A COMPARATIVE STUDY OF CONVENTIONAL PID AND FUZZY-PID FOR DC MOTOR SPEED CONTROL

AHMAD MUHYIDDIN BIN YUSOF

A thesis submitted in
fulfillment of the requirement for the award of the
Degree of Master of Electrical Engineering

Faculty of Electrical and Electronic Engineering
Universiti Tun Hussein Onn Malaysia

JUNE 2013

ABSTRACT

Self Tuning Fuzzy PID and proportional-integral-derivative (PID) controllers are developed and compared for used in direct current (DC) motor. A simulation study of both controller for the armature voltage controlled DC motors is performed to overcome the appearrance of nonlinearities and uncertainties in the system. The fuzzy logic controller is designed according to fuzzy rules so that the systems are fundamentally robust. There are 49 fuzzy rules for self tuning of each parameter of FUZZY-PID controller. Fuzzy Logic is used to tune each parameter of the proportional, integral and derivative (K_P,K_I,K_D) gains of the PID controller. The FLC has two inputs. One is the motor speed error between the reference and actual speed and the second is changed in speed error (rate of change error). The output of the FLC i.e the parameter of PID controller are used to control the speed of the DC motor [1]. Different types of membership functions such as triangular, trapezoidal, gaussian are evaluated in the fuzzy control and the best performance will be used in FUZZY PID for comparative analysis with the conventional PID. The membership function and the rules have been defined using FIS editor given in MATLAB. Three different scenario are simulated, which are step response, load disturbances and noise disturbance. The FUZZY-PID controller has been tuned by trial and error and performance parameters are Rise time, Settling Time, Percent Overshoot and Integral Absolute error [21].

ABSTRAK

Pengawal penalaan sendiri Fuzzy PID dan pengawal terbitan-kamiran-berkadaran (PID) telah dibangunkan dan dibandingkan untuk digunakan pada motor (DC). Satu kajian simulasi kedua-dua pengawal voltan angker terkawal DC motor telah dijalankan untuk mengatasi ketaklelurusan dan ketidaktentuan di dalam sistem. Pengawal Fuzzy logik direka berdasarkan peraturan fuzzy supaya sistem asasnya kukuh. Terdapat 49 peraturan Fuzzy untuk penalaan sendiri bagi setiap parameter pengawal Fuzzy PID. Fuzzy logik digunakan untuk untuk menala setiap parameter gandaan terbitan, kamiran, berkadaran ($K_{p,}$ K_i, K_d) bagi pengawal PID. FLC mempunyai dua input. Pertama adalah ralat kelajuan motor diantara rujukan dan kelajuan sebenar, dan keduanya adalah perubahan dalam ralat kelajuan (kadar perubahan ralat). Keluaran daripada FLC iaitu paramater pengawal PID digunakan untuk mengawal kelajuan motor DC [1]. Jenis fungsi keahlian yang berbeza seperti segitiga, trapezoid, gaussian dinilai dalam kawalan fuzzy dan fungsi keahlian yang menunjukan prestasi yang terbaik akan digunakan di dalam Fuzzy PID untuk analisis perbandingan dengan PID konvensional. Fungsi keahlian dan peraturan telah ditentukan dengan menggunakan penyunting FIS didalam MATLAB. Tiga senario yang berbeza telah di simulasikan seperti langkah sambutan, gangguan bebanan dan gangguan bunyi. Pengawal Fuzzy PID telah ditala dengan kaedah cuba jaya dan parameter prestasi adalah masa naik, masa pengenapan, peratus terlajak, lajak turun dan ralat kamiran mutlak [21].

LIST OF CONTENTS

DESCRIPTION **PAGE**

CHAPTER 1 **INTRODUCTION**

CHAPTER 2 LITERATURE REVIEW

CHAPTER 3 METHODOLOGY

CHAPTER 4 RESULT AND DISCUSSION

CHAPTER 5 CONCLUSION AND FUTURE WORKS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOL

Abbreviation and symbols used in this thesis are tabulated below:

IAE	Integral Absolute Error
PID	Porportional Integral Derivative
DE	Different of Error
ce	Change Of Error
e	Error
FLC	Fuzzy Logic Controller
Mf	Membership Function
DC	Direct Current
AC	Alternating Current
PV	Process Variable
SP	Set Point
MV	Manipulated Variable
SISO	Single in Single out
MIMO	Multi input multi out
K_p	Porportional Gain
K_i	Integral Gain
K_d	Derivative gain

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

The development of high performance motor drives is very important in industrial as well as other purpose applications such as steel rolling mills, electric trains and robotics. Generally, a high performance motor drive system must have good dynamics speed command tracking and load regulating response to perform task. Dc drives, because of their simplicity, ease of application, high reliabilities, flexibilities and favorable cost have long been a backbone of industrial application, robot manipulators and home appliances where speed and position control of motor are required. DC drives are less complex with a single power conversion from AC to DC. Again the speed torque characteristics of DC motors are much more superior to that of AC motors. A DC motors provide excellent control of speed for acceleration and deceleration. Dc drives are normally less expensive for most horsepower ratings. Dc motors have a long tradition of use as adjustable speed machines and a wide range of options have evolved for this purpose. In these applications, the motor should be precisely controlled to give desired performance. The controller of the speed that are conceived for goal to control the speed of DC motor to execute one variety of tasks, is of several conventional and numeric controller types, the controllers can be: proportional integral (PI), proportional integral derivative (PID) Fuzzy Logic Controller (FLC) or the combination between them: Fuzzy-Neural Networks, Fuzzy- Genetics Algorithm, Fuzzy-Antz Colony, Fuzzy-Swarm [10]. The proportional – integral – derivative (PID) controller operates the majority of the control system in the world. It has been reported that more than 95% of the controllers in the industrial process control applications are of PID type as no other controller match the simplicity, clear functionality, applicability and ease of use offered by the PID controller [4],[5]. PID controllers provide robust and reliable performance for most systems if the PID parameters are tuned properly.

