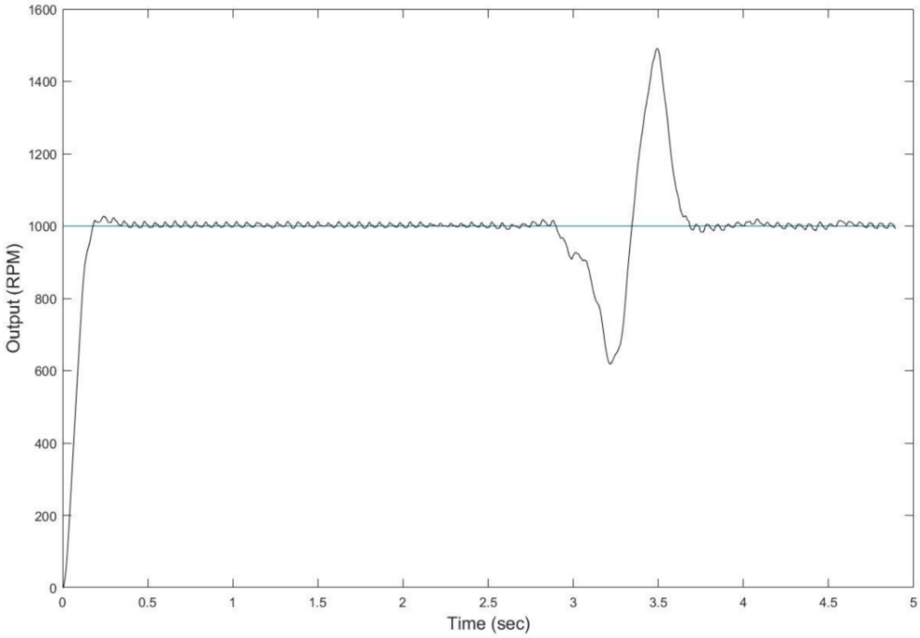


Figure 5.13. The disturbance rejection performance of ILC

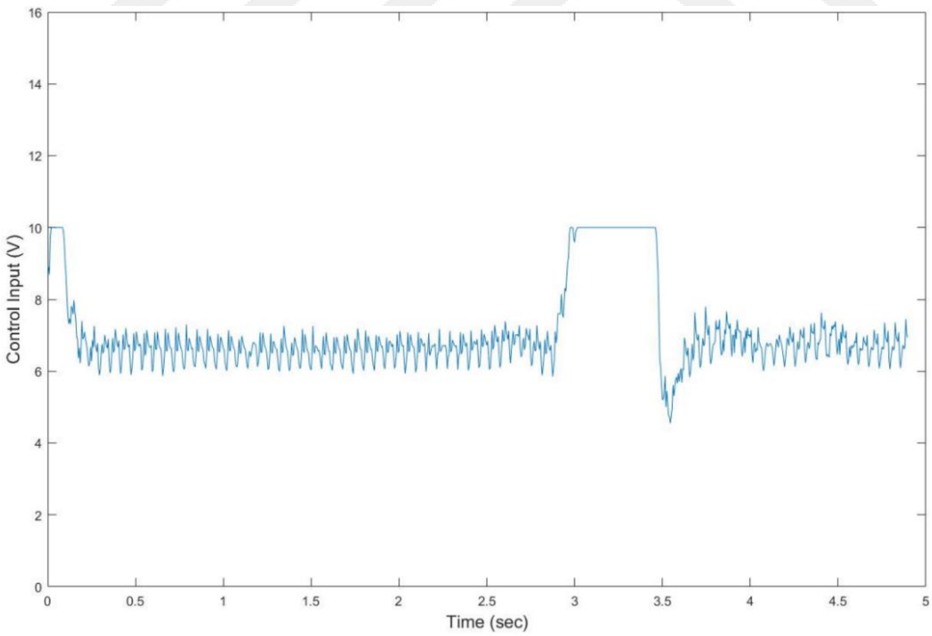


Figure 5.14. The control signal of disturbance rejection performance of ILC

5.3. Fuzzy Logic PID Controller Application and Results

The fuzzy logic controller is the method that calculates the PID coefficients in the fuzzy system and then controls the plant. As explained before, membership functions are determined for inputs and outputs and fuzzy rules are created for them. Thus, the outputs of the system are generated according to the input states over time. In this experiment, there are 7 membership functions for each input and output.

5.3.1. FLC Step Test

The transient performance of fuzzy logic controller is observed and the graph of transient performance is demonstrated in Figure 5.15 and its control input signal graph is demonstrated in Figure 5.16.

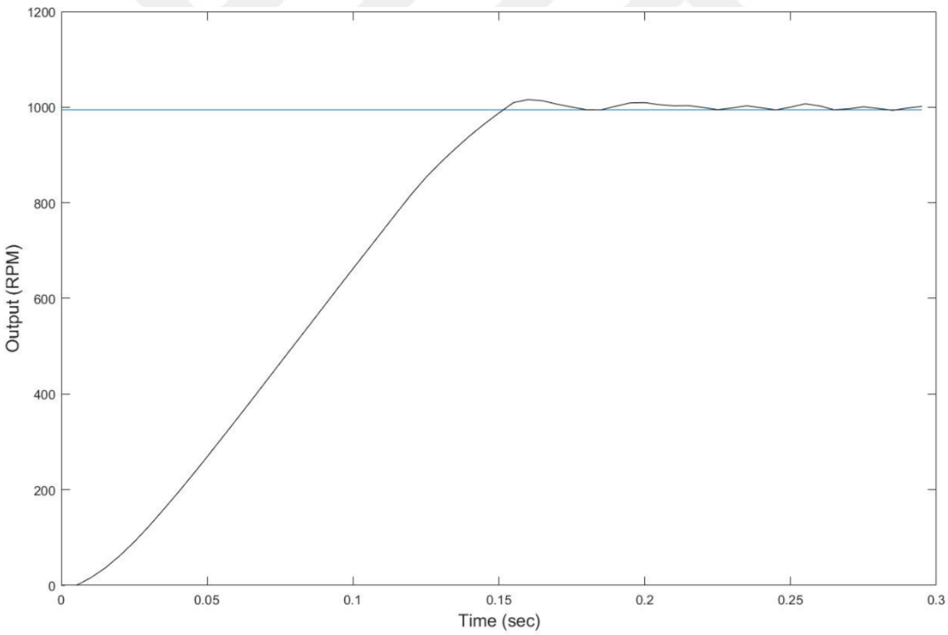


Figure 5.15. The transient performance of fuzzy logic controller

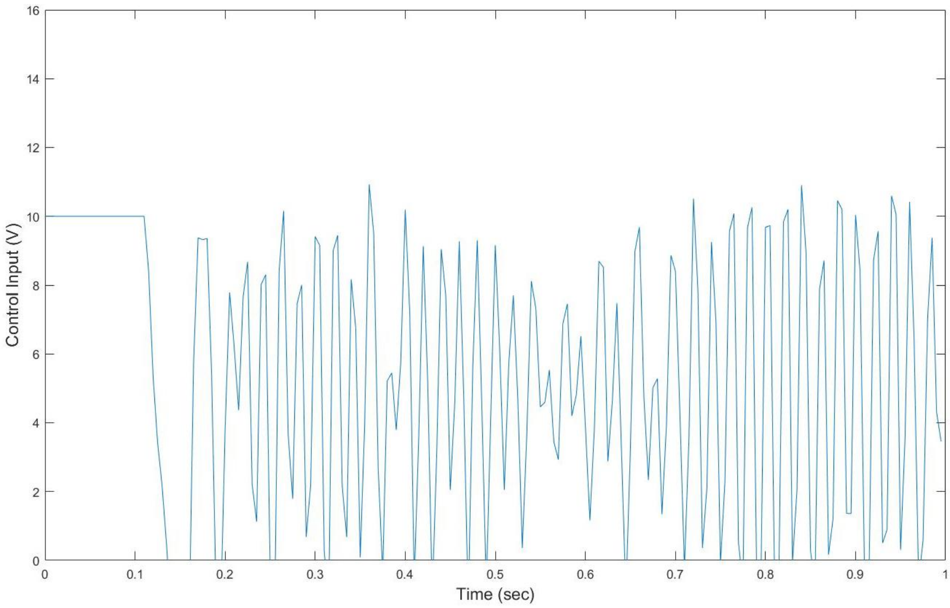


Figure 5.16 The control signal of fuzzy logic controller

The error signal graph of transient performance of fuzzy logic controller is demonstrated in Fig 5.17.

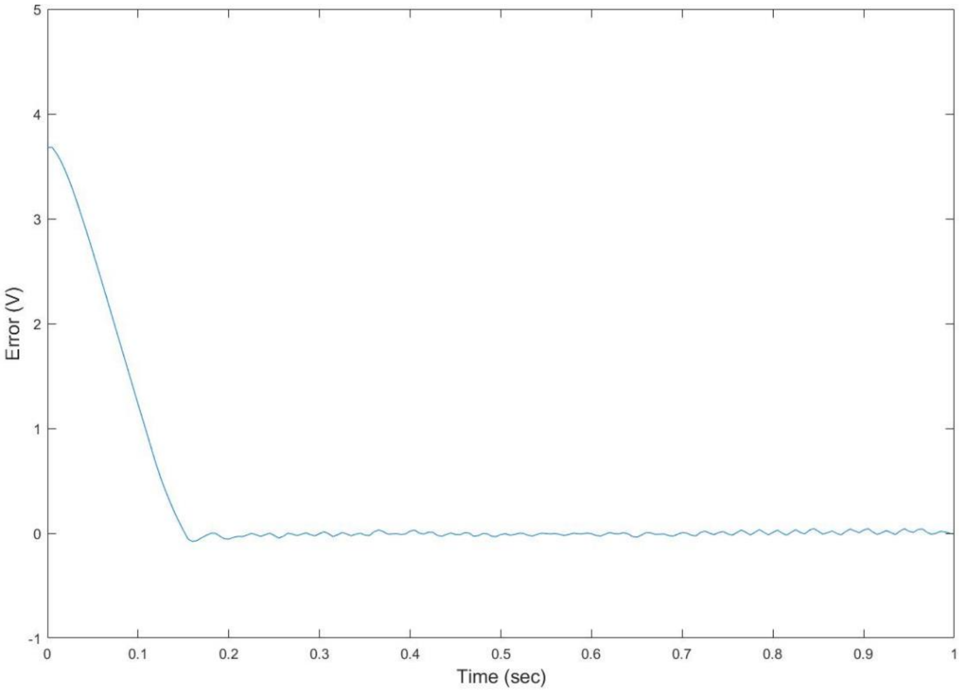


Figure 5.17. The error signal of fuzzy logic controller

According to transient performance datas of fuzzy logic controller, no overshoot occurrence has been observed. The rise time, settling time and overshoot values are given in Table 5.3.

