Volume 16, Number 2, 2014

Global Journal of Engineering Education

Robust design with PID controllers for implementing undergraduate students' laboratory exercises

Abubakar S. Bappah

Abubakar Tafawa Balewa University
Bauchi, Nigeria

ABSTRACT: PID (proportional integral derivative) controllers are the backbone of most undergraduate students' laboratory exercises, as well as industrial control systems. When operated in manual mode, they are normally set into operation using their default parameters through a simple and comprehensive rule of the thumb. The problem is that these loops often result in poor operation and regular retuning is essential to cater for ageing, non-linearity and time varying process characteristics. So, over the last few years, there has been significant development in the process control area to adjust the PID controller parameters automatically to achieve adequate servo and regulatory behaviour for a closed-loop plant. This article presents a MATLAB-based robust PID compensation design that will guide undergraduate control students and amateur control enthusiasts to implement their pertinent laboratory exercises. The design utilises a user friendly M-File whose input parameters are adjustable at the beginning of each simulation session. Simulation results were used to elicit the flexibility of PID controller over a wide variety of operating conditions.

Keywords: PID control, robust design, tuning methods, setpoint kicks, integral windup, auto-tuning, relay method

INTRODUCTION

The benchmark minimum academic standards for the training of undergraduate students in the areas of engineering and technology stipulates that 60% of their instruction should be in the form of practical laboratory experience to rise to the challenges of a developing economy [1]. The importance of laboratory practical in the training of the graduates was adequately emphasised in the benchmark, such that a minimum of nine hours per week (3 credits) should be spent on students' laboratory works. To achieve this, all the laboratory practical classes have been lumped together to form a course which the student must pass. These practical classes are designed to follow the trend in the current development of the programmes.

Standard reference textbooks of control engineering systems have established a well-balanced coverage of control theory and laboratory practice as evidenced in control systems engineering [2], digital control systems [3], process control [4], modern control systems [5] and robust process control [6] among others. The PID ((proportional integral derivative) control is the most common form of feedback control found in all these books. It was an essential element of early governors and it became the standard tool when process control emerged in the 1940s. Precisely, more than 95% of the controllers in process control applications are of PID type [7].

The continued acceptance of the PID controller is partly due to its practical performance in most continuous industrial processes, and partly due to its widespread understanding among the young students and trainees. For many processes, it gives good performance with reasonable robustness to incorrect process model assumptions and limited process parameter changes. Essentially, the controller is easy to understand amongst non-specialist plant operators, with tuning rules that have been validated in a wide variety of practical cases. The problem of determining PID controller parameters, then, is of great importance in the professional control domain.

Due to the importance of the problem, it has remained an active area of research, both among academics and in industry, with scholarly articles on this subject claiming better tuning strategies appearing regularly in process control journals. Many PID controllers nowadays incorporate auto-tuning capabilities [8]. Despite there being considerable information in the literature, PID tuning is still an active area of research due to the recognition that the majority of PID controllers in the field are poorly tuned. Selecting appropriate PID controller parameters for a given plant is often known as controller tuning. Provided these parameters are well chosen, PID controllers can control a considerable range of continuous technological processes. Nowadays, many different methods for PID controller design for continuous systems exist. These methods are compared in terms of quality of control. There are four major characteristics of the closed-loop step response: rise time, overshoot, settling time and steady-state error.

However, a good number of PID controllers are still poorly tuned due to a lack of understanding of process dynamics, a lack of understanding of the PID algorithm or a lack of knowledge regarding effective tuning procedures. This article reviews the basics of PID control, paying particular attention to design issues of the ideal PID controller often quoted in the technical literature. Methods that have evolved over the last 50 years as aids in control loop tuning are captured and simulated in as a MATLAB M-file to provide a self-taught guide for control students and practitioners.