The major problems in applying a conventional control algorithm (PI,PD,PID) in a speed controller are the effects of non-linearity in a DC motor. The nonlinear characteristics of a DC motor such as saturations and fictions could degrade the performance of conventional controllers [2], [3]. Generally, an accurate nonlinear model of an actual DC motor is difficult to find and parameter obtained from systems identification may be only approximated values. The field of Fuzzy control has been making rapid progress in recent years. Fuzzy logic control (FLC) is one of the most succesful applications of fuzzy set theory, introduced by L.A Zadeh in 1973 and applied (Mamdani 1974) in an attempt to control system that are structurally difficult to model. Since then, FLC has been an extremely active and fruitful research area with many industrial applications reported [6]. In the last three decades, FLC has evolved as an alternative or complementary to the conventional control strategies in various engineering areas. Fuzzy control theory usually provides non-linear controllers that are capable of performing different complex non-linear control action, even for uncertain nonlinear systems. Unlike conventional control, designing a FLC does not require precise knowledge of the system model such as the poles and zeroes of the system transfer functions. Imitating the way of human learning, the tracking error and the rate change of the error are two crucial inputs for the design of such a fuzzy control system [8], [9].

1.1 Advantages Of Using Fuzzy Technique

The advantages provided by a FLC are listed below:

- It is simple to design.
- It provides a hint of human intelligence to the controller.
- It is cost effective.
- No mathematical modeling of the system is required.
- Linguistic variables are used instead of numerical ones.
- Non-linearity of the system can be handled easily.
- System response is fast.
- Reliability of the system is increased.
- High degree of precision is achieved.

These advantages allow fuzzy controllers can be used in systems where description of the process and identification of the process parameters with precision is highly difficult. Hence it provides a fuzzy characteristic to the control mechanism [13].

1.2 Difficulties Of Using Fuzzy Technique

Fuzzy logic is gaining widespread acceptance in the control engineering community because of its continued success in control applications. However, certain inherent difficulties of the approach are restricting its grow. The following are some of the difficulties, which face its application development [11]:

- Difficulties in developing fuzzy rules by hand for large systems.
- Difficulties in selecting appropriate membership function shapes.
- Difficulties in fine tuning fuzzy solutions for specific levels of accuracy, and guaranteeing the reliability/robustness of solutions. The trial and error

method is still the basic method in improving the expert knowledge towards developing tuned and stable fuzzy controllers.

1.3 Applications Of Fuzzy Logic

Video Camcorder	:	Determine best focusing and lighting when there is Movement in the picture.
Washing Machine	:	Adjust washing cycle by judging the dirt, size of the load, and type of fabric.
Television	:	Adjust brightness, color and contrast of picture to please viewers.
Motor Control	:	Improve the accuracy and range of motion control under unexpected conditions.
Subway Train	:	Increase the stable drive and enhance the stop accuracy evaluating the passenger traffic conditions. Provide a smooth start and smooth stop.

1.4 Project Objective

The objectives of this project consist of five points that will be discussed:-

i. To model a Separately exited DC motor
ii. To control the DC motor speed with conventional controlling (PID) methods
iii. To control the DC motor speed with FUZZY-PID controller
iv. To analyze the sensitivity, evaluate and compare the effects of different types of MFs in the Fuzzy PID DC motor speed control
v. Compare the different speed controlling techniques.

1.5 Problem Statement

In classical control techniques PID controller was used as a standard control structure. Due to parameter variation and external disturbance in the process the performance of the industrial machinery is greatly distorted and the efficiency is reduced. The new technique which uses fuzzy and PID controllers is considered as the extension of the conventional technique, because it preserves the linear structure of PID controller. These controllers are designed using the basic principle of fuzzy logic control to obtain a new controller that possesses analytical formulas similar to digital PID controllers. Fuzzy PID controllers have variable control gains in their linear structure. These variable gains are nonlinear function of the errors and changing rates of error signals. These variable gains help in improving the overall performance due to their characteristics features like self-tuned mechanism which can adapt to rapid changes of the errors and rate of change of error caused by time delay effects, nonlinearities and uncertainties of the process[12].

An often remarked disadvantage of the methods based on the fuzzy logic is the lack of appropriate tools for analysing the controllers performance, such as stability, optimality, robustness, etc. The most important is to make a good choice of rule based and parameters of membership functions because Fuzzy Logic control is a control algorithm based on a linguistic control strategy, which derived from expert knowledge into an automatic control strategy. The operation of a FLC is based on qualitative knowledge about the system being controlled. An adequate knowledge and experience must be applied to ensure the system can give a good response.

PID controller can not be applied with the system which have a fast change of parameters, because it would require the change of PID constant in the time. It is necessary to further study the possible combinations of PID and FUZZY controller. It means that the system can be well controlled by PID which is supervised by a fuzzy system [13].

A number of different types of of membership functions (MFs) have been proposed for fuzzy control system. There is also provision to custom-design MFs in some fuzzy control software tools. The literature on fuzzy control indicates application of different types of MFs. For example, in modern neuro-fuzzy control, particularly where neural network techniques are used to tune and implement a fuzzy controller, sigmoid type MFs have been found very useful. Sometimes, MF types are hybrided for the input and output fuzzy variables. Although trapezoidal type MF has often been used in fuzzy control literature, triangular MFs are most commonly used almost intuitively for all the variables. Is there any justification for using triangular type MF compared to other types of MFs? Unfortunately, so far in the literature, there has been no systematic analysis, evaluation and comparison of fuzzy control with different types of MFs in order to established the superiority of a particularly type MFs [17].

1.6 Scope Of Project

The scope of this project is :

i. Design a PID and FUZZY PID controller to control the speed of the DC motor using Simulink in MATLAB
ii. Design different Membership functions (MFs) and rules using Fuzzy Toolbox in MATLAB
iii. Apply different membership function in FUZZY-PID controller
iv. Study the performance of a FUZZY-PID controller and compare it with the conventional control approach.