Table 5.3. The values of rise time, settling time and overshoot

	Rise Time	Settling Time	Overshoot	Output Deviation (rpm)
Fuzzy Logic Controller	145 ms	170 ms	No overshoot	± 5.5

In this experimant, the PID coefficients are calculated by fuzzy system and the produced coefficeints of PID parameters are demonstrated in Figure 5.18.

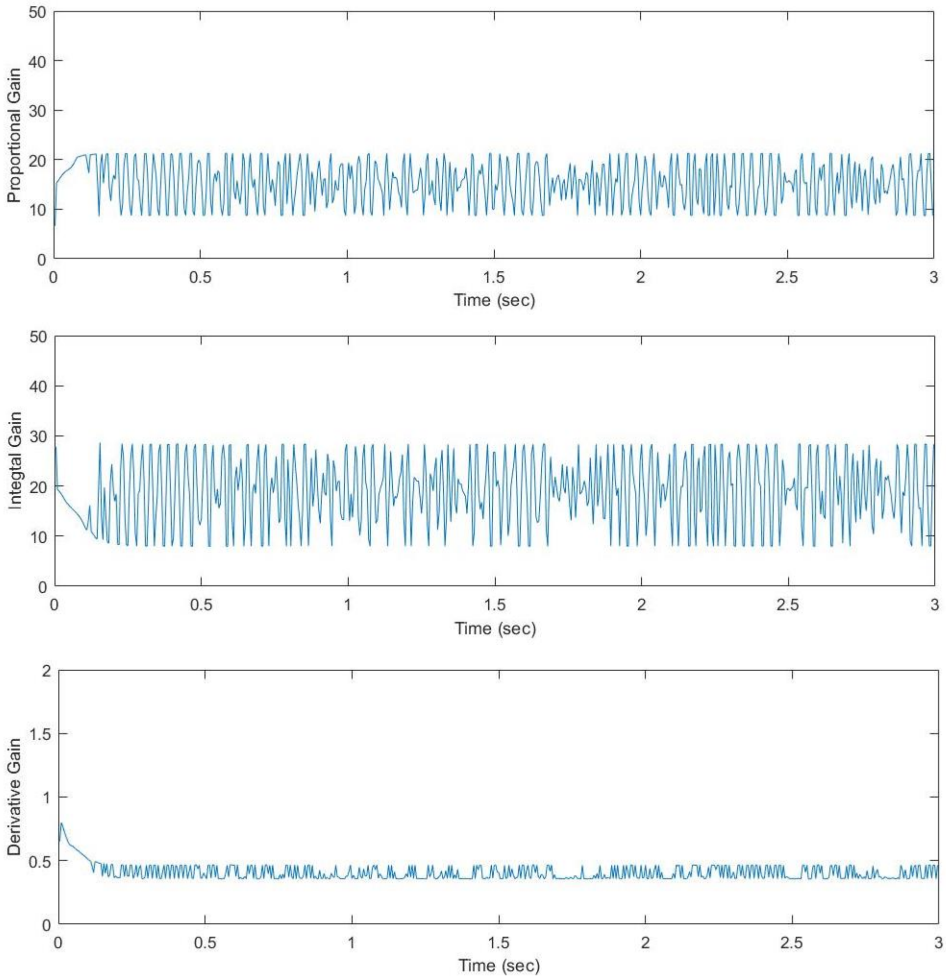


Figure 5.18. PID coefficents of fuzzy logic controller

5.3.2. FLC Tracking Test

The tracking performance of the fuzzy logic controller has been examined. The reference value has been changed as 900rpm and 1000rpm every second and it is observed how the fuzzy logic controller tracks these changes and generates a response. The trabcking performance and its control signal of fuzzy logic controller are demonstrated in Figure 5.19 and Figure 5.20 respectively.

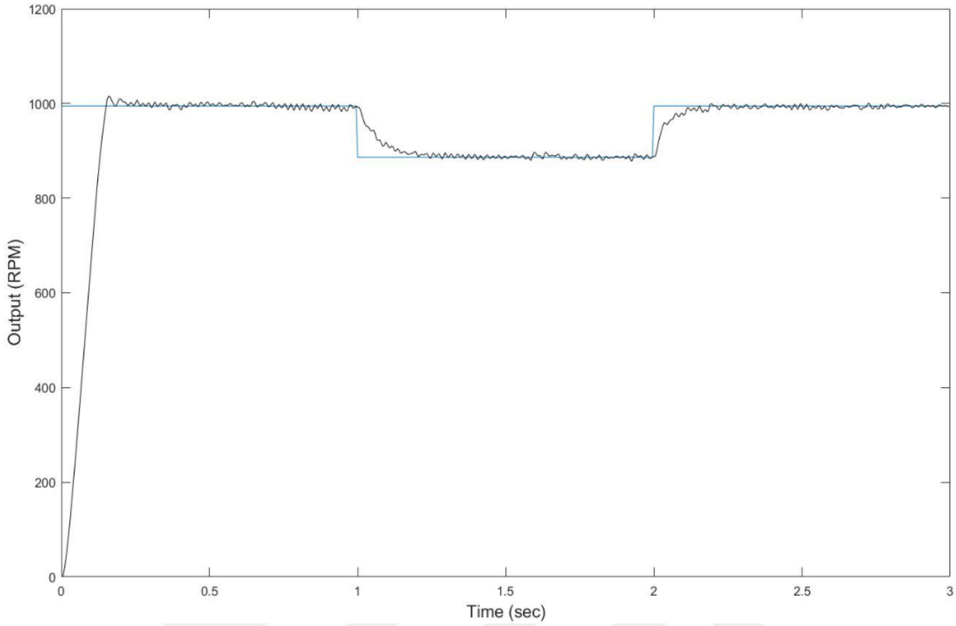


Figure 5.19. The tracking performance of fuzzy logic controller

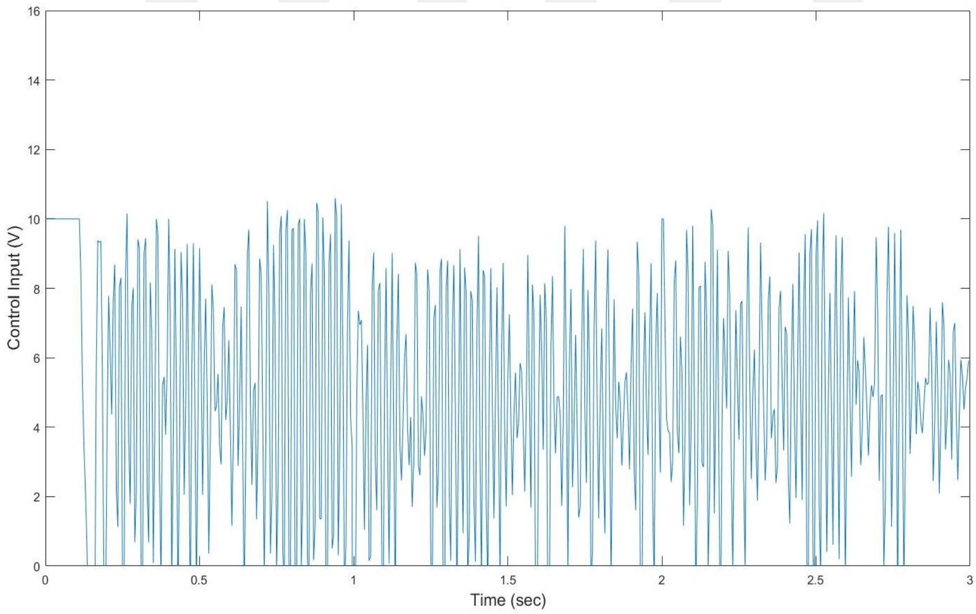


Figure 5.20. The control signal of tracking performance

5.3.3. FLC Load Test

The disturbance rejection performance of the fuzzy logic controller was examined and an external disturbing force was applied to the system between the 3rd and 4th seconds. The disturbance rejection performance and its control signal are demonstrated in Figure 5.21 and Figure 5.22 respectively.