On one hand, the article develops students' and practitioners' understanding of control system design such that they should have sufficient knowledge to effectively tune a PID control algorithm under variety of industrial operating conditions. On the other hand, it provides them with a MATLAB-based simulation tool that will guide them in their regular class work and drill exercises. A simulation laboratory is an educational tool for teaching students, the basic principles and methodology in performing a series of experiments in Control Engineering Systems class. The laboratory introduced in this article has the capacity to permit the implementation of strategies for conventional and robust proportional integral derivative (PID). It is a platform on which the undergraduate students and control practitioners test control algorithms. This approach is distinct from others in that it offers the user more flexibility and responsibility, since he/she compiles and executes the controller individually.

GENERIC PID DEIGN TECHNIQUES

In order to get the best performance from a PID controller, the amount of each control action has to be selected carefully. Provided these parameters are well chosen, PID controllers can control a considerable part of continuous technological processes [9]. A method of selecting appropriate PID parameters for a given plant is often known as controller tuning. Books have been written and articles on this subject claiming better tuning strategies appear regularly in process control journals [10]. The choice of appropriate controller parameters may be achieved experimentally, that is, by manual tuning. However, such an approach is time consuming and the process typically has to be driven to its stability limit [11].

A variety of methods have been proposed for gaining the information required for tuning, but must require the plant to be disturbed in some way. The process reaction curve was the first such tuning method for the tuning of P, PI and PID controller parameters of a process. This method was based on calculating the controller parameters from the model parameters determined from the open loop process step response. Originally, Ziegler and Nichols modelled the single input/single output (SISO) process by a first order process plus deadtime (FOPDT) model estimated the model parameters using a tangent and point method and defined tuning parameters for the P, PI and PID controllers.

The ultimate cycle method as defined by Ziegler and Nichols calculates parameters from the controller gain and oscillation period recorded at the ultimate frequency (i.e. the frequency at which marginal stability of the closed loop control system occurs). They realised that if the gain margin could be estimated quickly, then, it should be possible to find good controller settings for many practical situations from this information. They, therefore, suggested that one way to do this in practice would be by setting a PID controller into the P mode and adjusting the gain until an oscillation took place. Since nearly all the PID controllers then were pneumatic, the values of controller gain k_c, and the oscillation frequency, ω_c were easily measured.

However, a procedure that has received much attention recently is relay auto-tuning where the controller switches in the tuning mode to operate as an on-off relay and obtains data from the resulting limit cycle [12][13]. The resulting limit cycle data can be used to estimate the critical point of an assumed plant transfer function, that is, where its frequency response has a phase slight of 180˚ and, thus, may be regarded as an automated Ziegler-Nichol's test [14]. Assuming a linear plant model, then, the frequency of oscillation ω_c on the Nyquist plot of the plant frequency response has a phase shift of -180˚ and the gain margin is $1/k_c$ expressed in decibels (dB), the desired optimal PID controller parameter settings are obtained as recommended by D.P. Atherton [15].

Robust methods, based on the internal model control (IMC) design procedure, may be used to design analytically an appropriate PID controller for a FOPDT process model both with delay uncertainty and with general parameter uncertainty [6]. In practical implementations the PID controller ensures bumpless transfer from manual to automatic mode. For some plant, however, these approaches may not yield suitable parameters, particularly for systems characterised by large and varying dead time [16]. This has been the motivation for some methods that aim to select the PID terms, such that some optimisation criteria are minimised. Optimisation criteria may be used to map optimal PID parameters by artificial neural networks, fuzzy logics or more recently genetic algorithms [17][18].

These have included backward rectangular method, trapezoidal rectangular method and Tustin approximation. This implies firstly, that the majority of applications do not appear to be critical at least as far as stability is concerned. Secondly, which is far more relevant to this study, the importance of having PID loops tuned correctly is often underestimated. Since few loops are independent, a poorly tuned loop means more hassle for the upper levels of the control hierarchy, where interactions are even more evident. Thus, having a loop tuned incorrectly often results in the necessity of taking more complex solutions at the higher hierarchy levels or of reducing the overall plant expectations. The problem is that these loops must be identified, and above all that every loop must be tuned as well as possible, which is actually very time consuming.