CHAPTER 2

LITERATURE REVIEW

2.1 Background works

Inspite of the development of power electronics resources, the direct current machines are becoming more and more useful in so far as they have found wide applications i.e. automobiles industry (electric vehicle), the electric traction in the multi-machine systems etc. The speed of DC motor can be adjusted to a great extent so as to provide easy control and high performance. There are several conventional and numeric controllers intended for controlling the DC motor speed: PID controllers, fuzzy logic controllers; or the combination between them, fuzzy neural networks etc. The nonlinearity of the series/shunt-connected motors complicates their use in applications that require automatic speed control. Major problems in applying a conventional control (Liu et. al 1999) algorithm in a speed controller are the effects of non-linearity in a DC motor. One of intelligent technique, fuzzy logic by Zadeh is applied for controller design in many applications. The advantage of fuzzy control methods is the fact that they are not sensitive to the accuracy of the dynamical model. In motion control systems, fuzzy logic can be considered as an alternative approach to conventional feedback control. It has been demonstrated in the literature that dynamic performance of electric drives as well as robustness with regard to parameter variations can be improved by adopting the non-linear speed control techniques. Fuzzy control is a non-linear control and it allows the design of optimized non-linear controller to improve the dynamic performance of conventional regulators. Several works are reported in literature (Iracleous and Alexandris 1995; B. Singh et. al 2000;

Montiel et. al 2007) where conventional controller is combined with the fuzzy controller to improve the response of the DC motor under non-linearity, load disturbances, parameter variations etc. From the application of fuzzy control arise two problems: how to select the fuzzy control rules and how to set the membership functions. Two approaches are normally used to accomplish this task. One consists of acquiring knowledge directly from skilled operators and translates it into fuzzy rules. This process, however, can be difficult to implement and time consuming. As an alternative, fuzzy rules can be obtained through machine learning techniques, where the knowledge of the process is automatically extracted or induced from sample cases or examples. Many machine learning methods developed for building crisp logic systems can be extended to learn fuzzy rules [7].

2.2 Basic control theory

Control theory is an interdisciplinary branch of engineering and mathematics, that deals with the behavior of dynamical systems. The desired output of a system is called the reference. When one or more output variables of a system need to follow a certain reference over time, a controller manipulates the inputs to a system to obtain the desired effect on the output of the system.

Control theory is a

“Theory that deals with influencing the behavior of dynamical systems an interdisciplinary subfield of science, which originated in engineering and mathematics, and evolved into use by the social sciences, like psychology, sociology and criminology”.

Let take an example of automobile's cruise control, which is a device designed to maintain a constant vehicle speed; the desired or reference speed, provided by the driver. The system in this case is the vehicle. The system output is the vehicle speed, and the control variable is the engine's throttle position which influences engine torque output.

A primitive way to implement cruise control is simply to lock the throttle position when the driver engages cruise control. However, on mountain terrain, the vehicle will slow down going uphill and accelerate going downhill. In fact, any parameter different than what was assumed at design time will translate into a proportional error in the output velocity, including exact mass of the vehicle, wind resistance, and tire pressure. This type of controller is called an open-loop controller because there is no direct connection between the output of the system (the vehicle's speed) and the actual conditions encountered; that is to say, the system does not and can not compensate for unexpected forces.

In a closed-loop control system, a sensor monitors the output (the vehicle's speed) and feeds the data to a computer which continuously adjusts the control input (the throttle) as necessary to keep the control error to a minimum (that is, to maintain the desired speed). Feedback on how the system is actually performing allows the controller (vehicle's on board computer) to dynamically compensate for disturbances to the system, such as changes in slope of the ground or wind speed. An ideal feedback control system cancels out all errors, effectively mitigating the effects of any forces that might or might not arise during operation and producing a response in the system that perfectly matches the user's wishes. In reality, this cannot be achieved due to measurement errors in the sensors, delays in the controller, and imperfections in the control input [18].

2.3 Classical control theory

To avoid the problems of the open-loop controller, control theory introduces feedback closed-loop controller uses feedback to control states or outputs of a dynamical system. Its name comes from the information path in the system: process inputs (e.g. voltage applied to an electric motor) have an effect on the process outputs (e.g. velocity or torque of the motor),which is measured with sensors and processed by the controller; the result (the control signal) is used as input to the process, closing the loop.

Closed-loop controllers have the following advantages over open-loop controllers:

1. Disturbance rejection (such as unmeasured friction in a motor)
2. Guaranteed performance even with model uncertainties, when the model structure
3. Does not match perfectly the real process and the model parameters are not exact
4. Unstable processes can be stabilized
5. Reduced sensitivity to parameter variations
6. Improved reference tracking performance

In some systems, closed-loop and open-loop control are used simultaneously. In such systems, the open-loop control is termed feedforward and serves to further improve reference tracking performance [18].

2.3.1 Control loop basics

A familiar example of a control loop is the action taken when adjusting hot and cold faucet valves to maintain the faucet water at the desired temperature. This typically involves the mixing of two process streams, the hot and cold water. The person touches the water to sense or measure its temperature. Based on this feedback they perform a control action to adjust the hot and cold water valves until the process temperature stabilizes at the desired value. Sensing water temperature is analogous to taking a measurement of the process value or process variable (PV). The desired temperature is called the setpoint (SP). The input to the process (the water valve position) is called the manipulated variable (MV). The difference between the temperature measurement and the setpoint is the error (e), that quantifies whether the water is too hot or too cold and by how much.

After measuring the temperature (PV), and then calculating the error, the controller decides when to change the tap position (MV) and by how much. When the controller first turns the valve on, they may turn the hot valve only slightly if warm water is desired, or they may open the valve all the way if very hot water is desired. This is an example of a simple proportional control. In the event that hot water does not arrive quickly, the controller may try to speed-up the process by opening up the hot water valve more-and-more as time goes by. This is an example of an integral control. By using only the proportional and integral control methods, it is possible that in some systems the water temperature may oscillate between hot and cold, because the controller is adjusting the valves too quickly and over-compensating or overshooting the setpoint.

In the interest of achieving a gradual convergence at the desired temperature (SP), the controller may wish to damp the anticipated future oscillations. So in order to compensate for this effect, the controller may elect to temper their adjustments. This can be thought of as a derivative control method.

Making a change that is too large when the error is small is equivalent to a high gain controller and will lead to overshoot. If the controller were to repeatedly make changes that were too large and repeatedly overshoot the target, the output would oscillate around the setpoint in either a constant, growing, or decaying sinusoid. If the oscillations increase with time then the system is unstable, whereas if they decrease the system is stable. If the oscillations remain at a constant magnitude the system is marginally stable. A human would not do this because we are adaptive controllers, learning from the process history; however, simple PID controllers do not have the ability to learn and must be set up correctly. Selecting the correct gains for effective control is known as tuning the controller.