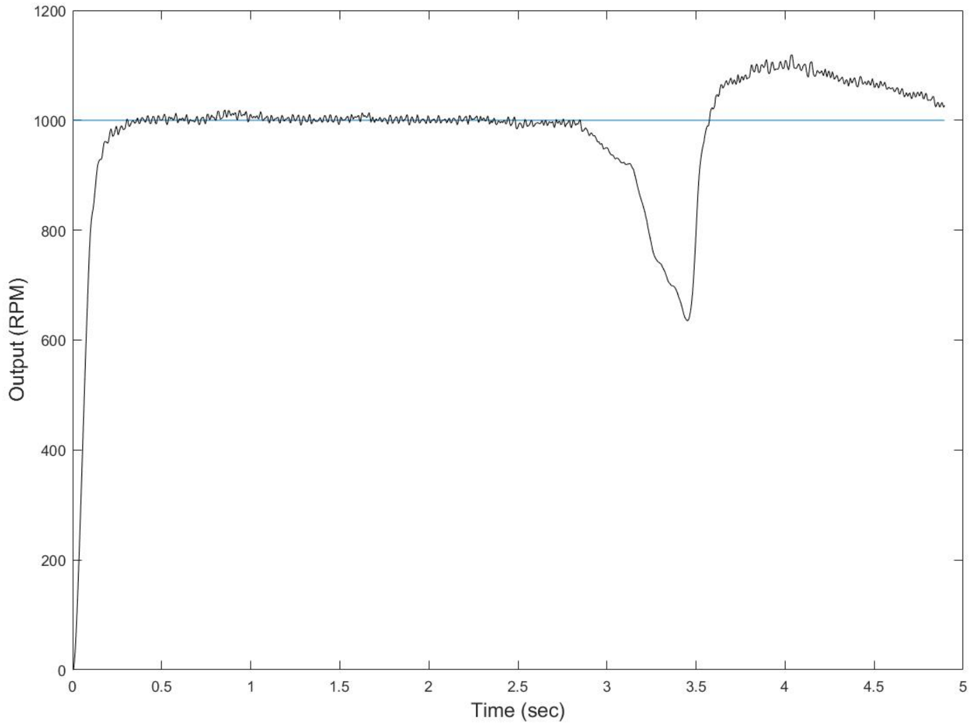


Figure 5.21. The disturbance rejection performance of fuzzy logic controller

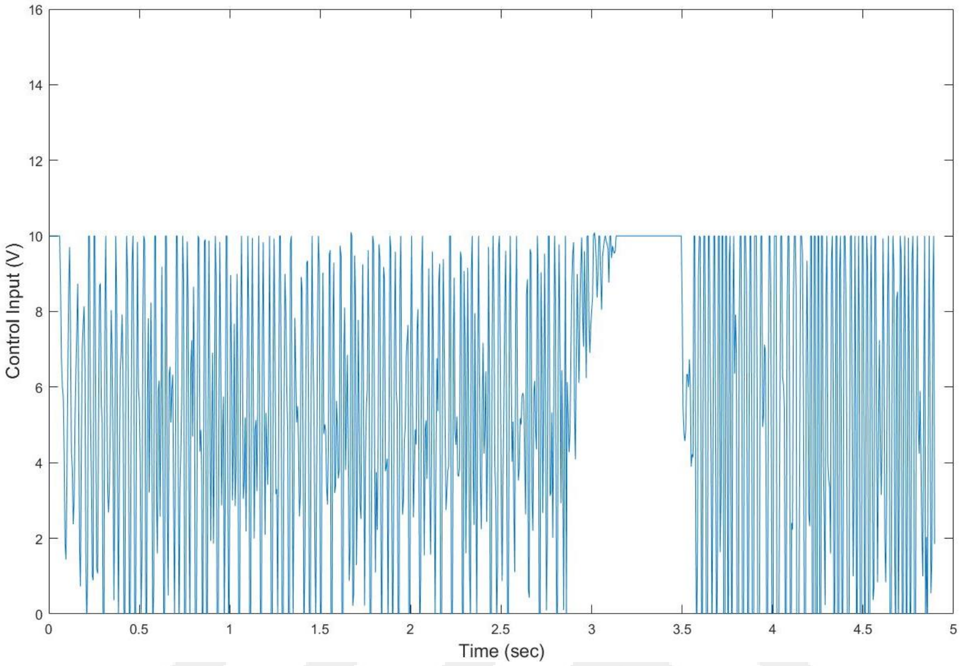


Figure 5.22. The control signal of disturbance rejection performance

5.4. Fuzzy PID-type Iterative Learning Controller Application and Results

The fuzzy PID-type ILC is created by combining fuzzy logic control and iterative learning control methods as explained in previous chapers. In this developed method, the PID parameters entering the iterative learning control block are produced from the fuzzy logic control block. By including the previous control signal in the ILC block, the control input is generated. The algorithm which is used in fuzzy logic controller is used in this method as well, Because of that, there are 7 membership for each input and output.

5.4.1. Fuzzy PID-type Iterative Learning Controller Step Test

The set-point transient performance of fuzzy PID-type ILC has been examined and the graphes of transient performance and its control signal are demonstrated in Figure 5.23 and Figure 5.24 respectively.

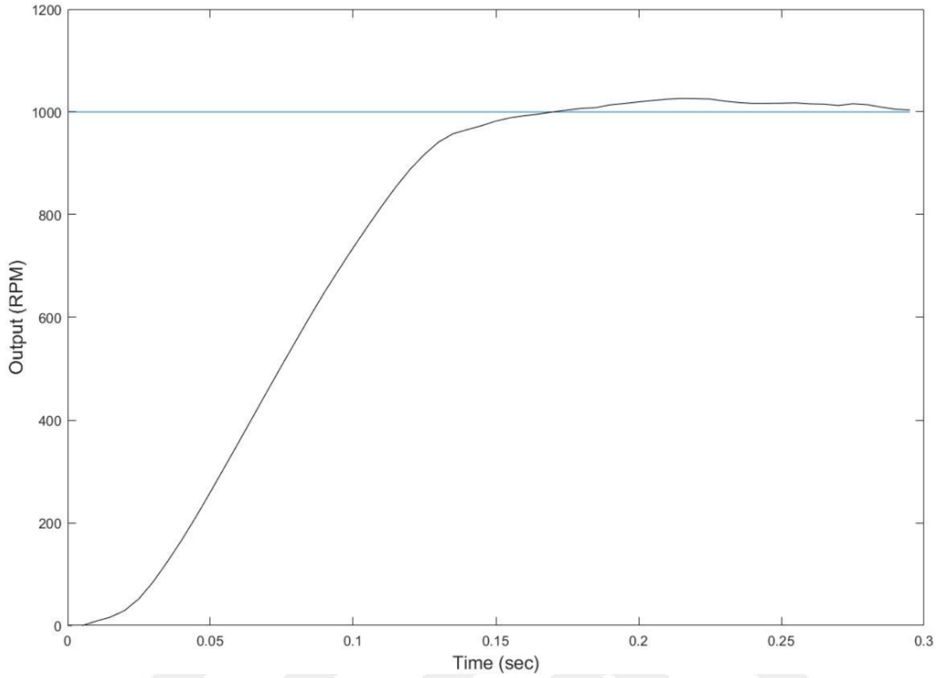


Figure 5.23. The transient performance of fuzzy PID-type ILC

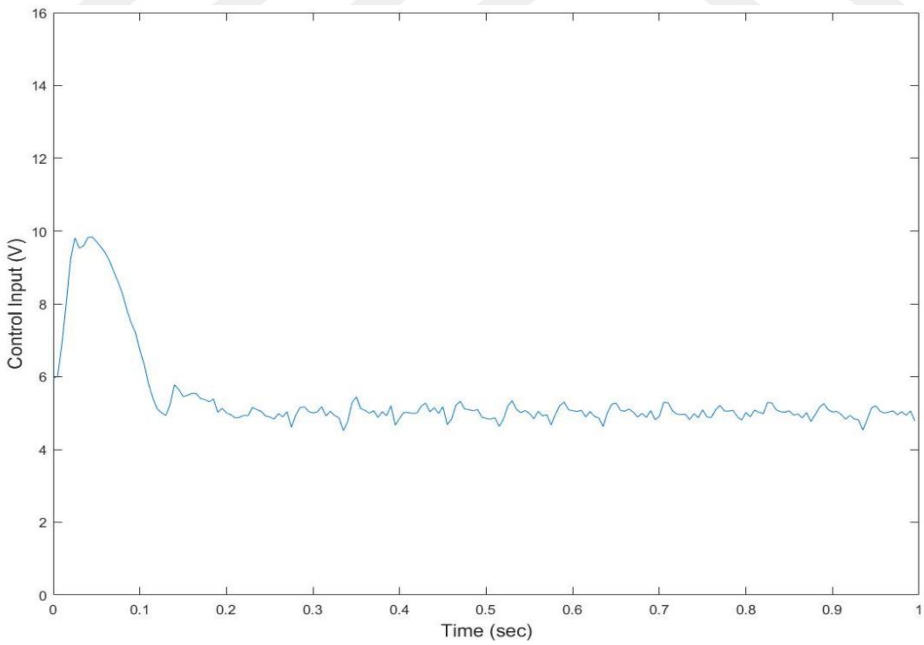


Figure 5.24. The control signal of fuzzy PID-type ILC

The error signal graphes of transient performance of fuzzy PID-type ILC is shown in Figure 5.25.

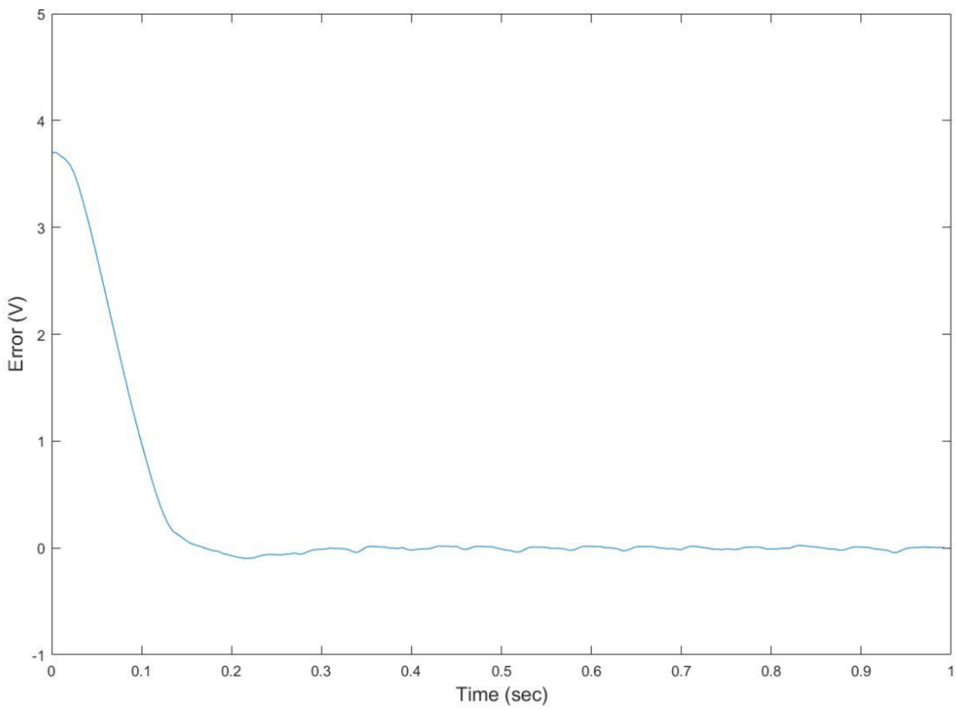


Figure 5.25. The error signal of fuzzy PID-type ILC

According to the transient performance, the rise time, settling time and overshoot values are given in Table 5.4.

Table 5.4. The values of rise time, settling time and overshoot

	Rise Time	Settling Time	Overshoot	Output Deviation (rpm)
Fuzzy PID-type ILC	120 ms	190 ms	2%	± 4.8

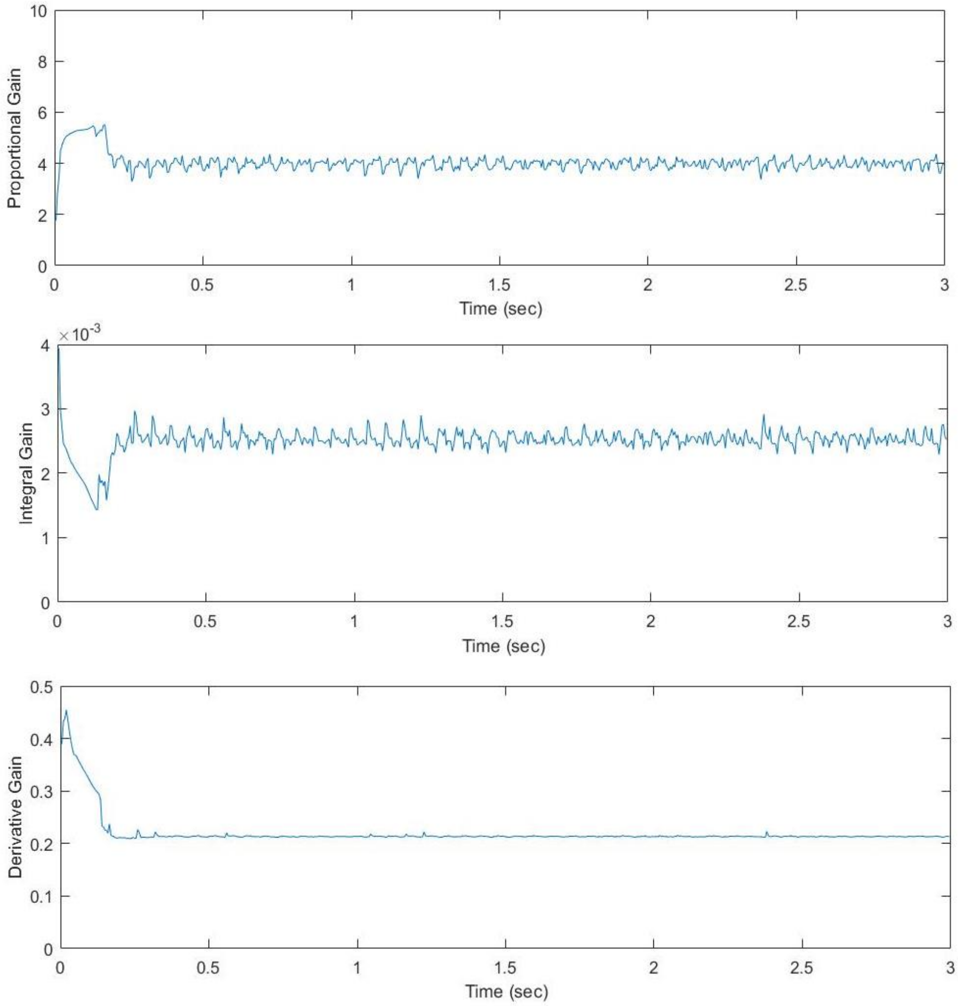


Figure 5.26. PID coefficents of fuzzy PID-type ILC

The PID coefficients which is produced in fuzzy block of fuzzy PID-type ILC are demonstrated in Figure 5.26.

