IOP Conf. Series: Materials Science and Engineering **437** (2018) 012013 doi:10.1088/1757-899X/437/1/012013

Practical PID controller implementation for the speed control of a motor generator system

M Paez[1]**, J Viola**[1] **and L Angel**[1]

[1] Departament of Electronic Engineering, Universidad Pontificia Bolivariana, Bucaramanga Colombia

jairo.viola@ upb.edu.co

Abstract. Windup effect in a PID controller is the condition of cumulative error in the integral action. It saturates the system actuators, letting the system out of control with a higher controller action. Putting a saturator for the controller action, it is limited into a linear range of operating values. However, it does not eliminate the cumulative error of the integral term. This paper presents the speed control of a motor generator system using a digital PID controller with antiwindup scheme to eliminate the cumulative error of the integral term of controller action. Initially, a linear model of the motor generator system was identified. From the linear model of the system, a PID controller was designed using pole placement technique. Controller discrete implementation was perform using the backward difference discretization methodology. PID controller consider the antiwindup scheme. PID controller practical implementation employed Matlab Stateflow and a data acquisition card. Obtained results showed that discrete implementation of the PID controller has a good performance against external disturbances and random noise.

1. Introduction

PID controller is one of the most employed control techniques to control industrial process due to its easy design and implementation [1,2]. The design of PID controllers is to find the value of the proportional, integral, and derivate parameters to accomplish the desired closed loop specifications. The wind-up effect of the integral action is a common problem in the PID controllers. It is caused by the cumulative error on the integral term which can lead to the actuator saturation, causing the system to behave as if it were in open loop. This is a not desirable condition in the system because it disappear the control effect over the system.

To solve the wind-up effect, there are many antiwind-up methods as incremental algorithm, back calculation, conditional integration and the observer approach [1–5]. These techniques act over the controller integral action when the actuator reach the saturation limits.

This paper presents the design and implementation of a speed control for a motor-generator system employing a digital PID controller tuned by the pole placement technique. PID controller is discretized employing the backward difference approximation with the back calculation antiwind-up technique. Initially, the PID controller structure and the back calculation antiwind-up technique are presented. Then, the discrete PID controller model is obtained with the backward difference method. After that, the motor generator system is descripted and a linearized model of the system is identified. From the

Content from this work may be used under the terms of the Creative Commons Attribution 3.0 licence. Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.
Published under licence by IOP Publishing Ltd

linearized model, PID controller is designed using the pole placement method. Then, the digital PID controller and the antiwind-up technique are implemented employing the Matlab stateflow toolbox. To evaluate the performance of the digital PID controller, it is tested in the presence of external disturbances, random noise in the feedback loop and a resistive payload in the generator output. As performance indexes, the root median square error (RMSE) is employed for the system time response and the root mean square value (RMS) is employed for the control action. This paper is structured as follows. First, the PID controller structure and the antiwind-up technique are explained. Second, the PID controller discretization is developed. Third, the motor generator system description and identification are shown. Fourth, the practical implementation of the digital PID controller and its validation against external disturbances are performed. Finally, the conclusions are presented.

2. PID controller

From [1], a PID controller is represented in the frequency domain by the transfer function (1), where $U(s)$ is the control action, $E(s)$ the feedback system error, k_p the proportional constant of the system, τ_i the integral time constant and t_d the derivate time constant.

$$C(s) = \frac{U(s)}{E(s)} = k_p\left(1 + \frac{1}{\tau_i s} + \tau_d s\right) \tag{1}$$

3. Antiwind-up back calculation method

Figure 1 shows a block diagram representation of the back calculation antiwind-up technique with a PID controller. As shown, the system has an additional path, generated by measuring the actual actuator output or the output of a mathematical model of the actuator with saturation, forming a new error signal (es), which represent the difference between the controller output (v) and the actuator output (u). This signal is feedback to the integrator via gain $1/T_t$. This signal is zero when the actuator is not saturated and will have no effect on the control action. When the actuator becomes saturated, the signal (s) is different from zero, so, the controller output is placed at the upper or lower saturation limit and the integral term of the controller is recalculated so that its new value changes dynamically using the time constant T_t. This constant must be greater than τ_d and smaller than τ_i. A heuristic rule suggested by [1] stablish that $T_t = \sqrt{\tau_i \tau_d}$.

