Performance Analysis Of Speed Control Of Dc Motor Using P, PI, PD And PID Controllers

V M Venkateswara Rao
Assistant professor
Dnyanganga College of Engineering and Research, Pune

Abstract

In this paper deals the usage of continuous time P-I-D controllers. In the first part of the recitation, it was aimed to show the how P, P-I, P-I-D controllers change the steady state response of the closed loop systems. Moreover, the methods to tune P-I-D controllers were introduced. It was meant to show that how hard it could get to properly tune a P-I-D controller. Secondly, it was intended to show how P, P-D, P-I, and P-I-D controllers affect the transient response of the closed loop system. It was designed to show how one can gain a feature but lose the other. Thirdly, it was intended to show how one should estimate the dynamics of the continuous time plant and use proper sampling time for discrete time P-I-D controller. It was also show how changing transformation method may cause different pole locations on the z-plane.

1. Introduction

It was mainly about P, P-D, P-I and P-I-D controllers, their digital versus continuous time realizations and their characteristics including sampling period effects on the response of digital ones. Apart from these topics, P-I-D tuning methods such as manual tuning, Ziegler-Nichols tuning and MATLAB tuning method were discussed. Transient performances of P, P-D, P-I and P-I-D controllers were explained in detail. Modelling a discrete time P-I-D controller to control a continuous time plant was explained over a MATLAB code introducing the effect of sampling time and the choice of s*-domain to z-domain transformation method on MATLAB. It was explained how to remove poles that cause instability in discrete time by adding a new pole. Finally, it was shown how one could control the speed **and position of the vehicle** using discrete time P-I-D controller **on the ‘Gate’ project.**

P Controller:

P controller is mostly used in first order processes with single energy storage to stabilize the unstable process. The main usage of the P controller is to decrease the steady state error of the system. As the proportional gain factor K increases, the steady state error of the system decreases. However, despite the reduction, P control can never manage to eliminate the steady state error of the system. As we increase the proportional gain, it provides smaller amplitude and phase margin, faster dynamics satisfying wider frequency band and larger sensitivity to the noise. We can use this controller only when our system is tolerable to a constant steady state error. In addition, it can be easily concluded that applying P controller decreases the rise time and after a certain value of reduction on the steady state error, increasing K only leads to overshoot of the system response. P control also causes oscillation if sufficiently aggressive in the presence of lags and/or dead time. The more lags (higher order), the more problem it leads. Plus, it directly amplifies process noise

P-I Controller:

P-I controller is mainly used to eliminate the steady state error resulting from P controller. However, in terms of the speed of the response and overall stability of the system, it has a negative impact. This controller is mostly used in areas where speed of the system is not an issue. Since P-I controller has no ability to predict the future errors of the system it cannot decrease the rise time and eliminate the oscillations. If applied, any amount of I guarantees set point overshoot

P-D Controller:

The aim of using P-D controller is to increase the stability of the system by improving control since it has an ability to predict the future error of the system response. In order to avoid effects of the sudden change in the value of the error signal, the derivative is taken from the output response of the system variable instead of the error signal. Therefore, D mode is designed to be proportional to the change of the output variable to prevent the sudden changes occurring in the control output resulting from sudden changes in the error signal. In addition D directly amplifies process noise therefore D-only control is not used.

P-I-D Controller:

P-I-D controller has the optimum control dynamics including zero steady state error, fast response (short rise time), no oscillations and higher stability. The necessity of using a derivative gain component in addition to the PI controller is to eliminate the overshoot and the oscillations occurring in the output response of the system. One of the main advantages of the P-I-D controller is that it can be used with higher order processes including more than single energy storage.

