INTERNATIONAL JOURNAL OF ENHANCED RESEARCH IN SCIENCE TECHNOLOGY & ENGINEERING
VOL. 2 ISSUE 3, MARCH-2013 ISSN NO: 2319-7463

Approaches for tuning of PID Controller

Sami Jan Lolu[1], Sukhdeep Singh Bhatti[2], Prof. R. K. Sharma[3]
[1,2]M. Tech Scholars (Electrical Engineering), Lovely Professional University, Punjab, India
[3]Head of Department, Electrical Engineering, Lovely Professional University, Punjab, India
[1]sammymanzoor@gmail.com, [2]sukhdeep.singh87@yahoo.com, [3]rk.sharma@lpu.co.in

Abstract: Controlling the process is the main issue that rises in the process industry. It is very important to keep the process working probably and safely in the industry, for environmental issues and for the quality of the product being processed. PID control is a control strategy that has been successfully used over many years. Simplicity, robustness, a wide range of applicability and near optimal performance are some of the reasons that have made PID control so popular in the academic and industry sectors. Recently, it has been noticed that PID controllers are often poorly tuned and some efforts have been made to systematically resolve this matter. In the paper a brief summary of PID theory is given, then, some of the most used PID tuning methods are discussed.

Keywords: Proportional (P), Integral (I), Derivative (D), controller, control system.

INTRODUCTION

The theory of control deals with the methods, which leads to the change of behavior of controlled dynamic system. The desired output of a system is called the reference or set point. When one or more outputs of the system need to follow a certain reference over time then a controller modifies the inputs of system to obtain the desired value on the output of the system as shown in Fig. 1.

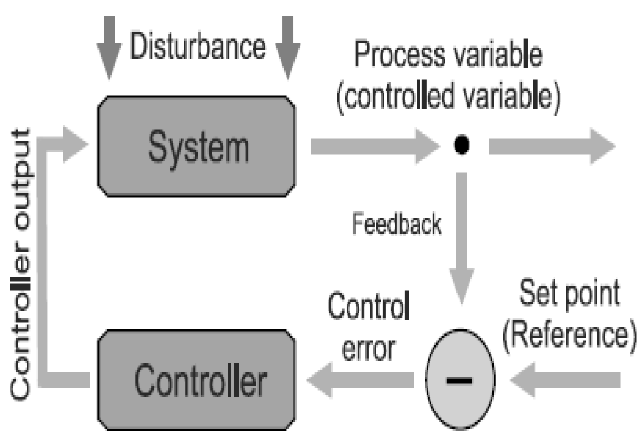

Fig. 1: The general concept of the negative feedback loop to control the dynamic behavior of the system

The PID controller has three separate constant parameters: Proportional (P), Integral (I) and Derivative (D). It can be said the P depends on present error, I on accumulation of past errors and D is prediction of future errors based on rate of change. The PID controller calculates an error value as the difference between a measured process variable and a desired set point. The controller attempts to minimize the control error by adjusting the process controller outputs. After corrective action from the controller, the system should reach point of stability. As stability means the set point is being held on the output without oscillating around it.

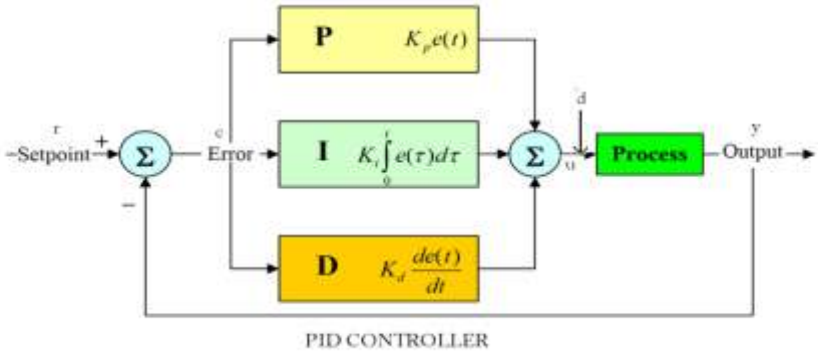

Fig. 2: The block diagram of the PID controller

Basic block diagram of standard PID controller is based on parallel circuit, Fig. 2. The proportional, integral, and derivative terms are summed to calculate the output of the PID controller. Defining *u*(*t*) as the controller output, the general ideal form of the PID algorithm is:

$$u(t) = Kp\ e(t) + \frac{1}{Ti}\int_0^t e(\tau)dt + T_d \frac{de(t)}{dt}$$

where, K_p is the single gain, Ki = Kp / Ti, Kd = Kp.T_d, Ti is the integral time constant and T_d is the derivative time constant. The variants of PID controller given in standard form by equation as shown:

$$u(t) = Kp\,[e(t) + Ki \int_0^t e(\tau)dt + Kd \frac{de(t)}{dt}]$$

Here, the parameters have a clear physical meaning. In particular, the inner summation produces a new single error value which is compensated for future and past errors. The addition of the proportional and derivative components effectively predicts the error value at *Td* seconds (or samples) in the future, assuming that the loop control remains unchanged. The integral component adjusts the error value to compensate for the sum of all past errors, with the intention of completely eliminating them in *TI seconds* (or samples). The resulting compensated single error value is scaled by the single gain *Kp* [1].