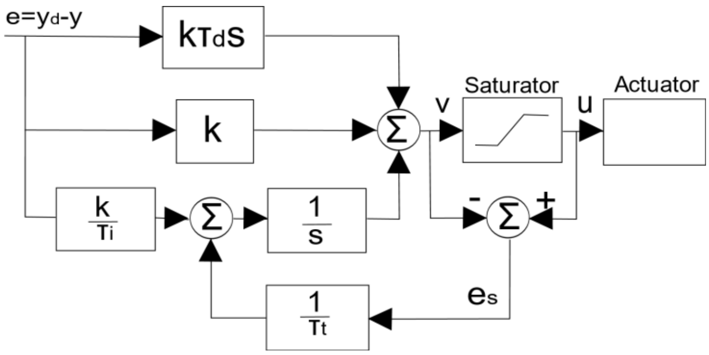


Figure 1. Back calculation antiwind-up scheme

4. PID controller discretization

The PID controller (1) can be discretized by different discretization techniques [6]. In this paper, the backward difference technique is employed, which approach the derivate of $x(t)$ into a differences equation as shown in (2), where T_s is the sampling time.

$$\frac{dx(t)}{dt} \approx \frac{x(k) - x(k-1)}{T_s} \tag{2}$$

IOP Conf. Series: Materials Science and Engineering **437** (2018) 012013 doi:10.1088/1757-899X/437/1/012013

From (2), the relation between the Laplace transform and Z transform on the frequency domain is given by (3).

$$s = \frac{1 - z^{-1}}{T_s} \tag{3}$$

Replacing (3) in (1)

$$C(z) = k_p + \frac{k_p T_s}{\tau_i}\left(\frac{1}{1 - z^{-1}}\right) + \frac{k_p \tau_d (1 - z^{-1})}{T_s} \tag{4}$$

Since $C(z) = U(z)/E(z)$, (4) can be expressed as:

$$U(z) = k_p E(z) + \frac{k_p T_s}{\tau_i}\left(\frac{1}{1 - z^{-1}}\right) E(z) + \frac{k_p \tau_d (1 - z^{-1})}{T_s} E(z) \tag{5}$$

$U(z)$ can be represented in function of the proportional $U_p(z)$, integral $U_i(z)$ and derivate $U_d(z)$ components as shown in (6).

$$U(z) = U_p(z) + U_i(z) + U_d(z) \tag{6}$$

where:

$$\begin{aligned} U_p(z) &= k_p E(z) \\ U_i(z) &= \frac{k_p T_s}{\tau_i}\left(\frac{1}{1 - z^{-1}}\right) \\ U_d(z) &= \frac{k_p \tau_d (1 - z^{-1})}{T_s} E(z) \end{aligned} \tag{7}$$

The representation given by (6) and (7) allows to implement the antiwind-up technique later. From (6) and (7), the following differences equations are obtained which represents the discrete PID controller model.

$$U(k) = U_p(k) + U_i(k) + U_d(k) \tag{8}$$

where:

$$\begin{aligned} U_p(k) &= k_p e(k) \\ U_i(k) &= U_i(k-1) + \frac{k_p T_s}{\tau_i} e(k) \\ U_d(k) &= \frac{k_p \tau_d (e(k) - e(k-1))}{T_s} \end{aligned} \tag{9}$$

5. Motor generator system

Fig 4 shows the motor generator system. This system is a motor-generator conformed by two DC motors Bodine brand with a maximum speed of 2500 RPM and $\pm 130v$ DC nominal voltage. As can be shown, the motors M1 and M2 are mechanically coupled by its axis where M1 works as motor and M2 as generator. M1 has a power driver M4 with an $\pm 10v$ input. A tachogenerator M3 with $\pm 7v$ output voltage is coupled to M2 for the speed feedback of the system.

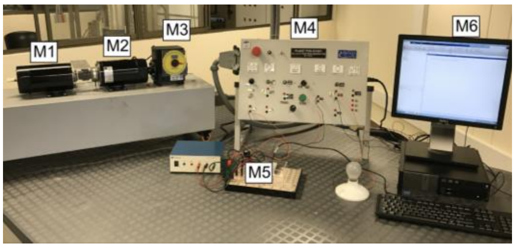


Figure 2. Motor Generator System

5.1. System identification

For the motor-generator system identification, the system is excited with a stepped speed profile shown in Figure 3 This profile allows to observe the dynamical behavior of the system for different operating points in both turn senses. As output signal for the identification process the tachogenerator voltage M3 is employed.