In order to observe the basic impacts, described above, of the proportional, integrative and derivative gain to the system response, see the simulations below prepared on MATLAB in continuous time with a transfer function $1/s^2+10s+20$ and unit step input. The results will lead to tuning methods

2.Simulations and Results to Find the Constraints on Loop Tuning:

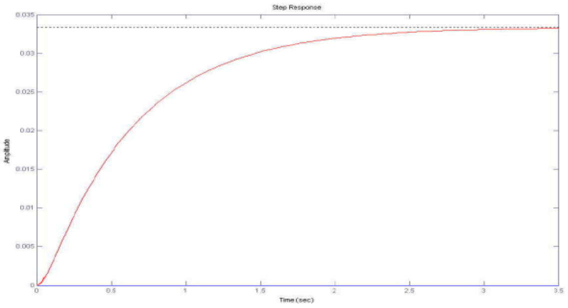


Figure 1: Step response without any controller
Steady state error (e_{ss}) =0.965(too high), Rise time(t_r) =3 sec.

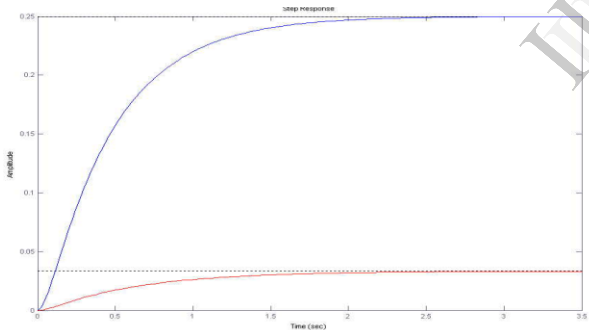


Figure 2: Step response with P controller, Kp = 10, K_i = 0, K_d = 0
Out put response improved, Rise time (t_r) =decreased

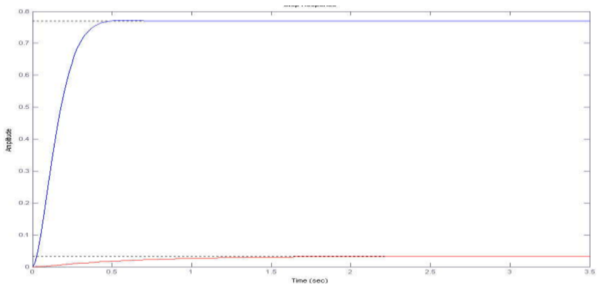


Figure 3: Step response with P controller, Kp = 100, K_i = 0, K_d = 0

Steady state error (e_{ss}) =0.23(decreased), Rise time (t_r) =decreased.

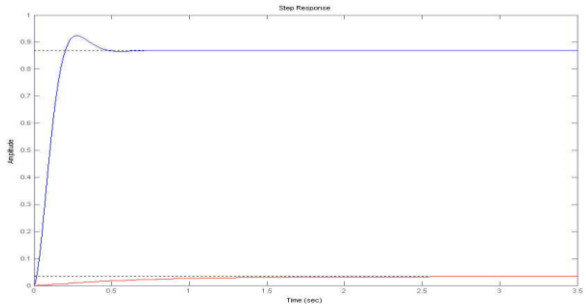


Figure 4: Step response with P controller, Kp = 200, Ki = 0, K_d = 0, Steady state error (e_{ss}) =0.23, Rise time (t_r) =0.5 sec, overshoot occurs output response.

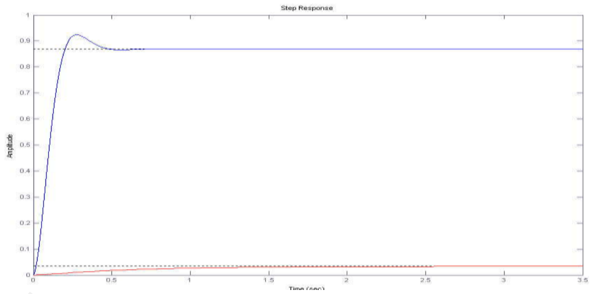


Figure 5: Step response with P-I controller, Kp = 200, K_i = 100, K_d = 0, Steady state error (e_{ss}) =0, Rise time (t_r) =0.3 sec, overshoot remains

