

Performing Speed Control of a DC Motor with Auto-Tuning PID

Said Mahmut ÇINAR¹, Zekeriya BALCI², İsmail YABANOVA³

¹Afyon Kocatepe Üniversitesi, Mühendislik Fakültesi, Elektrik Mühendisliği Bölümü, Afyonkarahisar.

²Van Yüzüncü Yıl Üniversitesi, Çaldıran Yabancı Meslek Yüksekokulu, Elektronik ve Otomasyon Teknolojisi Bölümü, Van.

³Afyon Kocatepe Üniversitesi, Teknoloji Fakültesi, Mekatronik Mühendisliği Bölümü, Afyonkarahisar.

Sorumlu yazar e-posta: smcinar@aku.edu.tr ORCID ID: <https://orcid.org/0000-0002-8683-3884>

zekeriyabalci@yyu.edu.tr ORCID ID: <https://orcid.org/0000-0002-1389-1784>

iyabanova@aku.edu.tr ORCID ID: <https://orcid.org/0000-0001-8075-3579>

Geliş Tarihi: 18.07.2019;

Kabul Tarihi: 22.11.2019

Abstract

Although advanced controllers are used in control applications, proportional-integral-derivative (PID) controllers are preferred in industry due to their simple structure and ease of application. However, it can difficult to set these controller parameters for the controlled platform. These parameters tuning with trial-error method leads to time and job loss, and the parameters determined in this way cannot provide a sufficiently efficient operating characteristic. In order to overcome this problem related to PID parameter tuning many automatic tuning methods have been developed. In this paper, the automatic tuning method proposed by Åström and Hägglund was applied to a DC motor speed control system. The DC motor speed control system was implemented in an interface designed on Laboratory Virtual Instrument Electronic Workbench (LabVIEW) environment and CompactRIO unit. The PID parameters obtained with trial-error and two types auto-tuning methods were tested in the DC motor control system and achieved results were compared. The results showed that performance of the PID controller tuned with LabVIEW auto-tuning method is better than others.

Keywords

Self-Tuning PID; DC motor; LabVIEW; CompactRIO; Speed control

DC Motorun Otomatik Ayarlamalı PID ile Hız Kontrolünün Gerçekleştirilmesi

Öz

Kontrol uygulamalarında gelişmiş kontrolörler kullanılsa da PID (Proportional-Integral-Derivative) kontrolörler basit yapısından ve kolay uygulanabildiğinden dolayı endüstride tercih edilmektedir. Ancak kontrol edilecek platforma uygun PID parametrelerinin ayarlanması oldukça güç olabilmektedir. Bu parametrelerin deneme yanılma yöntemiyle ayarlanması zaman ve iş kaybına sebep olmakla birlikte bu yolla tespit edilen parametrelerle ayarlanan kontrolörler yeterince verimli bir çalışma karakteristiği sunmayabilmektedir. PID parametrelerinin ayarıyla ilgili bu sorunların üstesinden gelebilmek için çok sayıda otomatik ayar yöntemi geliştirilmiştir. Bu makalede Åström ve Hägglund tarafından önerilen otomatik ayar yöntemi bir DC motoru hız kontrol sistemine uygulanmıştır. Bu DC motor hız kontrol sistemi LabVIEW (Laboratory Virtual Instrument Electronic Workbench) ortamında geliştirilen ara yüz ve CompactRIO ünitesi üzerinde gerçekleştirilmiştir. Deneme yanılma ve iki farklı otomatik ayarlama yöntemiyle elde edilen PID parametreleri DC motor kontrol sistemi üzerinde denenmiş ve elde edilen sonuçlar karşılaştırılmıştır. Sonuçlar, LabVIEW otomatik ayarlama yöntemiyle elde edilen parametrelerle işletilen PID kontrolörün daha iyi performans gösterdiğini göstermiştir.

© Afyon Kocatepe Üniversitesi

1. Introduction

Today, proportional-integral-derivative (PID) controllers are still popular in the industry. The PID has become a popular controller because of its good performance of in many linear or nonlinear

systems and the steady-state characteristic of various operating conditions. Moreover, the PIDs can be simply applied to many areas of the industry.