5.4.2. Fuzzy PID-type Iterative Learning Controller Tracking Test

The tracking performance of the fuzzy PID-type ILC has been observed. The response of the system to the reference signal changed at 900 rpm and 1000 rpm at

1-second intervals was examined. The graphes of tracking performance and its control signal are demonstrated in Figure 5.27 and Figure 5.28 respectively.

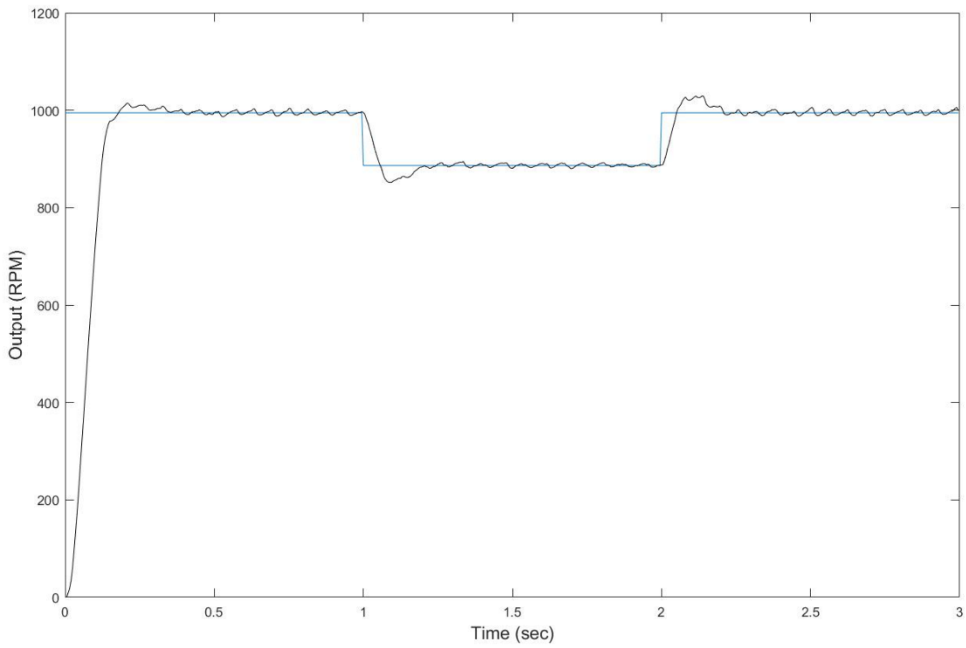


Figure 5.27. The tracking performance of fuzzy PID-type ILC

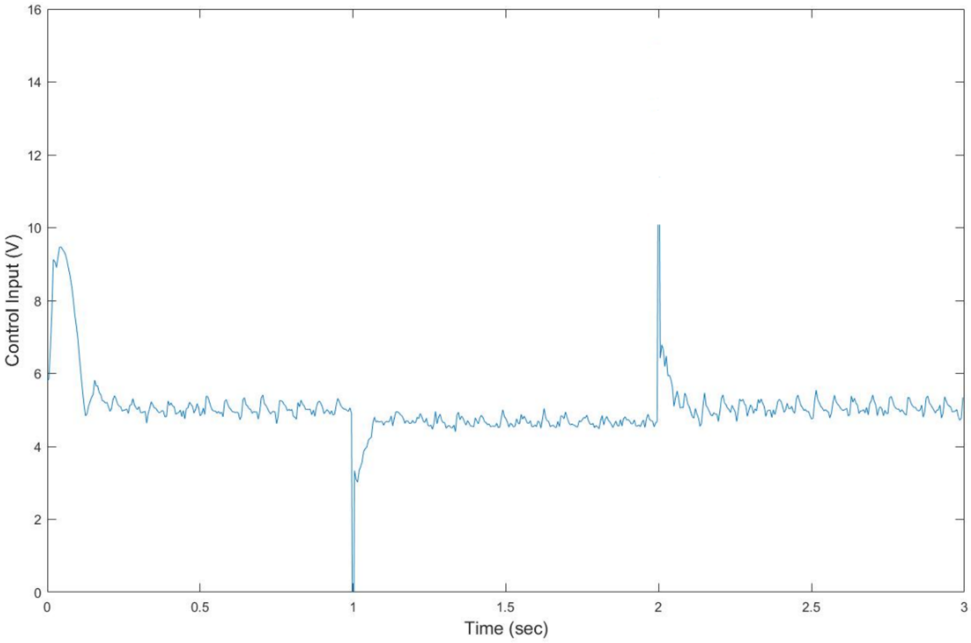


Figure 5.28. The control signal of tracking performance

5.4.3. Fuzzy PID-type Iterative Learning Controller Load Test

The disturbance rejection performance of the fuzzy PID-type ILC against an external disturbing force between the 4th and 5th seconds of the system was examined. The graphes of disturbance rejection performance and its control signal are demonstrated in Figure 5.29 and Figure 5.30 respectively.

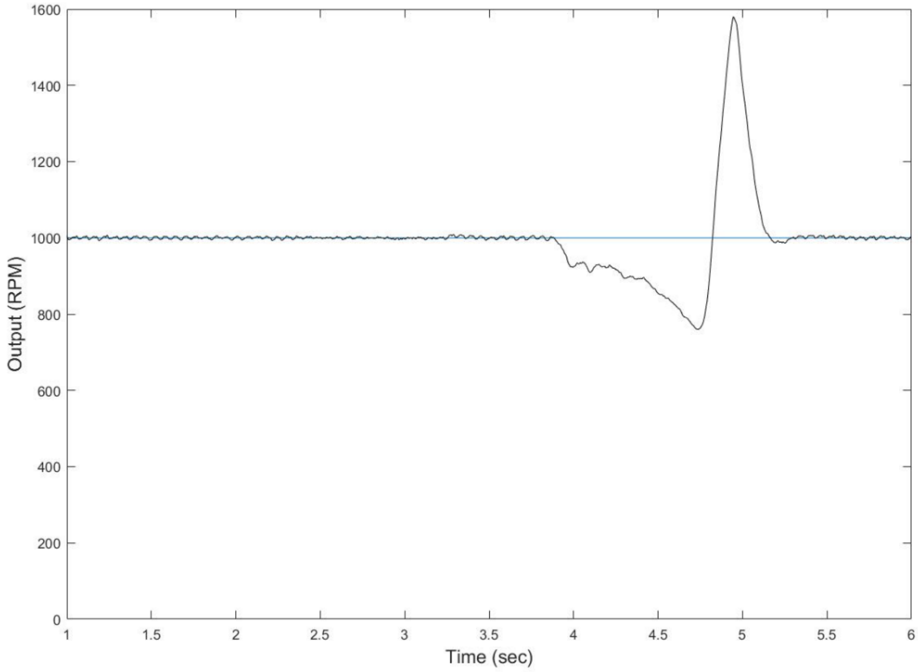


Figure 5.29. The disturbance rejection performance of fuzzy PID-type ILC

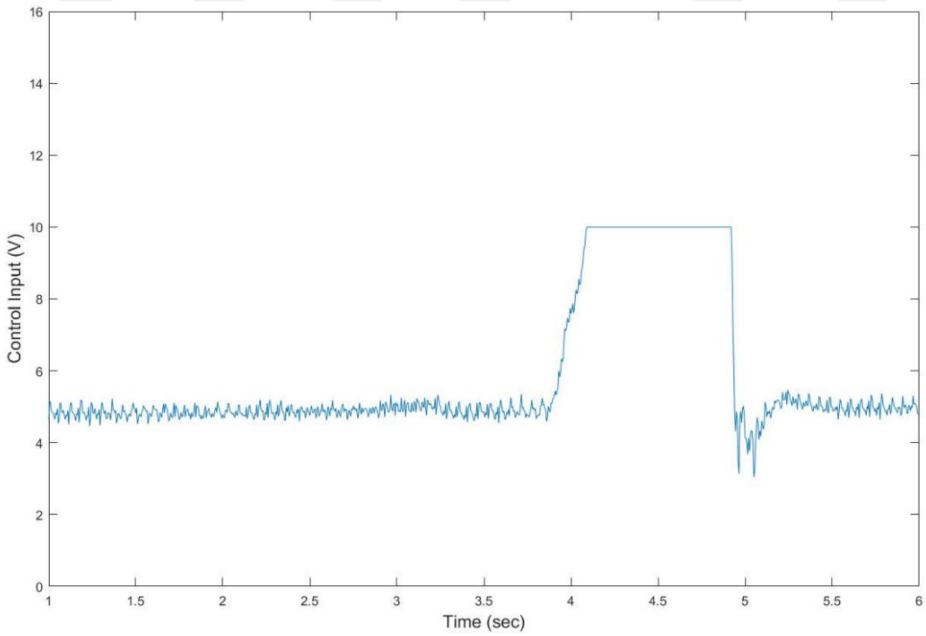


Figure 5.30. The control signal of disturbance rejection performance

5.5. Adaptive Fuzzy PID-type Iterative Learning Control Application and Result

Adaptive fuzzy PID-type ILC is created by combining adaptive algorithms with fuzzy PID-type ILC system using as scaling factor. The produced PID parameters are entered into the ILC block and the control signal is generated by using the previous control signal. The fuzzy logic control part has the same fuzzy rules as previous method. The outputs of fuzzy block is generated according to the instantaneous state of the fuzzy inputs and then they enter the ILC block.

5.5.1. Adaptive Fuzzy PID-type Iterative Learning Control Step Test

The set-point transient performance of adaptive fuzzy PID-tyoe ILC is observed and the graphes of transient performance, its control signal and error signal are demonstrated in Figure 5.31 and Figure 5.32 and Figure 5.33.