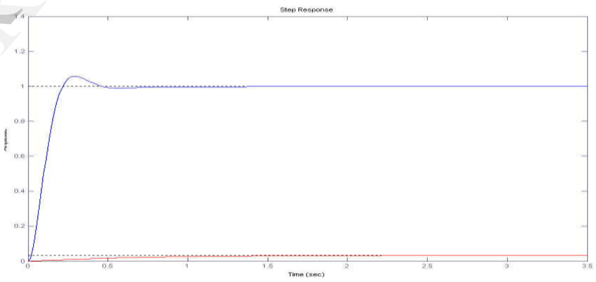


Figure 6: Step response with P-I controller, Kp = 200, Ki = 200, Kd = 0, Steady state error (e_{ss}) =0, Rise time (t_r) =0.3 sec, t_s=0.7sec, overshoot remains

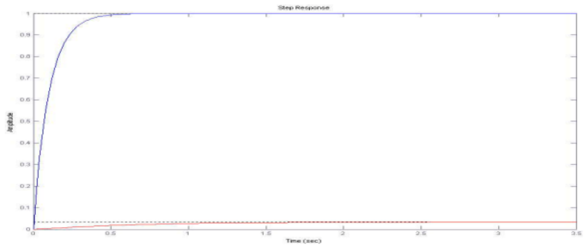


Figure 7: Step response with P-I-D controller, Kp = 200, Ki = 200 and Kd = 10, Steady state error (e_{ss}) =0, Rise time (t_r) =0.3 sec, t_s=0.5sec, overshoot removes

3.Loop Tuning:

Tuning a control loop is arranging the control parameters to their optimum values in order to obtain desired control response. At this point, stability is the main necessity, but beyond that, different systems leads to different behaviors and requirements and these might not be compatible with each other. In principle, P-I-D tuning seems completely easy, consisting of only 3 parameters, however, in practice; it is a difficult problem because the complex criteria at the P-I-D limit should be satisfied. P-I-D tuning is mostly a heuristic concept but existence of many objectives to be met such as short transient, high stability makes this process harder. For example sometimes, systems might have nonlinearity problem which means that while the parameters works properly for full load conditions, they might not work as effective for no load conditions. Also, if the P-I-D parameters are chosen wrong, control process input might be unstable, with or without oscillation; output diverges until it reaches to saturation or mechanical breakage.

For a system to operate properly, the output should be stable, and the process should not oscillate in any condition of set point or disturbance. However, for some cases bounded oscillation condition as a marginal stability can be accepted.

As an optimum behaviour, a process should satisfy the regulation and command breaking requirements. These two properties define how accurately a controlled variable reaches the desired values. The most important characteristics for command breaking are rise time and settling time. For some systems where overshoot is not acceptable, to achieve the optimum behaviour requires eliminating the overshoot completely and minimizing the dissipated power in order to reach a new set point.

In today's control engineering world, P-I-D is used over % 95 of the control loops. Actually if there is control, there is P-I-D, in analogy or digital forms. In order to achieve optimum solutions Kp, Ki and Kd gains are arranged according to the system characteristics. There are many tuning methods, but most common methods are as follows:

4.Manual Tuning Method

Ziegler-Nichols Tuning Method
PID Tuning Software Methods (ex. MATLAB)

Manual Tuning Method:

Manual tuning is achieved by arranging the parameters according to the system response. Until the desired system response is obtained Ki, Kp and Kd are changed by observing system behavior.

Example (for no system oscillation): First lower the derivative and integral value to 0 and raise the proportional value 100. Then increase the integral value to 100 and slowly lower the integral value and observe the system's response. Since the system will be maintained around set point, change set point and verify if system corrects in an acceptable amount of time. If not acceptable or for a quick response, continue lowering the integral value. If the system begins to oscillate again, record the integral value and raise value to 100. After raising the integral value to 100, return to the proportional value and raise this value until oscillation ceases. Finally, lower the proportional value back to 100.0 and then lower the integral value slowly to a value that is 10% to 20% higher than the recorded value when oscillation started (recorded value times 1.1 or 1.2).