The PIDs show rapidly improvement with their ability to be digitally implemented on

programmable platforms such as microcontrollers. Many advanced control application such as self-tuning, and adaptive PID controllers can be easily applied with the using microcontrollers. The PID parameters are very important for the controlled system. Unfortunately, tuning the parameters is manually difficult, and this requires some information related to the system and the controller. In addition to, after manually tuning the PID parameters, it may be necessary to re-tune the PID parameters depending on the system dynamics and disturbances. Because of these difficulties, new methods are presented for automatically tuning the PID parameters. Procedure of the auto-tuning is consist of two step; firstly some experiments are made to definite the system characteristic, secondly parameters are calculated by an algorithm. In the industry, the Ziegler-Nichols (ZN) method is widely used to set automatically PID parameters. In the ZN method, an oscillation is created on the system output, and the oscillation of critical frequency and amplitude are determined. After determining the critical frequency and amplitude, the PID parameters can be calculated using ZN rule table. Because of the ZN methods is continuously executed under supervision and control, it is time consuming for practical application (Bolat et al. 2005, Tan et al. 2001, Jahanshahi et al. 2015). In the literature, a lot of studies have been carried out using the relay method (Ho et al. 1996, Liu et al. 2008, Hang et al. 2002). Some studies related to PID tuning are listed;

Marchetti et al. (2002) present an auto-tuning of multivariable system application on distributed control system. Wei et al. (2010) propose a simple method to auto-tune PID controller based on a modified relay feedback. The method was applied to pulp consistency control in pulp preparation and flowing section of paper machine and good performances was taken. Ho et al. (2003) auto-tuned the PID controller the relay experiment using iterative feedback tuning technique. The proposed

technique was tested in the laboratory on a coupled tank and good results was showed. Elmas et al. (2009) mentioned that high performance cannot be achieved with conventional PI controllers because of the nonlinear characteristic of switched reluctance motors. They suggested the need to adapt PI coefficients to achieve good performance under different speeds and loads, and they used a fuzzy logic controller for this adaptation. Levy et al. (2012) present a PID auto-tuning application based on Åström and Hägglund's method for industrial programmable logic controller (PLC) platforms. Kadu et al. (2015) realized the PID auto-tuning using relay based tuning method for two different process.

In this paper, we presented a speed control of DC motor with auto-tuning PID. The auto-tuning PID was performed on a Laboratory Virtual Instrument Electronic Workbench (LabVIEW) based platform with a simple and robust control structure. The platform was equipped with CompactRIO (CRIO) unit to control and observe the industrial process. Thanks to this platform, PID experiments was performed more easily and reliably and results was simultaneously observed via interface. The main contribution of this article is to introduce an experimental design in which PID parameters can be produced in a very simple manner.

2. Materials and Methods

In this section PID auto-tuning methods applied in this paper and DC motor speed control system are presented.

2.1. Auto-tuning methods

In this paper, two auto-tuning methods used to tune the PID parameters. First one of the methods is an auto-tuning method named as relay feedback and it is used in LabVIEW/PID toolbox. Second one is MATLAB auto-tuning method and it is used to

compare LabVIEW auto-tuning method. These auto-tuning methods is detail presented.

2.1.a. Auto-tuning methods with relay feedback: In order to achieve maximum performance and system efficiency, it is necessary to adjust the PID parameters according to process dynamics and disturbance effects. Determination of the PID parameters by trial and error method causes time loss and system performance is negatively affected too. For this reason, it has become important to determine the PID controller parameters using advanced tuning techniques (Hang et al. 2002).

Åström and Hägglund propose a PID tuning method known as relay feedback in 1984 (Wei et al. 2010). This tuning method is based on the determination of critical gain and critical frequency values. The tuning method is illustrated with the block diagram (Figure 1). The PID controller and, relay are activated with (a), and (b) position of switch, respectively.

In this method, to systematically determine the PID parameters, the system must first be steady-state. Then the switch shown in Figure 1 is taken to position (b), and an oscillation is started on the system output. Any control system generally increases its output, when increasing is occurred in its input. When the output over the set value, the relay signal is applied in the reverse direction and the system is allowed to enter the oscillation (de Sousa Neves 2009). The relay and system response signals are given in Figure 2 and 3, respectively. It is observed that the system enters the oscillation state due to the relay signal.