Using Laplace's transformation the transfer function of PID controller looks like [2]:

$$Gc(s) = P + I + D = Kp + \frac{Ki}{s} + Kd\ s$$

$$Gc(s) = Kp\ (1 + \frac{1}{Ti\ s} + Td\ s)$$

In practice, the following realisation is usually employed:

$$Gc(s) = Kp + \frac{Ki}{s} + \frac{Kd\ s}{1 + Tn\ s}$$

The effect of each parameter on the step response of the system is illustrated in below table [3]:

Parameter	Rising time	Overshoot	Settling time	Steady state error
Kp	decrease	increase	Small change	decrease
Ki	decrease	increase	increase	eliminate
Kd	Small change	decrease	decrease	Small change

VOL. 2 ISSUE 3, MARCH-2013 ISSN NO: 2319-7463

The remainder of this paper describes different PID parameter tuning methods together with a discussion on some of their advantages and disadvantages:

A. **The Ziegler-Nichols step response method**

The Ziegler-Nichols step response method is an experimental tuning method for open-loop plants. The first step in this method is to calculate two parameters A and L that characterize the plant. These two parameters (A, L) can be determined graphically from a measurement of the step response of the plant as illustrated in Figure 3. First, the point on the step response curve with the maximum slope is determined and the tangent is drawn. The intersection of the tangent with the vertical axis gives A, while the intersection of the tangent with the horizontal axis gives L.

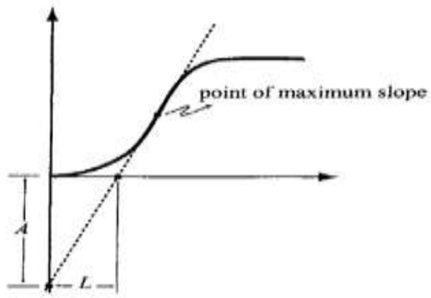

Fig.3: Graphical determination of parameters A and L.

Once A and L are determined, the PID controller parameters are then given in terms of A and L by the following formulas:

$$Kp = \frac{1.2}{A}$$

$$Ki = \frac{0.6}{AL}$$

$$Kd = \frac{0.6L}{A}$$

When using the previous formulas for Kp, Ki, and Kd, the amplitude decay ratio is 0.25, which means that the first overshoot decays to 1/4th of its original value after one oscillation. It has been verified by several experimental results that this method gives a small settling time [3].

B. The Ziegler-Nichols frequency response method

The Ziegler-Nichols frequency-response method is a closed-loop tuning method. In this method, the two parameters to be calculated are the ultimate gain Ku and the ultimate period Tu which can be calculated experimentally in the following way:

Set the integral and differential gains to zero and hence the controller become in the proportional mode only. Close loop system is shown in fig.4. The proportional gain Kp is then increased slowly until a periodic oscillation in the output is observed. This critical value of Kp is called the ultimate gain Ku.The resulting period of oscillation is referred to as the ultimate period Tu. Based on Ku and Tu, the Ziegler-Nichols frequency response method gives the following simple formulas for setting PID controller parameters according to table (shown in fig.5)[3] :

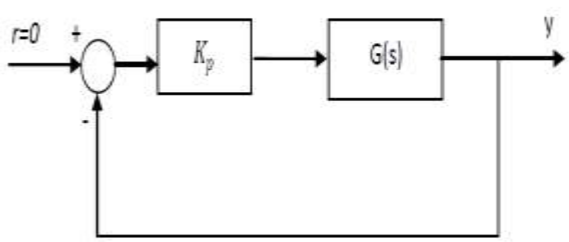

Fig .4: The closed-loop system with the proportional

Type of controller	Kp	Ti	Td
P	0.5Ku	-	-
PI	0.45Ku	0.833Tu	-
PID	0.6Ku	0.5Tu	0.125Tu