If a controller starts from a stable state at zero error (PV = SP), then further changes by the controller will be in response to changes in other measured or unmeasured inputs to the process that impact on the process, and hence on the PV. Variables that impact on the process other than the MV are known as disturbances. Generally controllers are used to reject disturbances and/or implement setpoint changes. Changes in feedwater temperature constitute a disturbance to the faucet temperature control process. In theory, a controller can be used to control any process which has a measurable output (PV), a known ideal value for that output (SP) and an input to the process (MV) that will affect the relevant PV. Controllers are used in industry to regulate temperature, pressure, flow rate, chemical composition, speed and practically every other variable for which a measurement exists. Automobile cruise control is an example of a process which utilizes automated control. PID controllers are the controllers of choice for many of these applications, due to their well-grounded theory, established history, simplicity, and simple setup and maintenance requirements. A common closed-loop controller architecture is the PID controller [18].

2.3.2 Closed-loop transfer function

The output of the system y(t) is fed back through a sensor measurement F to the reference value r(t). The controller C then takes the error e (difference) between the reference and the output to change the inputs u to the system under control P. This is shown in the figure 2.1. This kind of controller is a closed-loop controller or feedback controller.

This is called a single-input-single-output (SISO) control system; MIMO (i.e. Multi-Input-Multi-Output) systems, with more than one input/output, are common. In such cases variables are represented through vectors instead of simple scalar values. For some distributed parameter systems the vectors may be infinite-dimensional (typically functions).

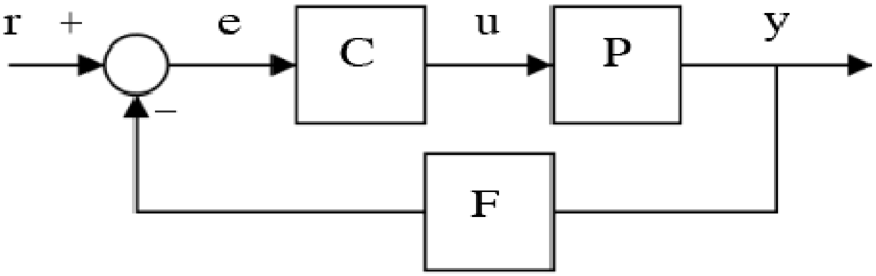


Figure 2.1: Closed Loop Control System

If we assume the controller C, the plant P, and the sensor F are linear and time-invariant (i.e.: elements of their transfer function C(s), P(s), and F(s) do not depend on time), the systems above can be analyzed using the Laplace transform on the variables. This gives the following relations:

$$Y(s) = P(s)U(s) \tag{2.1}$$

$$U(s) = C(s)E(s) \tag{2.2}$$

$$E(s) = R(s) - F(s)Y(s) \tag{2.3}$$

Solving for Y(s) in terms of R(s) gives:

$$Y(s) = \left(\frac{P(s)C(s)}{1+F(s)P(s)C(s)}\right)R(s) = H(s)R(s) \tag{2.4}$$

$$H(s) = \frac{P(s)C(s)}{1+F(s)P(s)C(s)} \tag{2.5}$$

The above expression is referred to as the closed-loop transfer function of the system. The numerator is the forward (open-loop) gain from r to y, and the denominator is one plus the gain in going around the feedback loop, the so-called loop gain. If |P(s)C(s)| >> 1, i.e. it has a large norm with each value of s, and if $|F(s)| \approx 1$, then Y(s) is approximately equal to R(s). This simply means setting the reference to control the output [18].

2.4 PID Controller

2.4.1 Introduction to PID controller

A proportional–integral–derivative controller (PID controller) is a generic control loop feedback mechanism (controller) widely used in industrial control systems – a PID is the most commonly used feedback controller. A PID controller calculates an "error" value as the difference between a measured process variable and a desired setpoint. The controller attempts to minimize the error by adjusting the process control inputs. In the absence of knowledge of the underlying process, PID controllers are the best controllers. However, for best performance, the PID parameters used in the calculation must be tuned according to the nature of the system – while the design is generic, the parameters depend on the specific system.

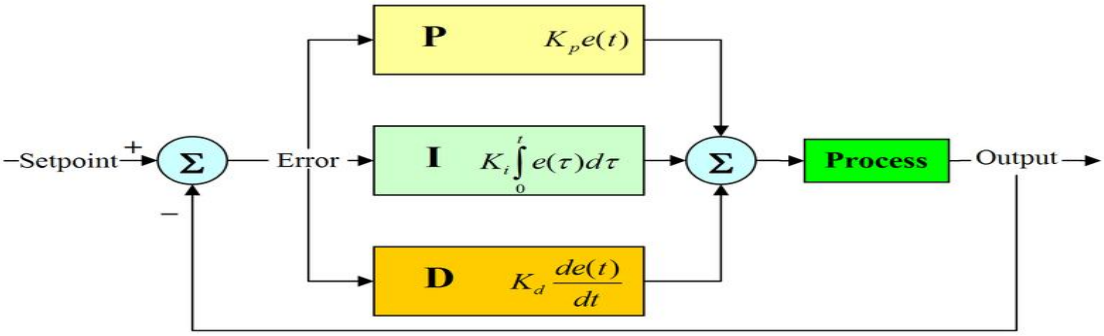


Figure 2.2: Block Diagram of PID Controller

The PID controller calculation (algorithm) involves three separate parameters, and is accordingly sometimes called three-term control: the proportional, the integral and derivative values, denoted P, I, and D. The proportional value determines the reaction to the current error, the integral value determines the reaction based on the sum of recent errors, and the derivative value determines the reaction based on the rate at which the error has been changing. The weighted sum of these three actions is used to adjust the process via a control element such as the position of a control valve or the power supply of a heating element. Heuristically, these values can be interpreted in terms of time: P depends on the present error, I on the accumulation of past errors, and D is a prediction of future errors, based on current rate of change.