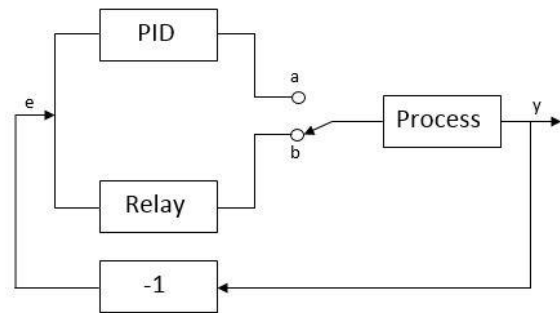


Figure 1. Principle diagram of PID tuning method with relay feedback.

The critical period (T_c) and gain (K_c) values of the oscillation can be calculated from the system response given Figure 3. In the practical applications, the critical period is calculated determining zero crossing point of the oscillation.

The critical frequency (ω_c) is calculated by the formula (1) using of the critical period (T_c) value.

$$\omega_c = \frac{2\pi}{T_c} \tag{1}$$

The critical gain (K_c) is calculated using the equation (2).

$$K_c = \frac{4d}{\pi\sqrt{a^2 - \varepsilon^2}} \tag{2}$$

where;

d = relay amplitude, V

a = amplitude of oscillation signal, V

ε = hysteresis value, s

Using the critical values of T_c , and K_c , the PID gain values are calculated with the ZN rule table (Table 1).

Table 1. Table of ZN calculation according to T_c and K_c .

Controller	K_p	K_i	K_d
P	$0.5K_c$	-	-
PI	$0.45K_c$	$T_c/1.2$	-
PID	$0.6K_c$	$T_c/2$	$T_c/8$

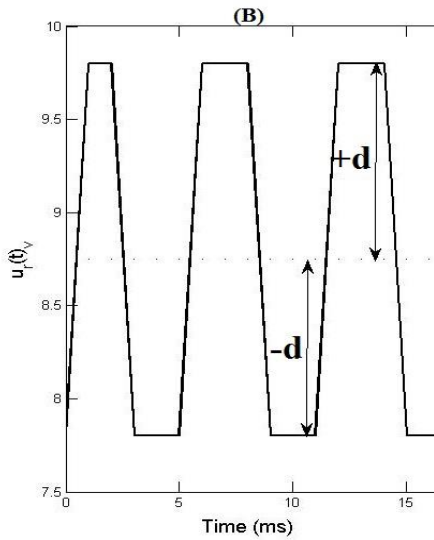


Figure 2. Relay signal - $u_r(t)$.

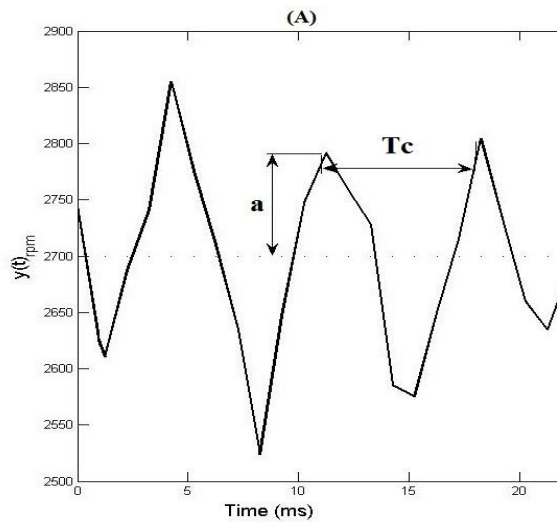


Figure 3. Response signal of the system - $y(t)$.

2.1.b MATLAB auto-tuning method: MATLAB software is widely used academic studies. MATLAB software has a lot of toolbox, and there are some auto-tuning tools such as “pidTuner” for PID tuning jobs. The pidTuner can compute the PID parameters considering to plant transfer function or input-output data of plant.

The pidTuner can be run typing "pidTuner" in MATLAB command window (Figure 4). In the PID tuner tool, input-output data of the plant are firstly imported, then plant is identified, and PID parameter are finally computed according to the identified plant.