Although manual tuning method seems simple it requires a lot of time and experience

Ziegler-Nichols Method:

More than six decades ago, P-I controllers were more widely used than P-I-D controllers. Despite the fact that P-I-D controller is faster and has no oscillation, it tends to be unstable in the condition of even small changes in the input set point or any disturbances to the process than P-I controllers. Ziegler-Nichols Method is one of the most effective methods that increase the usage of P-I-D controllers

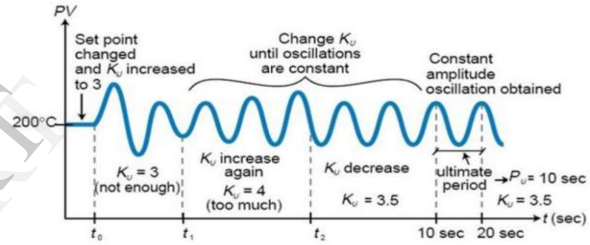
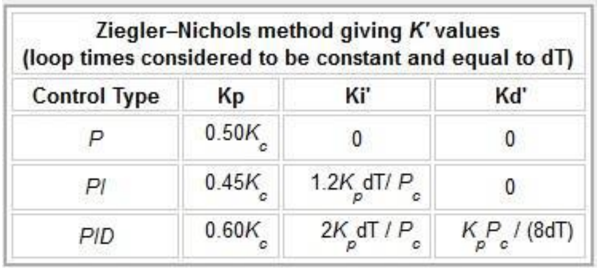


Figure 8: Ziegler-Nichols P-I-D controller tuning method

The logic comes from the neutral heuristic principle. Firstly, it is checked that whether the desired proportional control gain is positive or negative. For this, step input is manually increased a little, if the steady state output increases as well it is positive, otherwise; it is negative. Then, Ki and Kd are set to zero and only Kp value is increased until it creates a periodic oscillation at the output response. This critical Kp value is attained to be "ultimate gain", Kc and the period where the oscillation occurs is named as Pc "ultimate period". As a result, the whole process depends on two variables and the other control parameters are calculated according to the table in the Figure 9.



Ziegler–Nichols method giving *K'* values
(loop times considered to be constant and equal to dT)

Control Type	Kp	Ki'	Kd'
P	$0.50K_c$	0	0
PI	$0.45K_c$	$1.2K_p dT/ P_c$	0
PID	$0.60K_c$	$2K_p dT / P_c$	$K_p P_c / (8dT)$

Figure 9: Ziegler-Nichols P-I-D controller tuning method, adjusting Kp, Ki and Kd

Advantages:

It is an easy experiment; only need to change the P controller, Includes dynamics of whole process, which gives a more accurate picture of how the system is behaving

Disadvantages:

Experiment can be time consuming, it can venture into unstable regions while testing the P controller, which could cause the system to become out of control, For some cases it might result in aggressive gain and overshoot

5.PID Controller Design for Controlling DC Motor Speed

To design P-I-D controller is to make the actual motor speed match the desired motor speed. P-I-D algorithm will calculate necessary power changes to get the actual speed. This creates a cycle where the motor' speed is constantly being checked against the desired speed. The power level is always set based on what is needed to achieve the correct results.

By using P-I-D controller, we can make the steady state error zero with integral control. We can also obtain fast response time by changing the P-I-D parameters. P-I-D is also very feasible when it is compared with other controllers.

In our project, first of all we have obtained the P-I-D parameters for our system. Then we have constituted our own P-I-D controller.

6.The Block Diagram of the DC Motor Speed Control Loop:

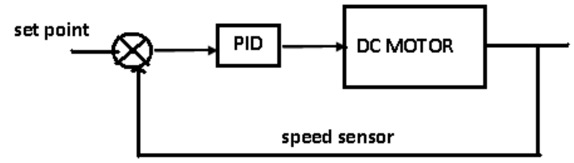


Figure 10: The Block Diagram of the DC Motor Speed Control Loop

As it is seen from the block diagram of the DC motor control loop, the speed sensor (encoder) measure the speed of the DC motor. In these loops we have the actual speed of the DC motor with the desired one. The DC speed measurement gives the actual speed value. The error between theoretical and practical values is corrected with PID controller. The parameters of the PID controller are determined with MATLAB results which will be explained in the following sections. The output of the PID controller gives the duty cycle of the square wave generator.

PID Parameters:

PID controller can be investigated under 3 main categories. Each controller has different properties in terms of controlling the whole system.

In proportional control, adjustments are based on the current difference between the actual and desired speed.