By tuning the three constants in the PID controller algorithm, the controller can provide control action designed for specific process requirements. The response of the controller can be described in terms of the responsiveness of the controller to an error, the degree to which the controller overshoots the setpoint and the degree of system oscillation. Note that the use of the PID algorithm for control does not guarantee optimal control of the system or system stability.

Some applications may require using only one or two modes to provide the appropriate system control. This is achieved by setting the gain of undesired control outputs to zero. A PID controller will be called a PI, PD, P or I controller in the absence of the respective control actions. PI controllers are fairly common, since derivative action is sensitive to measurement noise, whereas the absence of an integral value may prevent the system from reaching its target value due to the control action. More detail on PID control theory [18].

2.4.2 PID control theory

The PID controller is probably the most-used feedback control design. PID is an acronym for Proportional-Integral-Derivative, referring to the three terms operating on the error signal to produce a control signal. If u(t) is the control signal sent to the system, y(t) is the measured output and r(t) is the desired output, and tracking error e(t) = r(t) − y(t), a PID controller has the general form

$$u(t) = K_p\, e(t) + K_I e(t) + K_D \frac{d}{dt} e(t) \tag{2.6}$$

The desired closed loop dynamics is obtained by adjusting the three parameters K_p , K_I and K_D, often iteratively by "tuning" and without specific knowledge of a plant model. Stability can often be ensured using only the proportional term. The integral term permits the rejection of a step disturbance (often a striking specification in process control). The derivative term is used to provide damping or shaping of the response. PID controllers are the most well established class of control systems: however, they cannot be used in several more complicated cases, especially if MIMO systems are considered.

Applying Laplace transformation results in the transformed PID controller equation

$$U(s) = K_p e(s) + K_I \frac{1}{s} e(s) + K_D s e(s) \qquad (2.7)$$

$$U(s) = \left(K_p + K_I \frac{1}{s} + K_D s\right) e(s) \qquad (2.8)$$

with the PID controller transfer function

$$C(s) = (K_p + K_I \frac{1}{s} + K_d s) \qquad (2.9)$$

In other words, The PID control scheme is named after its three correcting terms, whose sum constitutes the manipulated variable (MV). Hence:

$$MV(t) = P_{out} + I_{out} + D_{out} \qquad (2.10)$$

where Pout, Iout, and Dout are the contributions to the output from the PID controller from each of the three terms, as defined below.

2.4.3 Proportional term

The proportional term (sometimes called gain) makes a change to the output that is proportional to the current error value. The proportional response can be adjusted by multiplying the error by a constant Kp, called the proportional gain.

The proportional term is given by:

$$P_{out} = K_p e(t) \tag{2.11}$$

where

Pout: Proportional term of output
Kp: Proportional gain, a tuning parameter
e: Error = SP − PV
t: Time or instantaneous time (the present)

A high proportional gain results in a large change in the output for a given change in the error. If the proportional gain is too high, the system can become unstable (see the section 2.5 on loop tuning). In contrast, a small gain results in a small output response to a large input error, and a less responsive (or sensitive) controller. If the proportional gain is too low, the control action may be too small when responding to system disturbances.

In the absence of disturbances, pure proportional control will not settle at its target value, but will retain a steady state error (known as droop) that is a function of the proportional gain and the process gain. Specifically, if the process gain – the long-term drift in the absence of control, such as cooling of a furnace towards room temperature – is denoted by G and assumed to be approximately constant in the error, then the droop is when this constant gain equals the proportional term of the output, Pout, which is linear in the error, G = Kpe, so e = G / Kp. This is when the proportional term, which is pushing the parameter towards the set point, is exactly offset by the process gain, which is pulling the parameter away from the set point. If the process gain is down, as in cooling, then the steady state will be below the set point, hence the term "droop".

Only the drift component (long-term average, zero-frequency component) of process gain matters for the droop – regular or random fluctuations above or below the drift cancel out. The process gain may change over time or in the presence of external changes, for example if room temperature changes, cooling may be faster or slower.

Droop is proportional to process gain and inversely proportional to proportional gain, and is an inevitable defect of purely proportional control. Droop can be mitigated by adding a bias term (setting the setpoint above the true desired value), or corrected by adding an integration term (in a PI or PID controller), which effectively computes a bias adaptively.

Despite the droop, both tuning theory and industrial practice indicate that it is the proportional term that should contribute the bulk of the output change.

2.4.4 Integral term

The contribution from the integral term (sometimes called reset) is proportional to both the magnitude of the error and the duration of the error. Summing the instantaneous error over time (integrating the error) gives the accumulated offset that should have been corrected previously. The accumulated error is then multiplied by the integral gain and added to the controller output. The magnitude of the contribution of the integral term to the overall control action is determined by the integral gain, Ki.

The integral term is given by:

$$I_{out} = K_i \int_0^t e(\tau) d\tau \qquad (2.12)$$

where

Iout: Integral term of output
Ki: Integral gain, a tuning parameter
e: Error = SP − PV
t: Time or instantaneous time (the present)
τ: a dummy integration variable

The integral term (when added to the proportional term) accelerates the movement of the process towards setpoint and eliminates the residual steady-state error that occurs with a proportional only controller. However, since the integral term is responding to accumulated errors from the past, it can cause the present value to overshoot the setpoint value (cross over the setpoint and then create a deviation in the other direction). For further notes regardingintegral gain tuning and controller stability, see the section on loop tuning [18].

2.4.5 Derivative term

The rate of change of the process error is calculated by determining the slope of the error over time (i.e., its first derivative with respect to time) and multiplying this rate of change by the derivative gain Kd. The magnitude of the contribution of the derivative term (sometimes called rate) to the overall control action is termed the derivative gain, Kd.

The derivative term is given by:

$$D_{out} = K_d \frac{d}{dt} e(t) \qquad (2.13)$$

where

Dout: Derivative term of output
Kd: Derivative gain, a tuning parameter
e: Error = SP – PV
t: Time or instantaneous time (the present)

The derivative noticeable close term slows the rate of change of the controller output and this effect is most to the controller setpoint. Hence, derivative control is used to reduce the magnitude of the overshoot produced by the integral component and improve the combined controller-process stability.