Fig 5: PID controller parameters

C. Kappa-Tau Tuning

The dynamics of a system can be described more accurately if three parameters are used in the design instead of two. The kappa-tau tuning method is used in automatic tuning. As in the ZN method it comes in two versions. One is based on the step response, in which the process is characterised by a static gain *Kp,* a gain *a* (the gain of the transient part of the open loop response), and a dead time *L*. The controller parameters are a function of the normalized dead time τ given by:

$$\tau = \frac{L}{L+T}$$

with *T* being the dominant time constant of the process. The second method is based on the frequency response; the process is characterized by a static gain Kp, an ultimate gain Ku and an ultimate period Tu. Here, the controller parameters are a function of the gain ratio *k,* where Maximum sensitivity is used as the design objective in both cases.[2]

$$k = 1/(k_p.\text{ku})$$

D. Genetic Algorithms for PID Tuning

Genetic algorithms are a rapidly expanding area in control systems design. A genetic tuning algorithm usually starts with no knowledge of the correct solution and depends on the responses from its environment to give an acceptable result. It has been shown that genetic algorithms are capable of locating optimal regions in complex domains avoiding the difficulties, or even erroneous results in some cases, associated with the gradient descent methods and with high-order systems. To obtain the PID tuning parameters one usually has to minimise a performance index. This, in the majority of the cases, is one of the following:

$$\text{ISE} = \int_0^T |r(t) - y(t)|2\text{dt}$$

$$\text{IAE} = \int_0^T |r(t) - y(t)|dt$$

$$\text{ITAE} = \int_0^T t|r(t) - y(t)|dt$$

with *r(t)* being the reference input and *y(t)* the output of the system. In [4], a genetic algorithm based on Gray coding is used. Each PID parameter *(Kp,* K,, *Kd)* is represented by 16 bits and a single individual is generated by concatenating the coded parameter strings. The genetic algorithm requires a population of initial approximations, which may be random, to start the search. The algorithm then checks the fitness of each individual (or chromosome), and then grades them. A selection process follows where five of the fittest individuals are chosen. The remaining individuals are selected probabilistically. The selected individuals are used to produce the next population, and the process is then repeated until the design requirements are met. This method is applicable to a wide range of system models due to its adaptability. High-order systems do not present a problem with this tuning procedure [2].

E. **Modulus Optimum:**

Modulus Optimum (MO) method is based on the transfer function of set point Gref(s), where this transfer function is ratio of Laplace s-domain of process output variable to set point input variables. In ideal case the transfer function would be Gref(s) = 1, i.e. step response of process variable is equal to set point. In frequency domain it corresponds with following condition.

$$Aref(\omega) = 1$$

$$|Gref(j\omega)| = Aref(\omega) = 1$$

This condition cannot be satisfied in reality, however it can be proven that control process ends the fastest when amplitude characteristics Aref (ω)will be flat at first and then it will monotonically decreasing as we can see in below figure 6

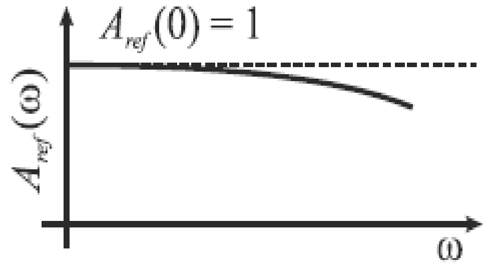

Fig 6: the block diagram of PID controller.

The setting of PID parameters Kp, Ti and Td by MO method is sorted in the table for practical use and it depends on the type of controlled plant as given below in fig.7.[1]:

Model of controlled plant.	Kp	Ti	Td
$\frac{k}{(T1s+1)\vert T2s+1)(T3s+1)}$ T1$\geq T2 \geq T3$	$\frac{T1}{2kT3}$	T1+T2	$\frac{T1T2}{T1+T2}$

Fig.7. calculation of PID controller's parameters by MO method

F. AMIGO tuning rules

AMIGO tuning rule consider a controller described by:

u(t) =kp [byr(t) – yf (t)] + ki $\int_0^t yr(\alpha) - yf(\alpha)d\alpha$ +kd$\left(c\frac{dyr(t)}{dt} - \frac{yf(t)}{dt}\right)$

Where u is the control variable, ysp the set point, y the process output, and yf is the filtered process variable, i.e. Yf(s) = Gf(s)Y(s) The transfer function Gf(s) is a first order filter with time constant Tf, or a second order filter if high frequency roll-off is desired .

G(s) =$\frac{1}{(1+sTt)^\wedge 2}$

Parameters b and c are called set-point weights. They have no influence on the response to disturbances but they have a significant influence on the response to set point changes. Neglecting the filter of the process output the feedback part of the controller has the transfer function

C(s) = K[1+$\frac{1}{\text{sTi}}$ +sTd]