In integral control, adjustments are based on recent errors.

In derivative control, adjustments are based on the rate of change of errors.

7.The Design Requirements of the System:

The design requirements of the systems may vary from one system to another. For our case, we want a fast response of the system to an error. The overshoot of the system should not be higher than %5 and the settling time should be smaller than 2 seconds.

The main design requirements are as follows;
Settling time should be less than 2 seconds;
Overshoot of the system should be less than 5%;
Steady state error should be less than 1%

The Schematic of the DC Motor:

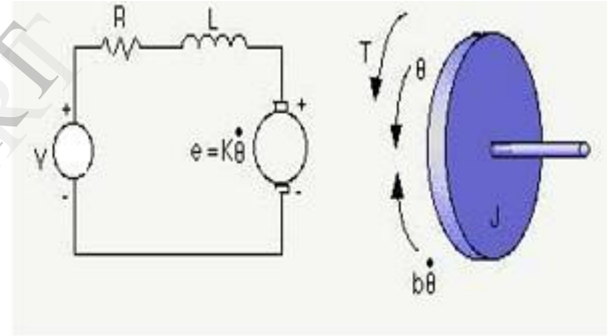


Figure 11-The Schematic of the DC Motor

The Parameters of the Dc Motor:

The parameters of the DC motors may change according to different torque and rpm values of the DC motors. For 1000 rpm DC motor that we have used in this discussions

- Rotor moment of inertia (Jm)=0.01kg*m^2/s^2
- Resistance=1Ω
- Inductor=0.5H
- Electromotive Force Constant Kt=0.01Nm/Amp
- Motor Viscous Friction Constant (Beq) =0.1Nms

The open loop transfer functions of the DC motor:

The transfer function of the DC motor can be found from the schematic of the DC motor in Figure 11.

$$s(Js+b)Q(s)=KI(s)$$

$$(Ls+R)I(s)=V(s)-KQ(s)$$

$$\frac{Q(s)}{V(s)}=\frac{K}{(Js+b)(Ls+R)+K^2}$$

From that point, we have to find the PID parameters for our PID control algorithm. To find the parameters of PID, we should start from proportional constant.

By using only proportional controller, the block diagram of the overall system would be as follows

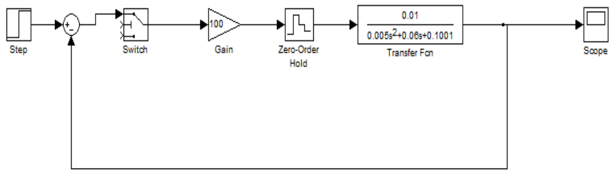


Figure 12-The Block Diagram of the System with Proportional Controller

The MATLAB Result for Kp=100:

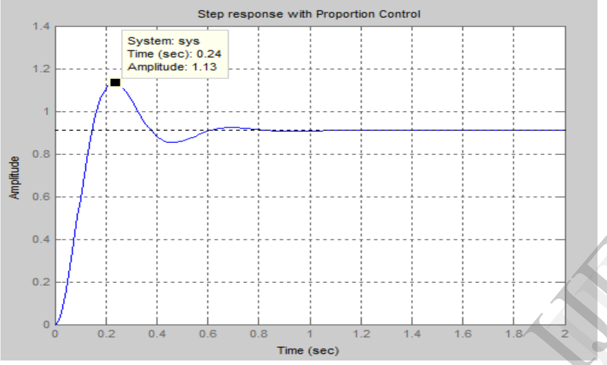


Figure 13-The MATLAB Result, Kp=100

The overshoot of the system with Kp = 100 is %25 which does not satisfy our design requirements. The settling time of the system is about 0.37 seconds. This satisfies our system requirement. The steady state error of the system is 0.1.