However, differentiation of a signal amplifies noise and thus this term in the controller is highly sensitive to noise in the error term, and can cause a process to become unstable if the noise and the derivative gain are sufficiently large. Hence an approximation to a differentiator with a limited bandwidth is more commonly used. Such a circuit is known as a Phase-Lead compensator.

The proportional, integral, and derivative terms are summed to calculate the output of the PID controller. Defining u(t) as the controller output, the final form of the PID algorithm is:

$$\boldsymbol{u(t) = MV(t) = K_p e(t) + K_i \int_0^t e(\tau) + K_d \frac{d}{dt} e(t)} \qquad (2.14)$$

where the tuning parameters are:

Proportional gain, Kp

Larger values typically mean faster response since the larger the error, the larger the proportional term compensation. An excessively large proportional gain will lead to process instability and oscillation.

Integral gain, Ki

Larger values imply steady state errors are eliminated more quickly. The trade-off is larger overshoot: any negative error integrated during transient response must be integrated away by positive error before reaching steady state.

Derivative gain, Kd

Larger values decrease overshoot, but slow down transient response and may lead to instability due to signal noise amplification in the differentiation of the error.

2.5 Loop Tuning

Tuning a control loop is the adjustment of its control parameters (gain/proportional band, integral gain/reset, derivative gain/rate) to the optimum values for the desired control response. Stability (bounded oscillation) is a basic requirement, but beyond that, different systems have different behavior, different applications have different requirements, and some desiderata conflict. Further, some processes have a degree of non-linearity and so parameters that work well at full-load conditions don't work when the process is starting up from no-load; this can be corrected by gain scheduling (using different parameters in different operating regions). PID controllers often provide acceptable control even in the absence of tuning, but performance can generally be improved by careful tuning, and performance may be unacceptable with poor tuning.

PID tuning is a difficult problem, even though there are only three parameters and in principle is simple to describe, because it must satisfy complex criteria within the limitations of PID control. There are accordingly various methods for loop tuning, and more sophisticated techniques are the subject of patents; this section describes some traditional manual methods for loop tuning.

2.5.1 Stability

If the PID controller parameters (the gains of the proportional, integral and derivative terms) are chosen incorrectly, the controlled process input can be unstable, i.e. its output diverges, with or without oscillation, and is limited only by saturation or mechanical breakage. Instability is caused by excess gain, particularly in the presence of significant lag.

Generally, stability of response (the reverse of instability) is required and the process must not oscillate for any combination of process conditions and setpoints, though sometimes marginal stability (bounded oscillation) is acceptable or desired.

2.5.2 Optimum behavior

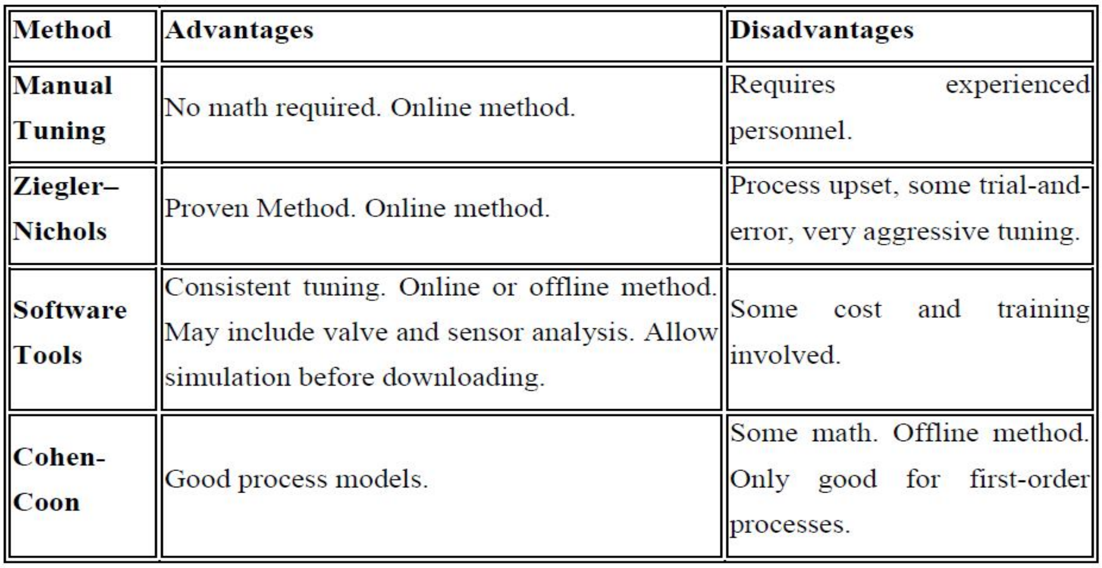
The optimum behavior on a process change or setpoint change varies depending on the application.

Two basic desiderata are regulation (disturbance rejection – staying at a given setpoint) and command tracking (implementing setpoint changes) – these refer to how well the controlled variable tracks the desired value. Specific criteria for command tracking include rise time and settling time. Some processes must not allow an overshoot of the process variable beyond the setpoint if, for example, this would be unsafe. Other processes must minimize the energy expended in reaching a new setpoint.

2.5.3 Tuning methods

There are several methods for tuning a PID loop. The most effective methods generally involve the development of some form of process model, then choosing P, I, and D based on the dynamic model parameters. Manual tuning methods can be relatively inefficient, particularly if the loops have response times on the order of minutes or longer. The choice of method will depend largely on whether or not the loop can be taken "offline" for tuning, and the response time of the system. If the system can be taken offline, the best tuning method often involves subjecting the system to a step change in input, measuring the output as a function of time, and using this response to determine the control parameters.