The advantage by feeding the filtered process variable into the controller is that the filter dynamics can be combined with in the process dynamics and the controller can be designed designing an ideal controller for the process P(s) Gf(s). The objective of AMIGO was to develop tuning rules for the PID controller in varying time-delay systems by analyzing different properties (performance, robustness etc.) of a process test batch. The AMIGO tuning rules are based on the KLT-process model obtained with a step response experiment. The AMIGO tuning rules are:

$$Kc = (0.2 + 0.45 * \frac{T}{L})$$

$$Ti = \left(\frac{0.4L+0.8T}{L+0.1T}\right) L$$

$$Td = \frac{0.5LT}{0.3L+T}$$

In order to use the PID controller with filtering, the rules are extended as follows:

$$\begin{cases} kc = Kc \\ ki = \dfrac{Kc}{Ti} \\ Kd = Kc * Td \end{cases} \quad \begin{cases} b = \begin{cases} 0 \; if \; \tau \le 0.5 \\ 1 \; if \; \tau > 0.5 \end{cases} \\ c = 0 \\ Tf = \begin{cases} \dfrac{.05}{\omega gc} \; if \; \tau \le 0.2 \\ 0.1 * L \; if \; \tau > 0.2 \end{cases} \end{cases}$$

Where: ωgc is the gain crossover frequency and $\tau = \frac{L}{L+T}$ is the relative dead-time of the process, which has turned out to be an important process parameter for controller tuning [5].

G. Jitter margin

The jitter margin is an upper bound for additional delay that can be added to a closed-loop control system while maintaining stability. The delay can be of any type (constant, time-dependent, random), but the jitter margin determines the upper bound for the delay. The formal definition of the jitter margin is given in [7], where three different controller/ plant–uncertainty combinations are investigated. The first one is shown in Fig. 1, left, where a continuous-time plant and a continuous-time controller with controller output uncertainty are shown. This continuous-time SISO system is stable for any time-varying delays defined by:

$$\Delta(v) = v\big(t - \delta(t)\big), \quad 0 \le \delta(t) \le \delta max$$

If,

$$\left|\frac{P(j\omega)C(j\omega)}{1+P(j\omega)C(j\omega)}\right| < \frac{1}{\delta max\omega}, \ \forall\omega \in [0,\infty]$$

δmax is the jitter margin. The proof of the result is based on presenting the uncertainty (varying delay) with an operator, $\Delta f \coloneqq (\Delta - 1).\frac{1}{s}$ (s being the Laplace operator) and on the small gain theorem. However, in this paper the jitter is assumed to be after the plant (e.g. sampling jitter) as depicted in Fig. 8, right. Since the signals in the control loop are all continuous, and only the plant and controller switch their positions, the small gain theorem-based stability proof still holds for the control system of Fig. 8, right [06].

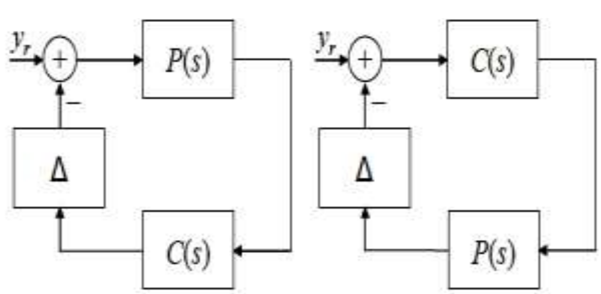

Fig. 8 Sampling Jitter

CONCLUSION

This paper has presented an overview of PID control, its advantages, disadvantages and different tuning methods. Only a flavor of the available PID tuning methods has methods here. However, it must also be pointed out that PID control may not be sufficient for some cases, for example, processes with more than one oscillatory mode or processes with large time delays or with complex disturbance behavior. It is concluded here that PID control is still of great interest, and is a promising control strategy that deserves further research and investigation. Both industry and academia have a lot to gain from this.

REFERENCES

[1]. R. Matousek, P. Minar, S. Lang, M. Seda,"HC12: Efficient PID Controller Design", Advance online publication, 27 February 2012.
[2]. P. Cominos, N.Munro, "PID controllers: recent tuning methods and design to specification", IEE Proc.-Control Theory Appl., Vol. 149, No. 1, January 2002.
[3]. Wesam H. Sakallah, "Comparative Study for Controller Designof Time-delay Systems", 2009.
[4]. Mahony T., Downing C. J., Fatla K., "Genetic algorithms for PID parameter optimization: minimizing error criteria", Cork Institute of Technology, Cork, Ireland, pp. 148-153.
[5]. Ala Eldin Abdallah Awouda, Rosbi Bin Mamat, "New PID Tuning Rule Using ITAE Criteria", International Journal of Engineering (IJE), Volume (3), Issue (6), pp. 597-608.
[6]. Lasse M. Eriksson, Mikael Johansson, "PID Controller Tuning Rules for Varying Time-Delay Systems", Proceedings of the 2007 American Control Conference, July 11-13, 2007.
[7]. C. Y. Kao, B. Lincoln, "Simple stability criteria for systems with time-varying delays", Automatica, vol. 40, pp. 1429-1434, Aug. 2004.