After adding derivative and integral controllers to the system; the block diagram of the system is the following;

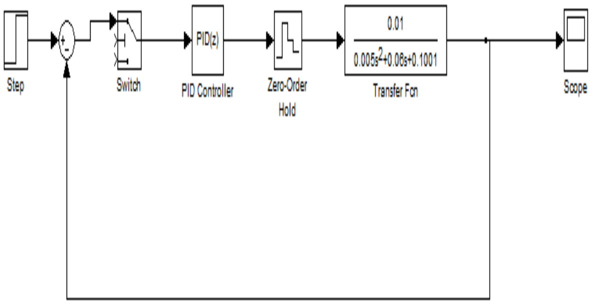


Figure 56-The Block Diagram of the Overall System after Adding Integral and Derivative Controllers

Initially we have chosen both of our integral and derivative controllers' parameters as 1;

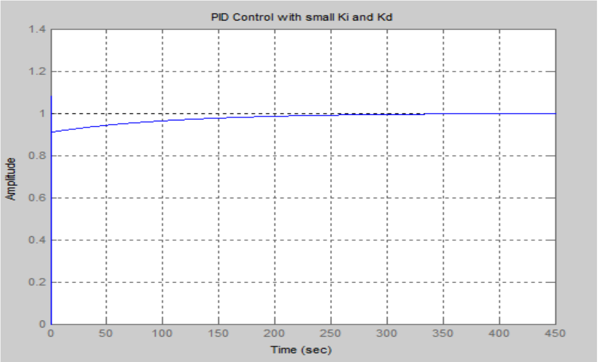


Figure 14-The MATLAB Result for Ki=1, Kd=1, Kp=100

The settling time of the new system is 400 seconds which is far away from satisfying our design requirement. There is also a pulse in t=0 which causes instability to our system. To obtain a better response, we have increased the value of Ki to 200;

The MATLAB Result For Ki=200; Kd=1; Kp=100:

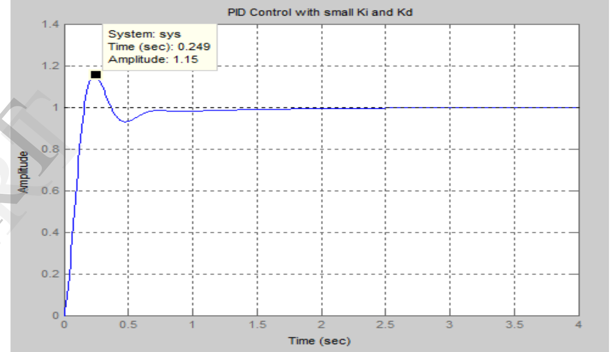


Figure 15- The MATLAB Result for Ki=200, Kd=1, Kp=100

As we have increased the value of Ki, the steady state value of the system becomes 0. Actually the aim in using the integral control is to make the steady state error zero. For The overshoot of the system does not satisfy the design requirement. For that reason let increase the value of K_d.

The MATLAB Result For Ki=200; K_d=10; Kp=100:

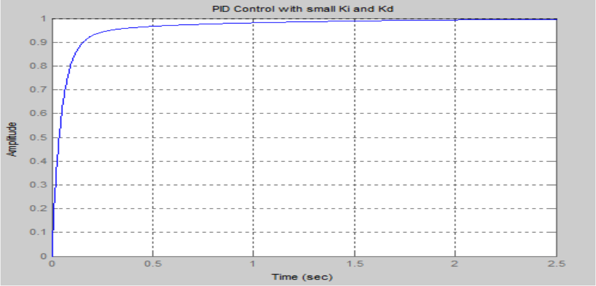


Figure 16- The MATLAB Result for Ki=200, Kd=10, Kp=100

As we have increased the value of Kp, the overshoot value of the system becomes 0, $e_{ss}=0$, $t_s=2$ sec and.

With those parameters of P-I-D controller, we have obtained the system design requirements.

Note that P-I-D parameters are found in continuous time system. So we have to check whether these parameters satisfy the system requirements in discrete time domain.

To be able to check it, first of all we have to obtain the DC motor transfer function in z domain. For the conversion from s to z, we have used ZOH method which is learnt in the class.

8.s*-domain to z-domain with ZOH (only plant-DC motor):

$$T(s) = \frac{2}{(s+9.997)(s+2.003)}$$

$$T(z) = \frac{0.0020586\,(z+0.8189)}{(Z-0.9047)(z-0.6066)}$$

Sampling time: 0.05, Note that sampling time of the system is defined according to dominant pole approximation.