Table 2.1: Selection of Tuning Method

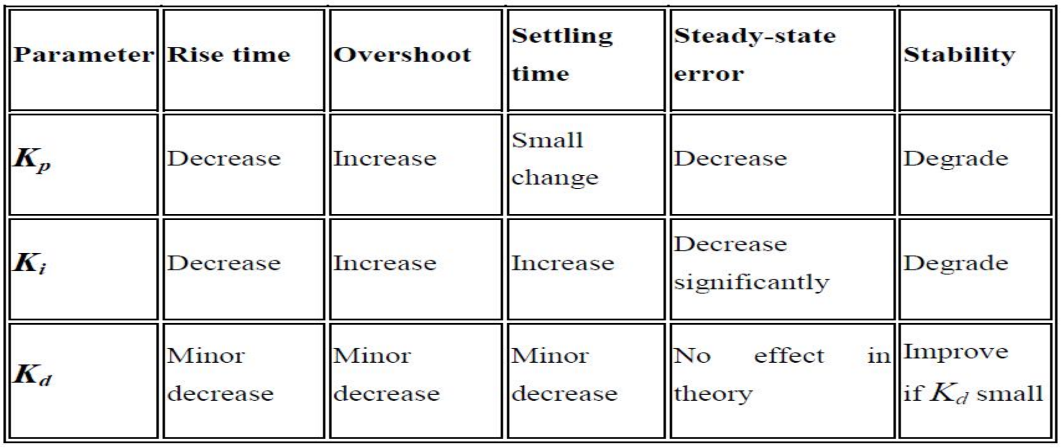


Method	Advantages	Disadvantages
Manual Tuning	No math required. Online method.	Requires experienced personnel.
Ziegler–Nichols	Proven Method. Online method.	Process upset, some trial-and-error, very aggressive tuning.
Software Tools	Consistent tuning. Online or offline method. May include valve and sensor analysis. Allow simulation before downloading.	Some cost and training involved.
Cohen-Coon	Good process models.	Some math. Offline method. Only good for first-order processes.

2.5.4 Manual tuning

If the system must remain online, one tuning method is to first set Ki and Kd values to zero. Increase the Kp until the output of the loop oscillates, then the Kp should be set toapproximately half of that value for a "quarter amplitude decay" type response. Then increase Ki until any offset is correct in sufficient time for the process. However, too much Ki will cause instability. Finally, increase Kd, if required, until the loop is acceptably quick to reach its reference after a load disturbance. However, too much Kd will cause excessive response and overshoot. A fast PID loop tuning usually overshoots slightly to reach the setpoint more quickly; however, some systems cannot accept overshoot, in which case an over-damped closed-loop system is required, which will require a Kp setting significantly less than half that of the Kp setting causing oscillation.

Table 2.2: Effects of increasing a parameter indenpendently

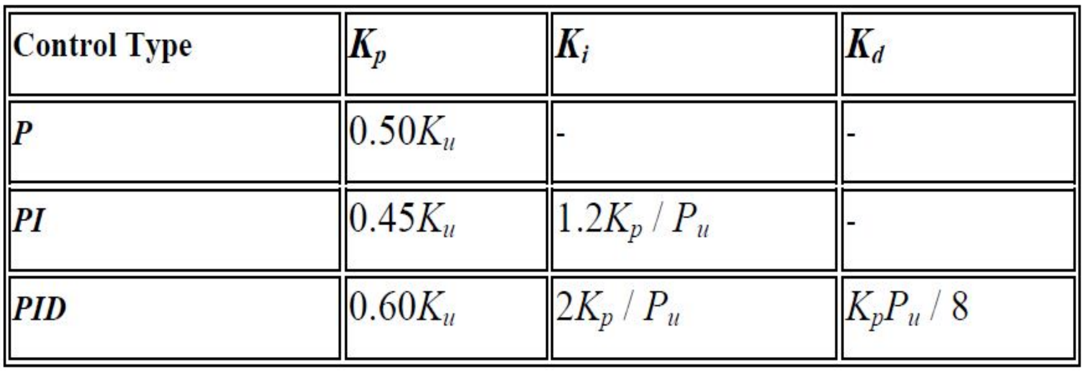


Parameter	Rise time	Overshoot	Settling time	Steady-state error	Stability
K_p	Decrease	Increase	Small change	Decrease	Degrade
K_i	Decrease	Increase	Increase	Decrease significantly	Degrade
K_d	Minor decrease	Minor decrease	Minor decrease	No effect in theory	Improve if K_d small

2.5.5 Ziegler– Nichols method

Another heuristic tuning method is formally known as the Ziegler–Nichols method, introduced by John G. Ziegler and Nathaniel B. Nichols. As in the method above, the Ki and Kd gains are first set to zero. The P gain is increased until it reaches the ultimate gain, Ku, at which the output of the loop starts to oscillate. Ku and the oscillation period Pu are used to set the gains as shown:

Table 2.3: Ziegler–Nichols method



Control Type	K_p	K_i	K_d
P	$0.50K_u$	-	-
PI	$0.45K_u$	$1.2K_p / P_u$	-
PID	$0.60K_u$	$2K_p / P_u$	$K_pP_u / 8$

REFERENCES

[1] Prahlad Kumar Sahoo, Nirmal Kumar Barik, “Speed Control Of Separately Excited Dc Motor Using Self Tuned Fuzzy Pid Controller”.

[2] B.J. Chalmers, “Influence of saturation in brushless permanent magnet drives”, *IEE proc. B*, Electr.Power Appl, vol.139, no.1, 1992.

[3] C.T. Johnson and R.D. Lorenz, “Experimental identification of friction and its compensation in precise, position controlled mechanism”, *IEEE Trans. Ind Applicat*, vol.28, no.6, 1992.

[4] J. Zhang, N. Wang and S. Wang, “A developed method of tuning PID controllers with fuzzy rules for integrating process,” *Proceedings of the American Control Conference*, Boston, 2004, pp. 1109-1114.

[5] K.H. Ang, G. Chong and Y. Li, “PID control system analysis, design and technology,” *IEEE transaction on Control System Technology*, Vol.13, No.4, 2005, pp. 559-576.

[6] H.X.Li and S.K.Tso, "Quantitative design and analysis of Fuzzy Proportional-Integral-Derivative Control- a Step towards Auto tuning", *International journal of system science*, Vol.31, No.5, 2000, pp.545-553.

[7] Tushir Meena, Srivastava Smriti, “Design And Simulation Of A Novel Clustering Based Fuzzy Controller For DC Motor Speed Control”, *Innovative Systems Design And Engineering*, ISSN 2222-1727 (Paper) ISSN 2222-2871 (Online) Vol 2, No 7, 2011.