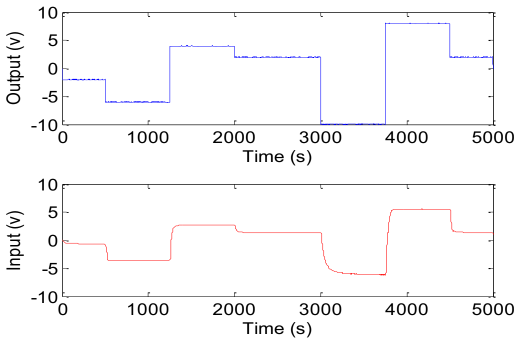


Figure 3. Identificacion trajectory

Data acquisition of the system is performed using a National Instruments data acquisition card NI DAQ 6008 M5 altogether with Matlab Simulink M6, where the steeped speed profile is generated and the system output is captured. From the acquired data, the model of the system is identified using the Matlab Identification Toolbox, resulting in a second order system with real poles given by (10) with a fit of 93%.

$$G_p(s) = \frac{0.62}{(21.725s + 1)(5.12s + 1)} \tag{10}$$

5.2. PID controller design

From the model of the system (10), the PID controller is tuned by the pole placement method. It is desired a closed loop overshoot of 10% and a setting time of $150s$. For this, a damping ratio $\zeta = 0.6$ and a frequency of $w_n = 0.04\, rad/s$. For a second order system with real poles (11) whose structure coincide with (10), the constants of the PID controller are calculated using the expressions (12) given by [1], where τ_1 and τ_2 are the time constants of the system k is the system gain and α is the location of a non-dominant pole of the closed loop system. Using (12) with $\alpha = 10$, the obtained constants for the PID controller are $k_p = 2.98$, $\tau_i = 19.02$ y $\tau_d = 15.35$.

$$G_p(s) = \frac{k}{(\tau_1 s + 1)(\tau_2 s + 1)} \tag{11}$$

$$k = \frac{\tau_1\tau_2 w_n^2(1+2\alpha\zeta)-1}{k} \qquad \tau_i = \frac{\tau_1\tau_2 w_n^2(1+2\alpha\zeta)-1}{\tau_1\tau_2\alpha w_n^3} \qquad \tau_d = \frac{\tau_1\tau_2 w_n(\alpha+2\zeta)-\tau_1-\tau_2}{\tau_1\tau_2 w_n^2(1+2\alpha\zeta)-1} \tag{12}$$

5.3. PID controller digital implementation using Matlab Stateflow

The digital implementation of the PID controller uses the difference equations (9) with the constants calculated above and a sampling time $T_s = 0.1\ s$. For the digital implementation of the back calculation antiwind-up technique proposed in Section 3, the pseudocode presented in Table1 is employed, where uk is the total control action, uik is the integral control action, lim_sat_p and $\lim\_sat\_n$ are the actuator saturation limits and T_t is the dynamic time reset constant of the integral action.

Table 1. Pseudocode for antiwindup back calculation digital implementation

Read *uk, uik, lim_sat_p, lim_sat_n*

If *uk>lim_sat_p* ***do****:*
$uik = uik + (1/T_t)(lim\_sat\_p - uk)$
$uk = lim\_sat\_p$

else if $uk < lim\_sat\_n$ ***do****:*
$uik = uik + (1/T_t)(lim\_sat\_n - uk)$
$uk = lim\_sat\_n$

else ***do****:*
$uk = uk$

To implement the digital PID controller tuned by the pole placement technique the matlab stateflow toolbox is employed with the data adquisition card NI DAQ which close the control loop for the motor generator system. Figure 4 shows the implementation of the PID controller employing the matlab stateflow toolbox. As can be observed, the digital implementation uses two states, the start state, and the control state. The start state is the first to be executed and it initializes the constants of the PID controller. In the control state, the error signal and the individual control actions are calculated as well as the antiwind-up calculation and the total control action. Variables uik_1, udk_1 and ek_1 are the previous values of the integral action, the derivate action, and the error signal.