Now let investigate the step response of the plant with zero order hold;

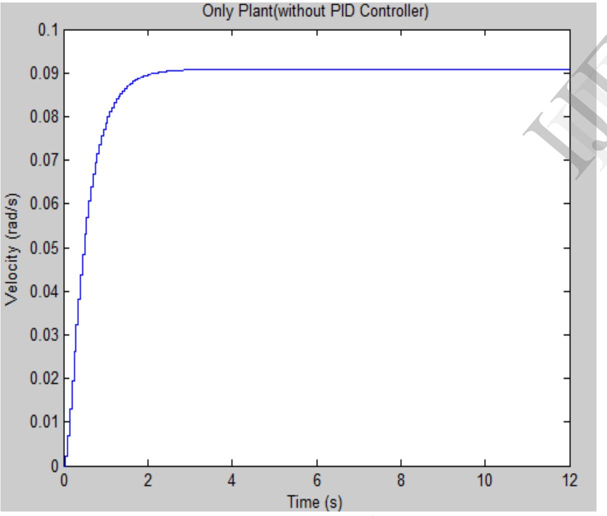


Figure 17-The Step Motor Response of the DC motor without PID controller

The steady state error of the system is increased to 0.9 which was 0.1 in continuous time. From that graph we can make the assumption that our system is required modification

Let investigate the step response of the compensated system with P-I-D;

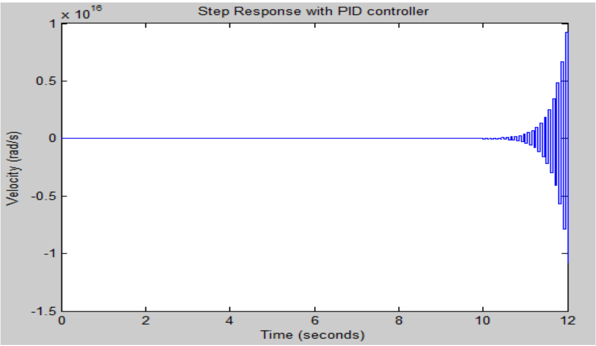


Figure 18-The Step Response of Plant with PID Controller

Our system's step response is unstable. To find the reason of instability, we have to check the root locus of the compensated system.

The root locus of the system is the following;

Root Locus of the Compensated System:

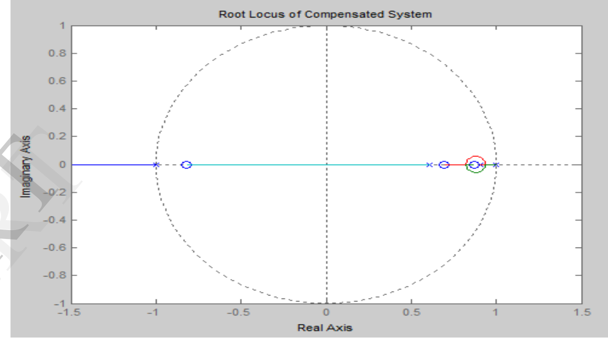


Figure 19-The Root Locus of the Compensated System

Note that the pole at -1 goes to infinity as the gain (K) of the system is increased. It is the reason of instability. To be able to make the system stable, let make a pole at -0.82. After adding a pole at -0.82 the root locus of the system is the following;

After Adding a Pole at -0.82:

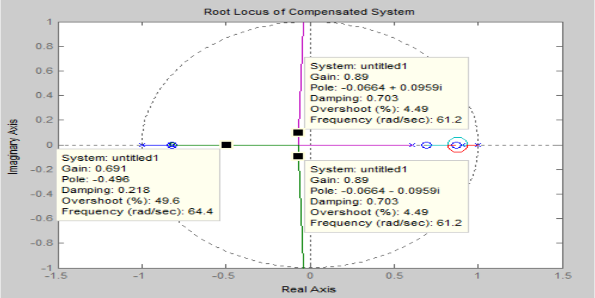


Figure 20-The Root Locus of the Compensated System

Note that for the values of the poles in the unit circle, we expect to obtain stable compensated system. For that purpose, to show the gain and other specifications of the system, we have taken 3 point. Note that any point in the unit circle can be chosen to obtain stable systems. At that point we have chosen gain=0.89