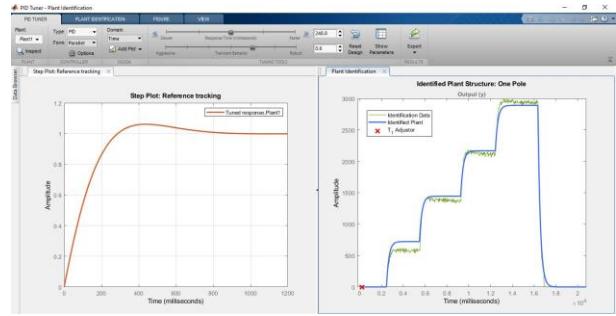


Figure 4. MATLAB/PID Tuner tool window

In the study, PID parameters were computed using the pidTuner. DC motor supply voltage (Input) and rotor speed (output) signals were acquired from DC motor driving system, and the acquired data were imported to the pidTuner. After the importing data, pidTuner was automatically computed the PID parameters according to the selected PID type.

2.2. System components

In this study, to tune parameters of the PID controller which is realize controlling of a DC motor, PC based test platform was used. The platform was equipped with a DC motor, a DC generator coupled with DC motor, an amplifier circuit of DC motor, and a CompactRIO data acquisition module. The CompactRIO module is produced by National Instrument (NI) company and, it performs some features such as conditioning of feedback and control signals. The system components are presented detail in below.

2.2.a. Motor-generator unit: This unit is composed of coupling DC motor and generator to realize DC motor speed control process (Figure 5). Speed value of DC motor is converted to voltage by a transducer. The model specifications are given;

Supply voltage range:	±15Vdc symmetrically
DC motor power:	10W
Maximum speed of DC motor:	300rpm
Output voltage range of DC generator:	0 ÷ 20Vdc
Output resolution of speed transducer:	1mV/rpm

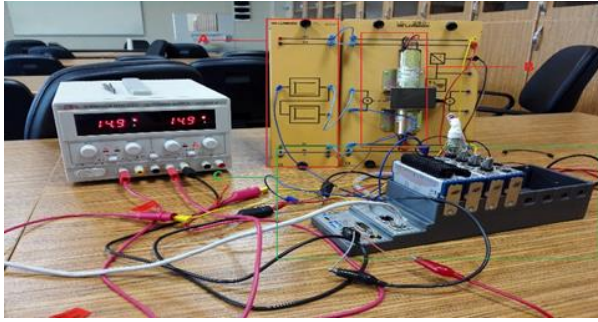


Figure 5. PID controller experimental setup.

2.2.b. Amplifier circuit: The circuit is formed with inverter and non-inverter two amplifier circuits. The amplifier gains are +1 and -1, respectively. The amplifier specifications are given;

Supply voltage range: $\pm 15\text{Vdc}$ symmetrically
 Input voltage range of the amplifier: $-10\text{--}+10\text{Vdc}$

2.2.c. CompactRIO unit: The CompactRIO (CRIO) is a unit to control and observe the industrial process, and it is putted by NI company on the market. The CRIO is a compact system combined with real time controller and field programmable gate array (FPGA). The CRIO specifications are given;

Supply voltage range: 19–30Vdc
 Processor frequency: 400MHz
 FPGA type: Spartan-6LX25
 DRAM capacity: 128MB
 Internal memory: 256MB

2.2.d. Software environment: Interface used to control the DC motor system is designed with LabVIEW software. The LabVIEW is putted by NI company on the market in early 1980s. The LabVIEW is an object-oriented software focused measuring and instrumentation. Because highly capable of process control and data acquisition process control applications are easily implemented with the LabVIEW (Kutlu and Turan 2010).

An interface was designed with the LabVIEW (Figure 6). In the interface some components were indicated by frame and number. These were; program control buttons (1), PID gains and PID initial gains (2), auto tuned PID gains (3), graphical chart of control and system output and set point signals (4), and controller setting components (5).

3. Results and Discussion

Firstly system components; motor-generator, amplifier, and CRIO modules were setup as making the mechanical and electrical connections. Than system was supplied by symmetric DC power supply. Finally, communicate between the experimental setup and interface were obtained via Ethernet port (Paç and Nadar 2008). After the system setup we started the PID controller tests.