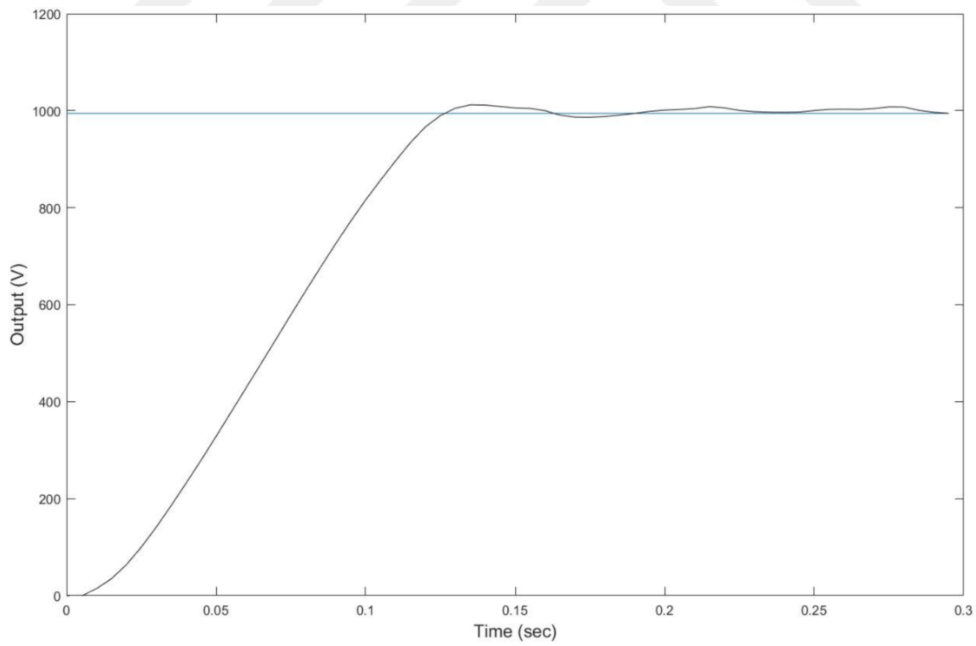


Figure 5.31. The transient performance of adaptive fuzzy PID-type ILC

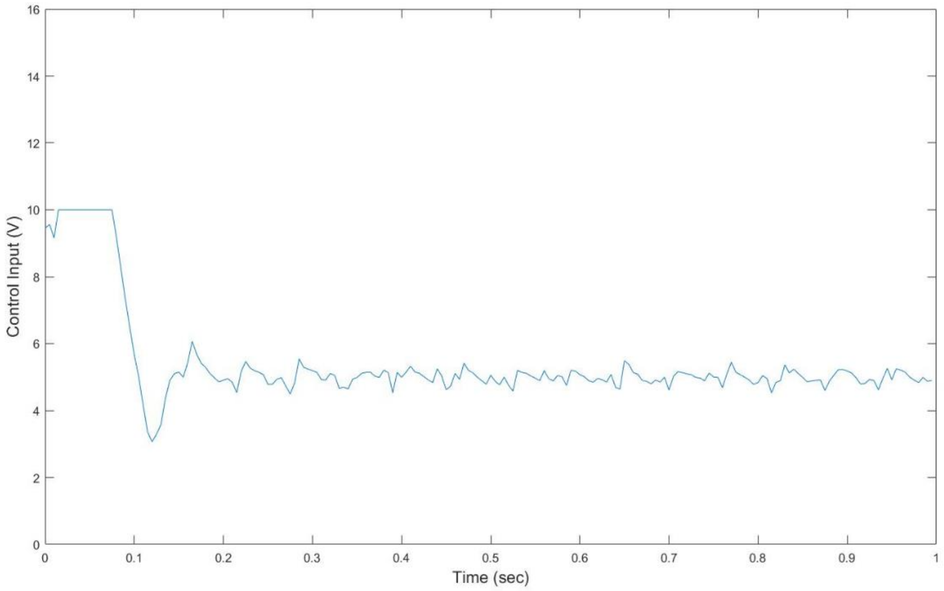


Figure 5.32. The control input of transient performance

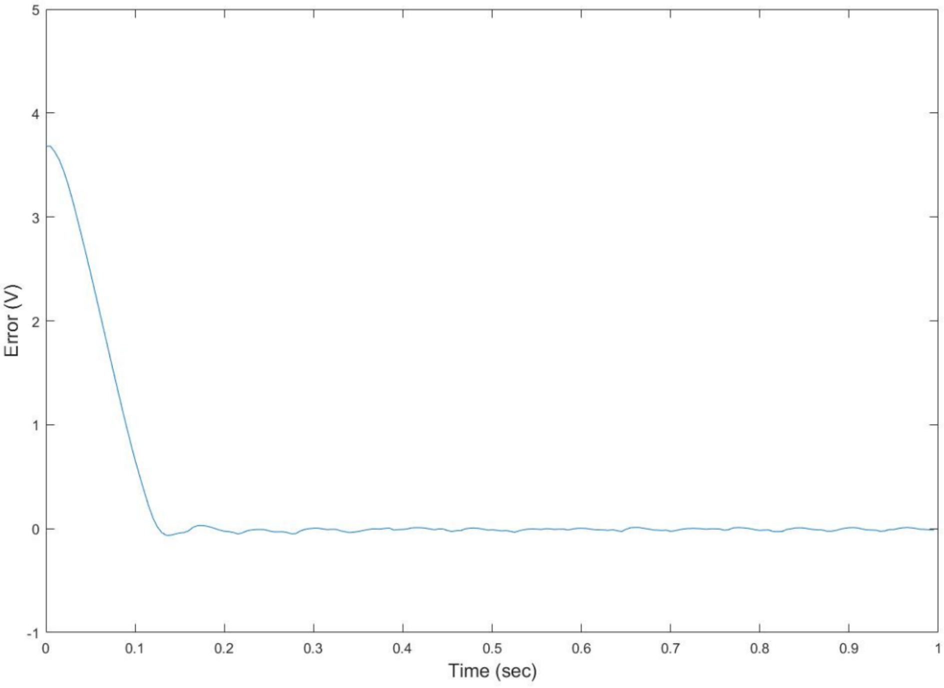


Figure 5.33. The error signal of transient performance

According to the transient performance of adaptive fuzzy PID-type ILC, the rise time,settling time and overhoot values are given in Table 5.5.

Table 5.5. The values of rise time, settling time and overshoot

	Rise Time	Settling Time	Overshoot	Output Deviation (rpm)
Adaptive Fuzzy PID-type ILC	105 ms	125 ms	No overshoot	± 4.6

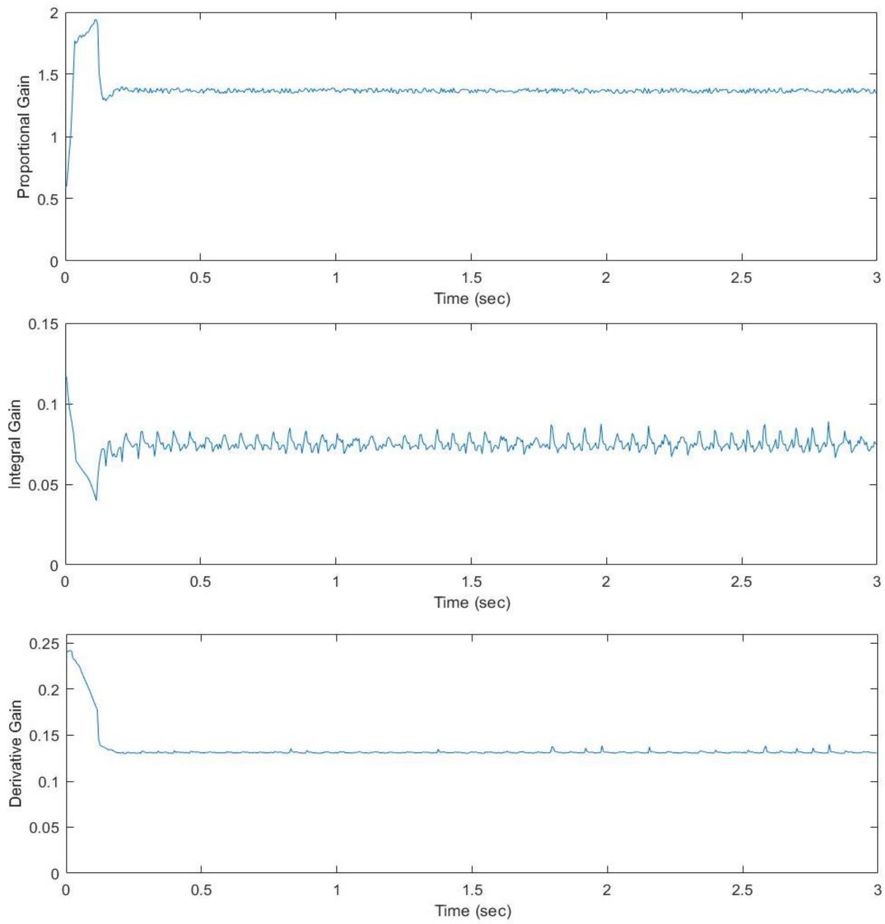


Figure 5.34. PID coefficients of adaptive fuzzy PID-type ILC

The PID coefficients which is produced in fuzzy block of adaptive fuzzy PID-type ILC are demonstrated in Figure 5.34.

5.5.2. Adaptive Fuzzy PID-type Iterative Learning Control Tracking Test

The tracking performance of adaptive fuzzy PID-type ILC has been examined. The response of the system was observed against the reference signal changed to 900 rpm and 1000 rpm. The graphes of tracking performance and its control signal of adaptive fuzzy PID-type ILC are demonstrated in Figure 5.35 and Figure 5.36 respectively.