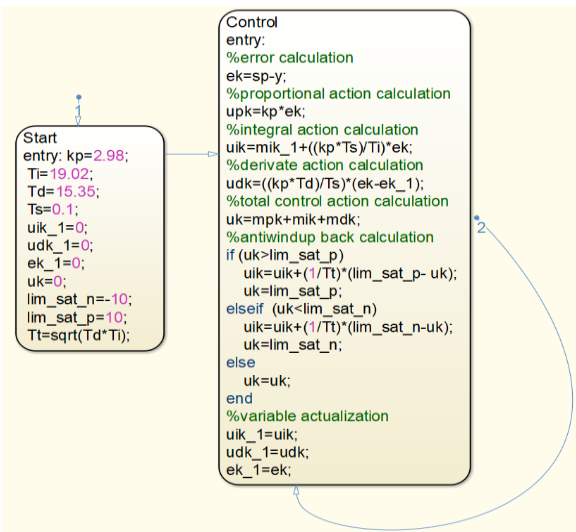
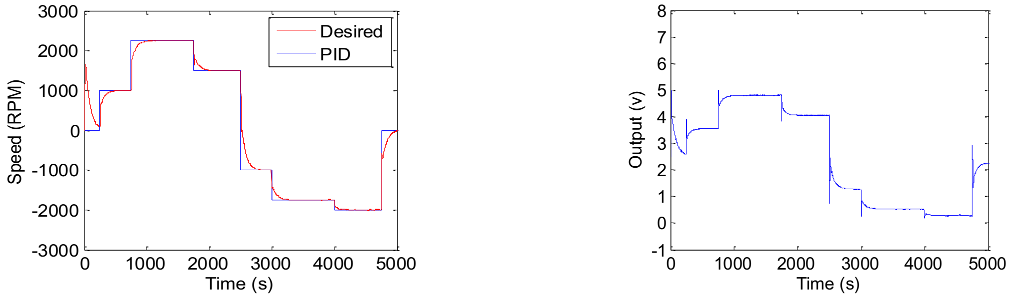


Figure 4. Matlab Stateflow PID implementation with antiwind-up back calculation technique

Figure 5 shows the motor-generator system time response and the control action of the digital PID controller tuned by the pole placement method for a stepped path which belongs to different desired speeds in both turn senses. As can be observed, The PID controller has a good tracking response and the control action is between the saturation limits.



(a) (b)

Figure 5. Time response (a) and control action (b) for the motor generator system with the digital PID controller

6. Robustness analysis

To evaluate the robustness of the digital PID controller, the time response of the system and the control action of the PID controller are tested in presence of external disturbance, random noise in the feedback loop and resistive payload at the generator output. As performance indexes, the RMSE value is selected for the time response and the RMS value is selected for the control action. Table 2 shows the obtained performance index values in presence of different disturbances. As shown, the PID controller performance is affected significantly by the presence of resistive payload in the generator output due to it has the greatest RMSE and RMS values regarding to the other disturbances.

Table 2. PID controller performance indexes in presence of external disturbances

Test	RMSE	RMS
Nominal operation	1	3,03
External disturbance	1,08	3,03
Random noise	1,09	3,05
Resistive payload	1,65	3,1

7. Conclusions

This paper presented the design and practical digital implementation of a PID controller applied to the speed control of a motor-generator system. The process was identified as a second order system with real poles. The PID controller was tuned employing the pole placement method and discretized through the backward difference discretization technique considering the implementation of the back calculation antiwind-up technique for the integral action. PID digital implementation was performed with the Matlab stateflow toolbox with a data acquisition card, showing a good performance for different operating points of the system. The PID controller performance was validated in presence of external disturbances, being the presence of resistive payload at the generator output that significantly affects the operation of the controller.

References

[1] Aström K J and Hägglund T 2009 *Control PID avanzado* (Madrid, España: Pearson Prentice Hall)
[2] Visioli A 2006 *Practical PID Control* (Springer London)
[3] Ribic A I and Mataurek M R 2016 Antiwindup PID controller design of TITO system based on ideal and inverted decoupling *2016 International Symposium on Industrial Electronics (INDEL)* pp 1–8
[4] Shin H B and Park J G 2012 Anti-Windup PID Controller With Integral State Predictor for Variable-Speed Motor Drives *IEEE Trans. Ind. Electron.* **59** 1509–16
[5] Wang Q, Wang C, An K, Shen G, Zhou X and Li Z 2014 Anti-windup PID position control of cylinders for a aircraft skin inspection robot with double-layer structure *Proceedings of the 33rd Chinese Control Conference* pp 8365–9
[6] Astrom K 1995 PID controllers: theory, design and tuning *Instrum. Soc. Am.* 343-