[8] Thana Pattaradej, Guanrong Chen and PitikhateSooraksa, "Design and Implementation of Fuzzy PID Control of a bicycle robot", *Integrated computer-aided engineering*, Vol.9, No.4, 2002.

[9] Weiming Tang, Guanrong Chen and Rongde Lu, “A Modified Fuzzy PI Controller for a Flexible-joint Robot Arm with Uncertainties”, Fuzzy Set and System, 118 (2001) 109-119.

[10] BomedieneAlloua and ABdessalamAbderrahamani, “Neuro-Fuzzy DC Motor speed Control Using Particle Swarm Optimization,” *Leonaro Electronic Journal of Practices and Technologies ISSN*,1583-1078.

[11] E. E. El-Kholy, A. M. Dabroom, And Adel El-Kholy, “Principle Of Adaptive Fuzzy Logic Controllers For Dc Motor Drives- Part 2.

[12] Rahul Malhotra,Tejbeer Kaur, “Dc Motor Control Using Fuzzy Logic Controller”, *(Ijaest) International Journal Of Advanced Engineering Sciences And Technologies Vol No. 8*, Issue No. 2, 291 – 296.

[13] Varuneet Varun, G. Bhargavi, Suneet Nayak, “Speed Control Of Induction Motor UsingFuzzy Logic Approach”.

[14] Jelena Godjevac, “Comparison between PID and Fuzzy Control”.

[15] Mohd Aizuddin Bin Abu Bakar, “ Development Of A Pid Controller For Dc Motor Using Microsoft Visual Basic”.

[16] Nor Halimahtun Saadiah Binti Mohd, “Speed Control For Direct Curent (Dc) Motor With An Approach Of Fuzzy Logic”.

[17] Jin Zhoa,Bimal K. Bose, “Evaluation Of Membership Function For Fuzzy Logic Controlled Induction Motor”.

[18] Manoj Kumar Pandey, “Analysis of Self Tuning Fuzzy PID Internal Mode Control”.

[19] M.Chow and A. Menozzi, “On The Comparison Of Emerging And Conventional Techniques For DC Motor Control” *proc.IECON*, PP.1008-1013, 1992.

[20] Essam Natsheh, Khalid A. Buragga, “Comparison between Conventional and Fuzzy Logic PID Controllers for Controlling DC Motors”, *IJCSI International Journal of Computer Science Issues*, Vol. 7, Issue 5, September 2010, ISSN(online): 1694-0814.

[21] Haytham M. Fayek, I. Elamvazuthi, “Type-2 Fuzzy Logic PI (T2FLPI) Based DC Servomotor Control”, *Journal of Applied Sciences Research*, 8(5):2564-2574, 2012.

[22] Dr. R. Arulmozhiyal,R. Kandiban, “An Intelligent Speed Controller for Brushless DC Motor”, 978-1-4577-2119-9/12,2012,IEEE.

[23] Malkeet Saini,Neeraj Sharma, “Speed Control of Separately excited D.C Motor Using Computional Method”, Vol 1 Issue 7, September-2012.

[24] Shaosheng Fan, Haizhou Zhu, “Simulation Of The Fuzzy PID Control System For Brushless Dc Motors Based On Matlab”.

[25] Qianhua Xiao, Deqiong Zou, Ping Wei, “Fuzzy Adaptive PID Control Tank Level”, *2010 International Conference on Multimedia Communications.*

[26] Jun Yang, Jinliang Wang, Lili Wang, “Self-Tuning-Parameter Fuzzy PID Control in Fresh Air System”, *Intenational Conference on Intelligent Control and Information Processing.*

[27] Bin Feng, Guofang Gong, huayong Yang, “Self-tuning-parameter fuzzy PID temperature control in a large hydraulic system”, *2009 IEEE/ASME International Conference on Advanced Intelligent Mechatronics.*

[28] Rishabh Abhinav,Satya Sheel, “An Adaptive, Robust control of DC motor Using Fuzzy-PID controller”, *2012 IEEE International Conference on Power Electronics, Drives and Energy System*, December 16-19, 2012, Bengaluru, India.

[29] Nasim Farouk, Tian Bingqi, “Application Of Self Tuning Fuzzy PID Controller On The AVR System”, *Proceedings of 2012 IEEE, International Conference on Mechatronics and Automation,*August 5-8, Chengdu, China.

[30] Haytham M. Fayek, I. Elamvazuthi, “Type-2 Fuzzy Logic PI (T2FLPI) Based Servomotor Control*”, Journal of Applied Sciences Reseacrch, 8(5): 2564-2574*, 2012,ISSN 1819-544X.

[31] Zhao Yongjuan, Pan Yutitan, “The Design and Simulation of FUZZY PID Controller”, *2010 International Forum on Information Technology and Applications.*

[32] Norman S. Nise, “*Control System Engineering* 5th edition”, 2008, John Wiley & Sons, Inc.

[33] Waleed I. Hameed, Khearia A. Mohamad, “Speed Control of Separately Excited DC Motor Using Fuzzy Neural Model Reference Controller”, *International Journal of Instrumentation and Control System (IJICS) Vol.2,* No.4, October 2012.

[34] Tushir Meena, Srivastava Smriti, “Design and Simulation of a Novel Clustering based Fuzzy Controller for Dc Motor Speed Control”, *Innovative System Design and Engineering*, ISSN 2222-1727 (Paper), ISSN 2222-2871 (Online), Vol 2, No 7, 2011.

[35] C. Subba Rami Reddy, M. Surya Kalavathi, “Performance Studies Of Integrated Fuzzy Logic Controller For Brushless Dc Motor Drives Using Advanced Simulation Model”, *Ictact Journal On Soft Computing: Special Issue On Fuzzy In Industrial And Process Automation*, July 2011, Volume: 02, Issue 01.

[36] Marcelo Godoy Simoes, “Introduction to Fuzzy Logic”, Colorado School of Mines,Engineering,*http://inside.mines.edu/~msimoes/documents/Intro_Fuzzy_Logic.pdf*, [Accessed on Jun 2013].

[37] Fayez, “Chapter (1) : Mathematical Modeling of DC Machines”, *http://faculty.ksu.edu.sa/fayez/Fayez%20Documents/Project%201.pdf,* [Accessed on Jun 2013].