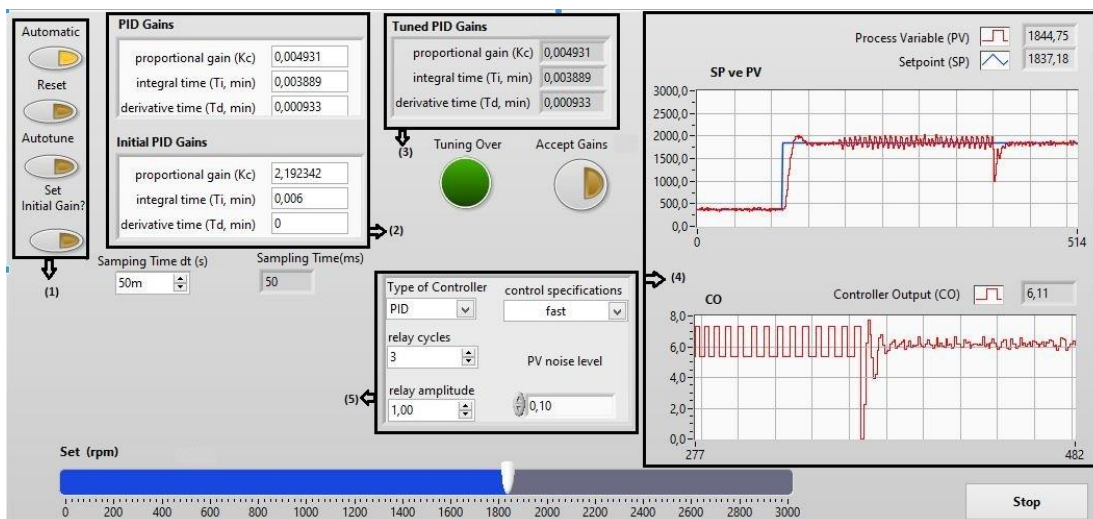


Figure 6. Interface designed for auto tuning.

PID parameters were manually adjusted with trial and error, and system was run with these parameters. After this the system output was come close to set point, and then tuning mode was switched to automatic mode. In this manner, PID parameters were automatically adjusted.

PID parameters were manually adjusted with trial and error, and system was run with these parameters. After this the system output was come close to set point, and then tuning mode was switched to automatic mode. In this manner, PID parameters were automatically adjusted.

The PID parameters tuned with trial-error, MATLAB and LabVIEW auto-tuning methods were applied to the system, and step responses of the system were compared. In the Figure 7, black, red, blue, and green graphs were represented set point signal, LabVIEW auto-tuning (purposed), MATLAB auto-tuning and trial-error tuning step response signals, respectively. The results were shown that in case of LabVIEW auto-tuning low overshoot and short settling time values were achieved according to trial-error and MATLAB auto-tuning methods.

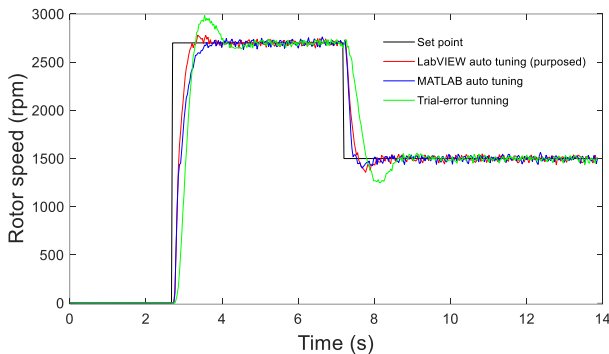


Figure 7. DC motor speed variation accordingly PID tuning methods.