The Step Response of the System with P-I-D Controller:

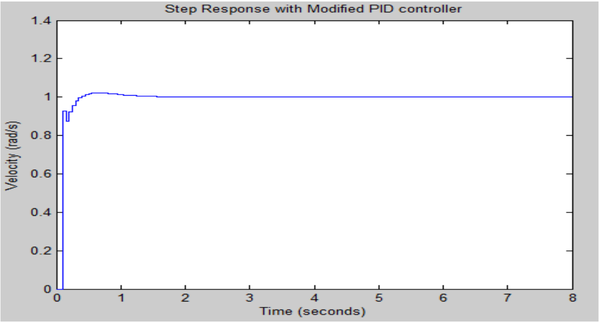


Figure 21-The Step Response of the System with modified P-I-D Controller

As it is seen from Figure 21, the system design requirements are also satisfied in discrete time model. In real life, the addition of pole can be done by adding a capacitor at the end of the PID controller.

9.Conclusions:

P-I-D control and its variations are commonly used in the industry. They have so many applications. Control engineers usually prefer P-I controllers to control first order plants. On the other hand, P-I-D control is vastly used to control two or higher order plants. In almost all cases fast transient response and zero steady state error is desired for a closed loop system. Usually, these two specifications conflict with each other which makes the design harder. The reason why P-I-D is preferred is that it provides both of these features at the same time. In this recitation, it was aimed to explain how one can successfully use P-I-D controllers in their prospective paper and focus on almost all aspects of P-I-D controller.

10.References

[1] Kim, H-W., Choi, J-W. and Sul, S-K., Accurate position control for servo motor using novel speed estimator, IECON’95, 1995,vol. 1, pp. 627–632.

[2] Patel, D. M., High speed floating-point robot controller, CAMC’87, 1987, pp. 13–22.

[3] Ohm, D. Y. and Mazurkiewicz, J., Control of AC motors for servo applications, PCIM’91, 1991, pp. 288–298.

[4]Galvan, E., Barrero, F., Aguirre, M. A.,Torralba, A. and ranquelo, L. G., A robust speed control of AC motor drives based on fuzzy reasoning, IAS’93, Part III, 1993, pp. 2055–2058.

[5] Takakura, S., Murakami, T. and Ohnishi, K., An approach to collision detection and recovery motion in industrial robot, IEEE,1989, pp. 421–426.

[6] Orges Gjini, Takaynki Kaneko and Hirosh Ohsawa. "A new controller for PMSM servo drive based on the sliding mode approach with parameter adaptation". IEE Trans. On IA, V 123, No. 6, 2003, pp. 675-680

[7] Ogata K., "Modern control Engineering". New Jersey, Prentice-Hall, 1990.

[8] Miran Rodic, Larel Jezernic ,. Direct torque control of PWM inverter fed ac motors . a survey., IEEE Trans. On Indust. Electronics., Vol. 51, No. 4, August 2004, pp. 744-757.

[9] M. F. Rahman, L. Zhong, M. Haque, and M. A. Rahman, .A direct torque controlled Interior permanent magnet synchronous motor drive without a speed sensor., IEEE Trans. On Energy Conversion, Vol. 18, No.1,March2003,pp.17-22.

[10] Jawad Faiz and S.Hossien Mohseni . . A novel technique for estimation and control of stator flux of a salient pole PMSM in DTC method based on MTPF.. IEEE Trans. On Indust. Appl., Vol. 50,No. 2, April 2003,pp.262-270

v m venkateswara rao(M’) was born in chittoor , in 1985. He received the B.Tech degree in electrical engineering from JNT University Hyderabad and the M.Tech. Degree in electrical drives and control from the Pondicherry University in 2006 and 2009 respectively. He is currently an Assistant Professor in Dnyanganga College of engineering and research, pune. His research interests include motor servo drives, , nonlinear control, and intelligent control system including neural networks and fuzzy logic