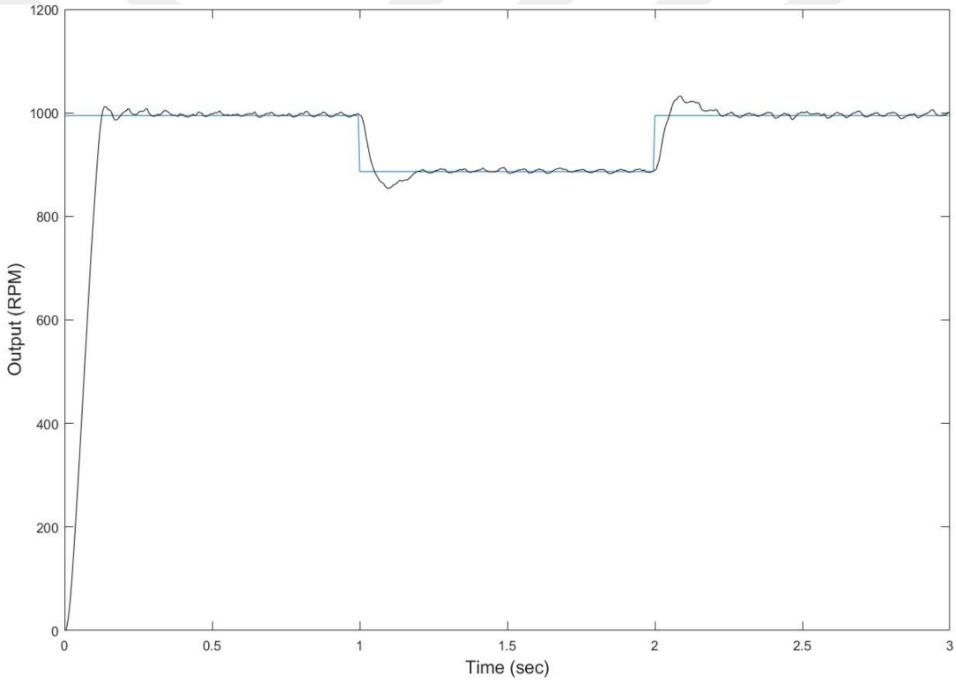


Figure 5.35. The tracking performance of adaptive fuzzy PID-type ILC

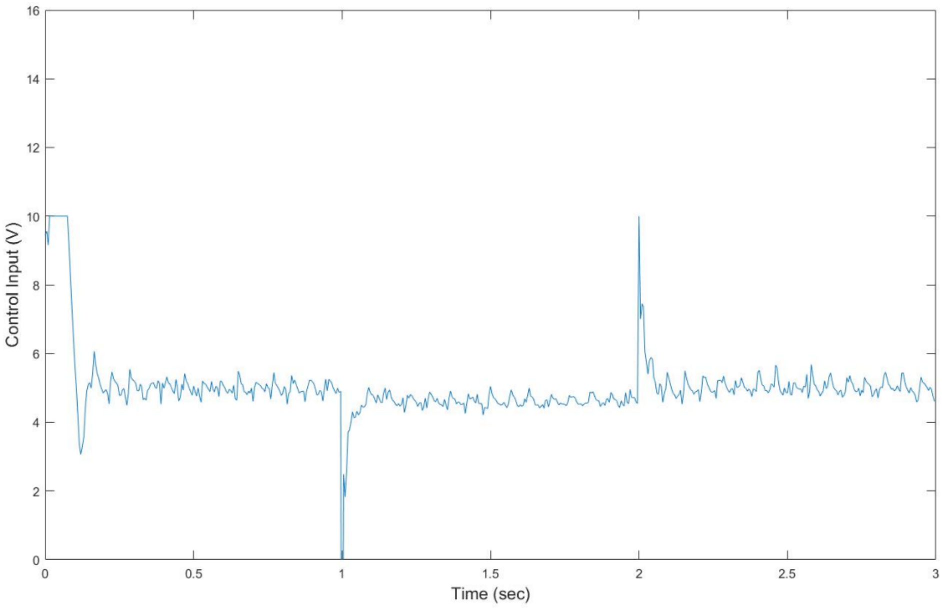


Figure 5.36. The control signal of tracking performance

5.5.3. Adaptive Fuzzy PID-type Iterative Learning Control Load Test

The disturbance rejection performance of the adaptive fuzzy PID-type ILC against an external disturbing force between the 4th and 5th seconds of the system was examined. The graphes of disturbance rejection performance and its control signal of the adaptive fuzzy PID-type ILC are demonstrated in Figure 5.37 and Figure 5.38 respectively.

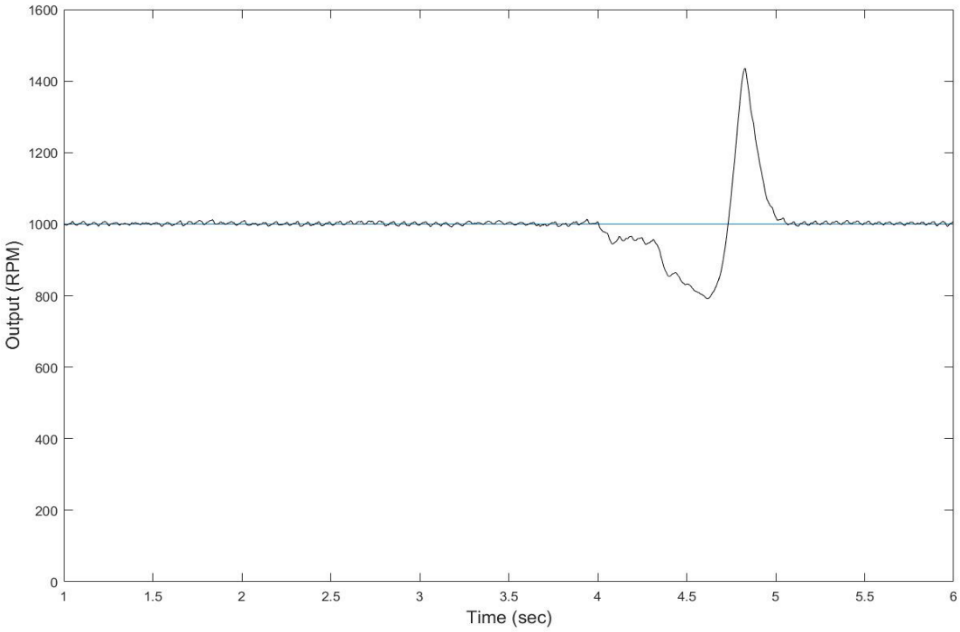


Figure 5.37. The disturbance rejection performance of adaptive fuzzy PID-type ILC

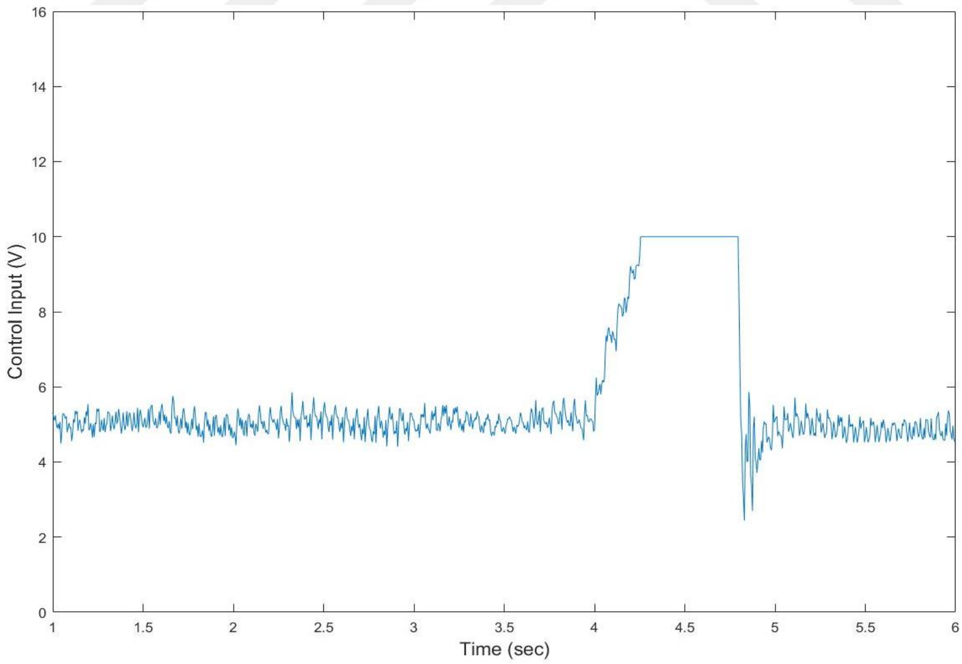


Figure 5.38. The control signal of disturbance rejection performance

5.6. Performance Indices of Control Systems

In order to evaluate the performance of control methods, some criteria are checked. The criteria used are as follows:

- Integral of absolute error (IAE)

$$IAE = \int |e(t)|\, dt \tag{5.1}$$

- Integral of square error (ISE)

$$ISE = \int e^2(t)\, dt \tag{5.2}$$

- Integral of time absolute error (ITAE)

$$ITAE = \int t|e(t)|\, dt \tag{5.3}$$

- Integral time squared error (ITSE)

$$ITSE = \int te^2(t)\, dt \tag{5.3}$$

- Integral squared control input (ISCI)

$$ISCI = \int u^2(t)\, dt \tag{5.4}$$

The performance indicies of control methods which are PID controller, iterative learning controller, fuzzy logic controller, fuzzy PID-type ILC, adaptive fuzzy PID-type ILC are given in Table 5.6.

Table 5.6. Performance indices of control methods

	IAE	ISE	ITAE	ITSE	ISCI
PID	0.3350	0.7520	0.0426	0.0290	58.5724
ILC	0.3306	0.7739	0.0490	0.0291	58.5659
FLC	0.3087	0.6736	0.0489	0.0229	69.3159
FILC	0.3045	0.7729	0.0319	0.0269	55.5119
AFILC	0.2716	0.6727	0.0340	0.0214	55.0008

The values of rise time, settling time, overshoot and output deviation of control methods which are PID controller, iterative learning controller, fuzzy logic controller, fuzzy PID-type ILC, adaptive fuzzy PID-type ILC are given in Table 5.7.

Table 5.7. The values of rise time, settling time and overshoot

	Rise Time	Settling Time	Overshoot	Output Deviation (rpm)
PID	130ms	345ms	15%	± 6.2
ILC	140ms	230ms	4.8%	± 5.3
FLC	145ms	170ms	No overshoot	± 5.5
FILC	120ms	190ms	2%	± 4.8
AFILC	105ms	125ms	No overshoot	± 4.6

5.7. Comparision of Control Methods

To summarize the results of experimental applications, the transient responces are illustrated in Figure 5.39 for all methods. It is demonstrated that the proposal adaptive fuzzy PID-type ILC method given better transient and error performance than the other methods in term of time domain specification in Table 5.6 and Table 5.7.

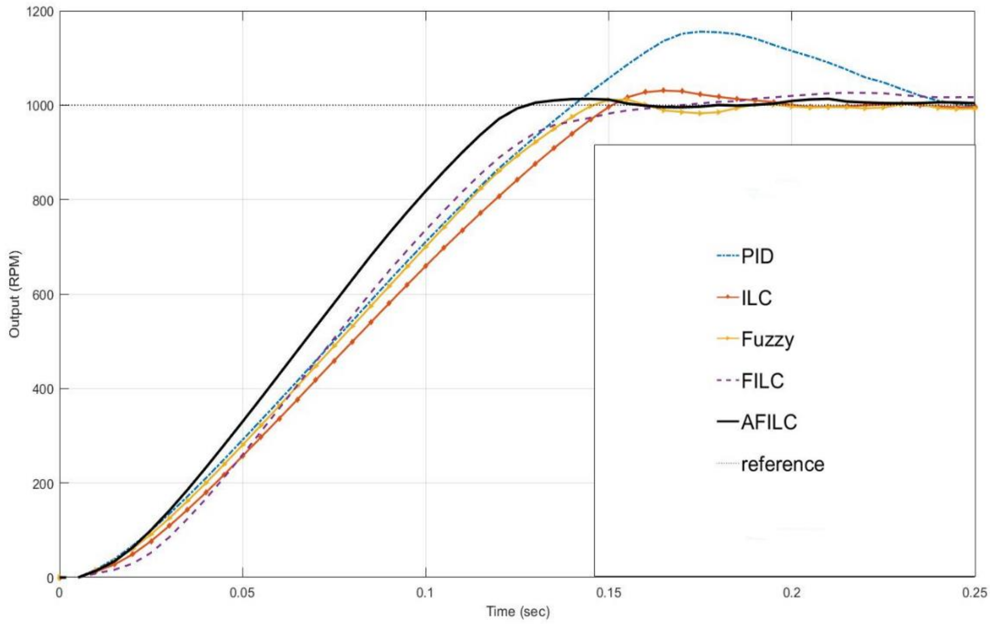


Figure 5.39. Transient performances of control methods

Tracking performance of the controllers are presented in Figure 5.40. The load rejection responses are summarized in Figure 5.41.