Table 2. K_p , T_i , and T_d parameters for tuning methods

Tuning methods	K_p	T_i	T_d
Trial-error tuning	0.0002	0.0003	0.000004
MATLAB auto-tuning	0.00495	0.0035	0.0007
LabVIEW auto-tuning (purposed)	0.002874	0.002417	0.00058

4. Conclusion

In this study, it is aimed to automatically tune the PID parameters. The PID parameters tuned to using trial-error, MATLAB and LabVIEW auto-tuning methods were applied to the system and obtained results were comparatively examined. The data collected from the system were transferred by CRIO to the interface. Time interval of data acquisition was set to 50ms for this study. This was caused that some program blocks which has smaller than 50ms time interval could not be run. To overcome this problem data acquisition and tuning time intervals can be decreased. In this study any load or distortion was not applied to the DC motor, these can be applied and their effects can be examined. Than a self-tuning controller can be developed in the future.

References

Bolat, E.D., Erkan, K. and Postalcioglu, S., 2005. Experimental Autotuning PID Control of Temperature Using Microcontroller. EUROCON, Serbia & Montenegro, Belgrade, 266-269.

de Sousa Neves, M.G., 2009. Relay method on auto-tuning automation solutions, Universidade Técnica de Lisboa, Instituto Superior Técnico.

Elmas, Ç., Akcayol, M.A. and Yiğit, T., 2009. Bulanık PI Denetleyici ile Bir Anahtarlamalı Relüktans Motorun Hız Denetimi, *Gazi Üniv. Müh. Mim. Fak. Der.*, **22/1**, 65-72.

Hang, C.C., Aström, K.J. and Wang, Q.G., 2002. Relay feedback auto-tuning of process controllers — a tutorial review, *Journal of Process Control*, **12/1**, 143-162.

Ho, W.K., Feng, E.B. and Gan, O.P., 1996. A Novel Relay Auto-Tuning Technique for Processes With Integration, *Control Eng. Practice*, **7/4**, 923-928.

Ho, W.K., Hong, Y., Hansson, A., Hjalmarsson, H. and Deng, J.W., 2003. Relay auto-tuning of PID controllers using iterative feedback tuning, *Automatica*, **39**, 149-157.

- Jahanshahi, E., Sivalingam, S. and Schofield, J.B., 2015. Industrial test setup for autotuning of PID controllers in large-scale processes: Applied to Tennessee Eastman process, *IFAC-PapersOnLine*, **48**, 469-476.
- Kadu, C.B., Bhusal, S.B. and Lukare, S.B., 2015. Autotuning of Pid Controller for Robot Arm and Magnet Levitation Plant, *International Journal of Research in Engineering and Technology*, **4/1**, 186-193.
- Kutlu, A. and Turan, C., 2010. Elektronik Deney Modüllerinin Labview ile Kontrolü, *SDU International Journal of Technologic Sciences*, **2/3**, 1-8.
- Levy, S., Korotkin, S., Hadad, K., Ellenbogen, A., Arad, M. and Kadmon, Y., 2012. PID autotuning using relay feedback, IEEE 27th Convention of Electrical and Electronics Engineers in Israel, Eilat, 1-4.
- Liu, T. and Gao, F. 2008. Relay-Based Autotuning of PID Controller for Improved Load Disturbance Rejection, Proceedings of the 17th World Congress The International Federation of Automatic Control, Seoul, Korea, July 6-11, 10933-10938.
- Marchetti, G., Scali, C. and Romagnoli, J.A., 2002. Relay Autotuning Of Multivariable Systems: Application To An Experimental Pilot-Scale Distillation Column, 15th Triennial World Congress of the International Federation of Automatic Control, Barcelona, 21–26 July, 127-132.
- Paç, M.R. and Nadar, A., 2008. Hidroelektrik Santral (HES) Otomasyonu için Ethernet Tabanlı Bir Haberleşme Alt-Sistemi Tasarımı, Elektrik-Elektronik-Bilgisayar Mühendisliği Sempozyumu ve Fuarı 26-30 Kasım (ELECO-2008), Bursa, 1-5.
- Tan, K.K., Lee, T.H. and Jiang, X., 2001. On-line relay identification, assessment and tuning of PID controller, *Journal of Process Control*, **11**, 483-496.
- Wei, T., Zhongfeng, W., Jinguo, Z. and Mengxiao, W., 2010. A Relay Feedback Based Autotuning PID Controller and Its Application to Pulp Consistency Control, Proceedings of the 29th Chinese Control Conference, Beijing, 3406-3410.