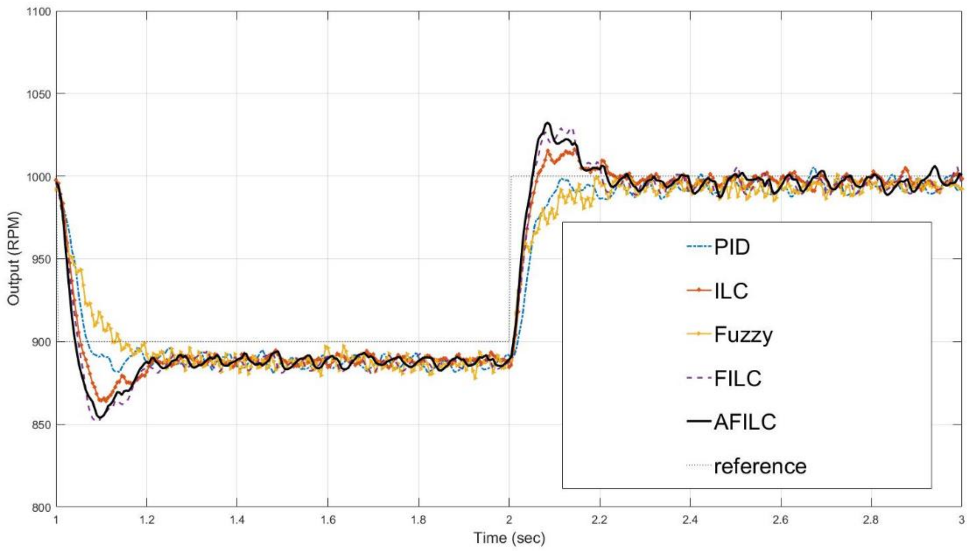


Figure 5.40. The tracking performance of control methods

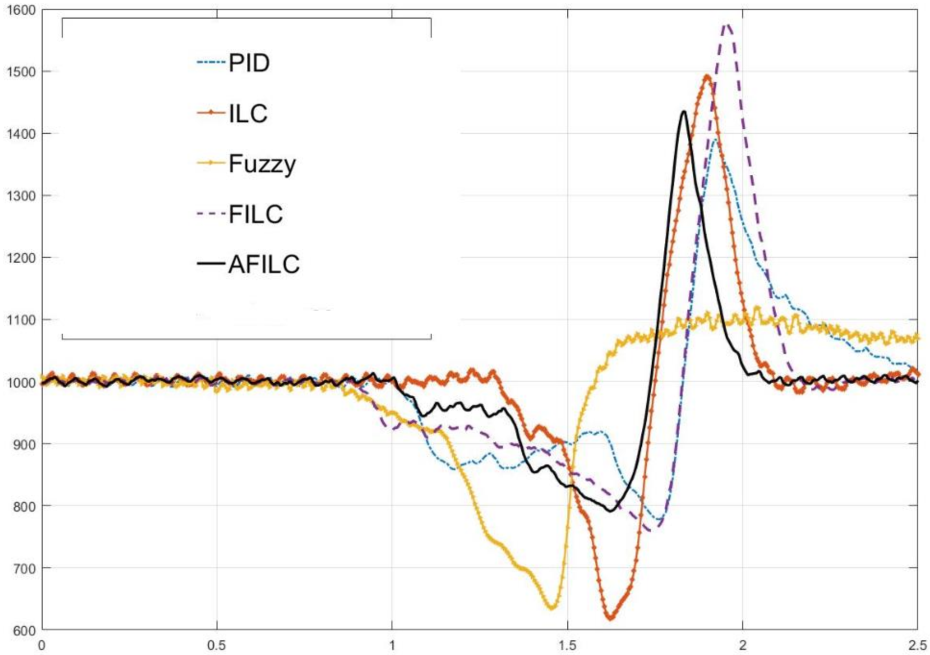


Figure 5.41. The distubance rejection performances of control methods

5.8. Summary

In this chapter, real-time experimental studies of five different control systems developed for the thesis have been made. Step test, tracking test and load test performances of each control system were examined in experimental studies. The experimental results are shown in graphics and the performance values of the system are given numerically in the tables.

6. CONCLUSION

In the present thesis, five different types of controllers have been examined with their experimental applications. The experimental studies of controllers have been carried out by applying to a DC motor.

First of all, these five different controller types were simulated with Matlab&Simulink. The control methods in these simulation studies are as follows, respectively:

- PID controller
- Iterative learning controller
- Fuzzy logic controller
- Fuzzy PID-type iterative learning controller
- Adaptive fuzzy PID-type iterative learning controller

Control methods were evaluated with the information obtained from simulation studies.

These studies have always progressed to improve their performance by improving the previous control method. The fuzzy logic control mechanism was developed for self-tuning the coefficients of conventional PID controllers. Thus, PID coefficients were calculated with the fuzzy method. The PID parameters required for the iterative learning control method were planned to be obtained by the fuzzy logic control method. Thus, it was aimed to combine two different intelligence control methods. Then, adaptive algorithms are proposed instead of scaling factors used for inputs and outputs of fuzzy logic controller. Hence, the structure of the control system was completed.

The adaptive fuzzy PID-type ILC was successfully applied to the DC motor. The PID-type ILC algorithm was used to adjust optimal control input for each iteration. The produced optimal control input from PID-type ILC makes the system error reduced. Compared to the experimental applicaiton results of other control methods, fuzzy PID-type ILC demonstrated much better performances. The adaptive fuzzy PID-type ILC algorithm, which will increase the performance of the fuzzy PID-type ILC algorithm and be an alternative to it, has also been developed.

As a result of this thesis, the developed fuzzy PID-type ILC algorithm and the adaptive fuzzy PID-type ILC algorithm developed as an alternative method were successful in the tests performed to be used instead of classical PID controllers.

6.1. Future Work

Robust control of Fuzzy PID-type ILC can be recommended as future work to cover uncertainties, parametric variations, disturbance and load variation with the stability analysis.

REFERENCES

Ahn, H. S., Chen, Y., & Moore, K. L. (2007). Iterative learning control: Brief survey and categorization. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, *37*(6), 1099-1121.

Amann, N., Owens, D. H., & Rogers, E. (1996). Iterative learning control for discrete-time systems with exponential rate of convergence. *IEE Proceedings-Control Theory and Applications*, *143*(2), 217-224.

Amini, A., & Nikraz, N. (2016). Proposing two defuzzification methods based on output fuzzy set weights. *International Journal of Intelligent Systems and Applications*, *8*(2), 1-12.

Arimoto, S. U. G. U. R. U., Kawamura, S. A. D. A. O., & Miyazaki, F. U. M. I. O. (1986). Convergence, stability and robustness of learning control schemes for robot manipulators. In *Proceedings of the International Symposium on Robot Manipulators on Recent trends in robotics: modeling, control and education* (pp. 307-316), October, Amsterdam, Netherlands.

Arimoto, S., Kawamura, S., & Miyazaki, F. (1984). Bettering operation of dynamic systems by learning: A new control theory for servomechanism or mechatronics systems. In *The 23rd IEEE Conference on Decision and Control* (pp. 1064-1069). IEEE, December, Las Vegas , NV, USA.

Ashraf, S., Muhammad, E., & Al-Habaibeh, A. (2008). Self-learning control systems using identification-based adaptive iterative learning controller. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, *222*(7), 1177-1187.

Åström, K. J., & Hägglund, T. (2004). Revisiting the Ziegler–Nichols step response method for PID control. *Journal of Process Control*, *14*(6), 635-650.

Bağiş, A. (2003). Determining fuzzy membership functions with tabu search—an application to control. *Fuzzy Sets and Systems*, *139*(1), 209-225.

Bien, Z., & Huh, K. M. (1989). Higher-order iterative learning control algorithm. In *IEE Proceedings D (Control Theory and Applications)* (Vol. 136, No. 3, pp. 105-112). IET Digital Library.

Bobyr, M. V., Milostnaya, N. A., & Kulabuhov, S. A. (2017). A method of defuzzification based on the approach of areas' ratio. *Applied Soft Computing*, *59*, 19-32

Chen, Y., & Moore, K. L. (2002). PI-type iterative learning control revisited. In *Proceedings of the 2002 American Control Conference (IEEE Cat. No. CH37301)* (Vol. 3, pp. 2138-2143). IEEE, Anchorage, AK, USA.

Chen, Y., & Wen, C. (1999). *Iterative Learning Control: Convergence, Robustness and Applications*. London: Springer.

Chen, Y., Wen, C., Gong, Z., & Sun, M. (1999). An iterative learning controller with initial state learning. *IEEE Transactions on Automatic Control*, *44*(2), 371-376.

D'Errico, G. E. (2001). Fuzzy control systems with application to machining processes. *Journal of Materials Processing Technology*, *109*(1-2), 38-43.

Eker, I. (2004). Experimental on-line identification of an electromechanical system. *ISA Transactions*, *43*(1), 13-22.

Eker, I. (2006). Sliding mode control with PID sliding surface and experimental application to an electromechanical plant. *ISA Transactions*, *45*(1), 109-118.

Eker, I., & Torun, Y. (2006). Fuzzy logic control to be conventional method. *Energy Conversion and Management*, *47*(4), 377-394.

Elshazly, O., El-Bardini, M., & El-Rabaie, N. M. (2012). Development of Self-Tuning Fuzzy Iterative LearningControl for Controlling a Mechatronic System. *International Journal of Information and Electronics Engineering*, *2*(4), 565.

Faruq, M. F. B. M. (2008). PID Controller Design for Controlling DC Motor Speed Using MATLAB Application. Universiti Malaysia Pahang, Electrical & Electronics Engineering.

Furat, M., & Eker, I. (2014). Second-order integral sliding-mode control with experimental application. *ISA transactions*, *53*(5), 1661-1669.

Furat, M., & Eker, İ. (2014). Computer-aided experimental modeling of a real system using graphical analysis of a step response data. *Computer Applications in Engineering Education*, *22*(4), 571-582.

Galichet, S., & Boukezzoula, R. (2008). Introduction to fuzzy systems, Guanrong Chen, Trung Tat Pham, Chapman & Hall/CRC, Applied Mathematics and Nonlinear Science Series, Taylor & Francis Group, ISBN 1-58488-531-9, 2006, 315p.

Hamdan, H., & Garibaldi, J. M. (2010). Adaptive neuro-fuzzy inference system (ANFIS) in modelling breast cancer survival. In *International Conference on Fuzzy Systems* (pp. 1-8). IEEE, Barcelona, Spain.

Hao, X., & Wang, D. (2017). Fuzzy PID Iterative learning control for a class of Nonlinear Systems with Arbitrary Initial Value. In *2016 7th International Conference on Education, Management, Computer and Medicine (EMCM 2016).* Atlantis Press, 304-311, Shenyang, China.

Hock, A., & Schoellig, A. P. (2019). Distributed iterative learning control for multi-agent systems. *Autonomous Robots*, *43*(8), 1989-2010.

Horiuchi, J. I., & Kishimoto, M. (2002). Application of fuzzy control to industrial bioprocesses in Japan. *Fuzzy sets and Systems*, *128*(1), 117-124.

Jee, S., & Koren, Y. (2004). Adaptive fuzzy logic controller for feed drives of a CNC machine tool. *Mechatronics*, *14*(3), 299-326.

Kaur, A., & Kaur, A. (2012). Comparison of mamdani-type and sugeno-type fuzzy inference systems for air conditioning system. *International Journal of Soft Computing and Engineering*, *2*(2), 323-325.

Kumar, V., & Mittal, A. P. (2010). Parallel fuzzy P+ fuzzy I+ fuzzy D controller: Design and performance evaluation. *International Journal of Automation and Computing*, *7*(4), 463-471.

Kumar, V., Mittal, A. P., & Singh, R. (2013). Stability analysis of parallel fuzzy P+ fuzzy I+ fuzzy D control systems. *International Journal of Automation and Computing*, *10*(2), 91-98.

Kuo, K. Y., & Lin, J. (2002). Fuzzy logic control for flexible link robot arm by singular perturbation approach. *Applied Soft Computing*, *2*(1), 24-38.

Lear, J. (1986). *Aristotle and logical theory*. CUP Archive.

Lee, H. S., & Bien, Z. (1996). Study on robustness of iterative learning control with non-zero initial error. *International Journal of Control*, *64*(3), 345-359.

Liang, M., Yeap, T., Rahmati, S., & Han, Z. (2002). Fuzzy control of spindle power in end milling processes. *International Journal of Machine Tools and Manufacture*, *42*(14), 1487-1496.

Lin, F. C., & Yang, S. M. (2003). Adaptive fuzzy logic-based velocity observer for servo motor drives. *Mechatronics*, *13*(3), 229-241.

Liu, Y., Li, L., Yang, X., & Tan, J. (2020). Enhanced kalman-filtering iterative learning control with application to a wafer scanner. *Information Sciences*, *541*, 152-165.

Lu Q,Fang, Y.C., Ren, X. (2014) Acceleration suppression of random errorson the initial state of iterative learning control, Acta Automatica Sinica, 40(7) : 1295-1302.

Ma, X., Guo, S., Xiao, N., Guo, J., Yoshida, S., Tamiya, T., & Kawanishi, M. (2012). Development of a novel robotic catheter manipulating system with fuzzy PID control. *International Journal of Intelligent Mechatronics and Robotics (IJIMR)*, *2*(2), 58-77.

Mamdani, E. H. (1974, December). Application of fuzzy algorithms for control of simple dynamic plant. In *Proceedings of the institution of electrical engineers* IET, Vol. 121, No. 12, pp. 1585-1588.

Markusson, O. (2001). *Model and system inversion with applications in nonlinear system identification and control* (Doctoral dissertation, Signaler, sensorer och system).

Mirinejad, H., Welch, K. C., & Spicer, L. (2012). A review of intelligent control techniques in HVAC systems. In *2012 IEEE Energytech* (pp. 1-5). IEEE.

Mitra, S., & Hayashi, Y. (2000). Neuro-fuzzy rule generation: survey in soft computing framework. *IEEE transactions on neural networks*, *11*(3), 748-768.

Moore, K. L. (2012). Iterative learning control for deterministic systems. Springer Science & Business Media.

Oppenheim, A. V., Willsky, A. S., & Hamid, S. (1997). Signals and systems, Processing series.

Owens, D. H., Amann, N., Rogers, E., & French, M. (2000). Analysis of linear iterative learning control schemes-a 2D systems/repetitive processes approach. *Multidimensional Systems and Signal Processing*, *11*(1-2), 125-177.

Ozbek, N. S., & Eker, İ. (2019). A Novel Modified Delay-Based Control Algorithm with an Experimental Application. *Information Technology and Control*, *48*(1), 90-103.

Ozbek, N. S., & Eker, İ. (2019). A Novel Modified Delay-Based Control Algorithm with an Experimental Application. *Information Technology and Control*, *48*(1), 90-103.

Özbek, N. S., & Eker, İ. (2015). An interactive computer-aided instructional strategy and assessment methods for system identification and adaptive control laboratory. *IEEE Transactions on Education*, *58*(4), 297-302.

Panagopoulos, H., Astrom, K. J., & Hagglund, T. (2002). Design of PID controllers based on constrained optimisation. *IEE Proceedings-Control Theory and Applications*, *149*(1), 32-40.

Phan, M. Q., Longman, R. W., & Morre, K. L. (2000). Unified formulation of linear iterative learning control. *Spaceflight Mechanics 2000*, 93-111.

Precup, R. E., Preitl, S., Clep, P. A., Ursache, I. B., Tar, J. K., & Fodor, J. (2008). Stable fuzzy control systems with iterative feedback tuning. In *2008*

International Conference on Intelligent Engineering Systems (pp. 287-292). IEEE, Miami, FL, USA.

Preitl, S., Precup, R. E., Rădac, M. B., Dragoş, C. A., Tar, J. K., & Fodor, J. (2008). On the Stable Design of Stable Fuzzy Control Systems with Iterative Learning Control. In *9th International Symposium of Hungarian Researchers on Computational Intelligence and Informatics, CINTI 2008* (pp. 345-360), Budapest, Hungary.

Prieto-Entenza, P. J., Cazarez-Castro, N. R., Aguilar, L. T., Cardenas-Maciel, S. L., & Lopez-Renteria, J. A. (2019). A lyapunov analysis for mamdani type fuzzy-based sliding mode control. *IEEE Transactions on Fuzzy Systems*, *28*(8), 1887-1895.

Qiu, K., Zheng, Z., & Chen, T. Y. (2017). Testing Proportional Integral Derivative (PID) Controller with Metamorphic Testing. In 2017 IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW) (pp. 88-89). IEEE, Toulouse, France.

Rajagiri, A. K., Rani, S., & Nawaz, S. S. (2019). Speed control of DC motor using fuzzy logic controller by PCI 6221 with MATLAB. In *E3S Web of Conferences* (Vol. 87, p. 01004). EDP Sciences.

Rogers, E., & Owens, D. H. (1992). *Stability Analysis for Linear Repetitive Processes* (Vol. 175). Berlin: Springer-Verlag.

Shen, D. (2018). Iterative learning control with incomplete information: A survey. *IEEE/CAA Journal of Automatica Sinica*, *5*(5), 885-901.

Sun, H., & Liu, L. (2002). A linear output structure for fuzzy logic controllers. *Fuzzy Sets and Systems*, *131*(2), 265-270.

Taylan, O., & Karagözoğlu, B. (2009). An adaptive neuro-fuzzy model for prediction of student’s academic performance. *Computers & Industrial Engineering*, *57*(3), 732-741.

Tong, S., Min, X., & Li, Y. (2020). Observer-based adaptive fuzzy tracking control for strict-feedback nonlinear systems with unknown control gain functions. *IEEE Transactions on Cybernetics*, *50*(9), 3903-3913.

Uchiyama, M. (1978). Formation of high-speed motion pattern of a mechanical arm by trial. *Transactions of the Society of Instrument and Control Engineers*, *14*(6), 706-712.

Verma, V., & Chauhan, S. (2019). Adaptive PID-fuzzy logic controller for brushless DC motor. In *2019 3rd International conference on Electronics, Communication and Aerospace Technology (ICECA)* (pp. 445-449). IEEE.

Verwoerd, M. H. (2004). Iterative learning control: A critical review. Ph.D. thesis, Univ. Twente, Twente, Netherlands.

Wai, R. J., Lin, C. M., & Hsu, C. F. (2004). Adaptive fuzzy sliding-mode control for electrical servo drive. *Fuzzy Sets and Systems*, *143*(2), 295-310.

Walia N, Singh H, Sharma A (2015) ANFIS: adaptive neuro-fuzzy inference system–a survey. Int. J. Comput. Appl. 123:32–38

Wang, L. X. (1994). Adaptive fuzzy systems and control. *Design and stability analysis*, Prentice Hall Englewood Cliffs, NJ, USA.

Wang, M., Bian, G., & Li, H. (2016). A new fuzzy iterative learning control algorithm for single joint manipulator. *Archives of Control Sciences*, *26*.

Wang, S., Liu, Y., Wang, Z., Dong, P., Cheng, Y., Xu, X., & Tenberge, P. (2019). Adaptive fuzzy iterative control strategy for the wet-clutch filling of automatic transmission. *Mechanical Systems and Signal Processing*, *130*, 164-182.

Wang, Y., Jin, Q., & Zhang, R. (2017). Improved fuzzy PID controller design using predictive functional control structure. *ISA Transactions*, *71*, 354-363

Wei, J., Hu, Y., & Sun, M. (2014). Adaptive iterative learning control for a class of nonlinear time-varying systems with unknown delays and input dead-zone. *IEEE/CAA Journal of Automatica Sinica*, *1*(3), 302-314.

Woo, Z. W., Chung, H. Y., & Lin, J. J. (2000). A PID type fuzzy controller with self-tuning scaling factors. *Fuzzy Sets and Systems*, *115*(2), 321-326.

Yan, J., Ryan, M. & Power, J. (1994). Using Fuzzy Logic. Prantice Hall, ISBN-13: 978-0131027329.

Yang, Z., & Seested, G. T. (2013). Time-Delay System Identification Using Genetic Algorithm–Part Two: FOPDT/SOPDT Model Approximation. *IFAC Proceedings Volumes*, *46*(20), 568-573.

Yu, X., He, W., Li, H., & Sun, J. (2020). Adaptive fuzzy full-state and output-feedback control for uncertain robots with output constraint. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*.

Zadeh, L. A., Klir, G. J., & Yuan, B. (1996). *Fuzzy Sets, Fuzzy Logic, and Fuzzy Systems: Selected Papers* (Vol. 6). World Scientific.

BIBLIOGRAPHY

I graduated from the department of Electrical and Electronics Engineering at Çukurova University. I am currently M.Sc. student in Electrical and Electronics Engineering department at Çukurova University in Adana, Turkey.