LIMITATIONS OF THE GENERIC PID STRUCTURE

Despite their widespread use and considerable information in the literatures, PID tuning is still an active area of research, because of its wide spread application and the recognition that majority of PID controllers in the field are poorly tuned. Most automatic PID controllers cannot cater for the practical realities of ageing, non-linearity and time-variation [16]. Strategies adopted in practice to address these issues are known as robust PID design techniques, so over the last few years, increasing effort to cater for the following cases has been observed:

- High order processes.
- Processes with harmonic disturbances.
- Highly coupled multi-input-multi-output systems.
- Processes with large uncertainties or non-linearity.
- Processes with the time delay greater than the time constant of a FOPDT process model.
- Processes with large variations that have a significant frequency content around the systems resonant frequency of oscillation.

ROBUST PID DESIGN

The PID controller may be implemented in continuous or discrete time, in a number of controller structures. A robust continuous time PID controller can be deduced from the simplified block diagram of a generic PID control scheme as shown in Figure 1.

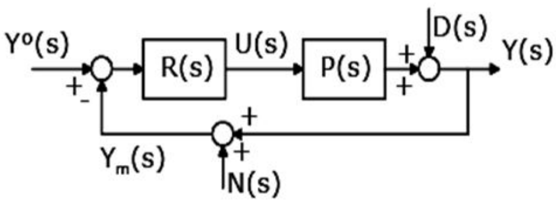

Figure 1: Robust PID control design.

The PID controller is governed by a mathematical expression also known as the control algorithm, which can be represented by the following expression:

$$u(t) \quad = \quad K\left[e(t) + \frac{1}{T_i}\int e(t)dt + T_d \frac{de(t)}{dt} \right] \tag{1}$$

This corresponds to the ideal continuous time PID controller, when expressed in the form of the transfer function from E(s) to U(s):

$$R(s) \quad = \quad \frac{U(s)}{E(s)} \quad = \quad k_c\left[1 + \frac{1}{sT_i} + sT_d \right] \tag{2}$$

with k_c = proportional gain, T_i = integral time constant, and T_d = derivative time constant. Here some definitions are consequently in order to clarify notation:

- Y°(s) is the reference signal;
- Y(s) is the controlled variable or process variable (PV);
- $Y_m(s)$ is the measured PV fed back to the PID controller;
- U(s) is the control signal or manipulated process variable, assumed to be limited by two bounds U_{min} and U_{max};
- The process is assumed to be described by a linear, time invariant dynamic system in the form of the transfer function P(s) while R(s) is the transfer function of the (PID) controller, which is considered linear as well;
- The controller input is the apparent error, E(s). The control objective is to lead Y(s) (not $Y_m(s)$) to Y°(s);
- The *true* error is defined as $E_{true}(s) = Y°(s) - Y(s)$. However the controller can only act on the measurement $Y_m(s)$ of Y(s) only since N(s) is by definition unknown.
- An *apparent* error can be defined as $E_{apparent}(s) = Y°(s) - Y_m(s)$. This is the *error* the controller will try to make as small as possible, and is a good representation of $E_{true}(s)$ if N(s) is small.

- This means that the controller has *one degree of freedom* (1-d.o.f.) in that the transfer functions from Y°(s) to U(s) and from $Y_m(s)$ to U(s) differ only by sign, so it is not possible to specify how the control will react to a set point change and to a measurement change separately. Such controllers are frequently termed *error-input* in the professional literature [9].
- D(s) and N(s) are disturbance and measurement noise, respectively. There are usually more than two sources of disturbances and noise in practice.

SIMULATION RESULTS

MATLAB provides tools for automatically choosing optimal PID gains, which makes the trial and error process described in the previous sections unnecessary. The tuning algorithm can be accessed directly using *pidtune* or through a user-friendly graphical user interface (GUI) using *pidtool.* The MATLAB automated tuning algorithm chooses PID gains to balance performance (response time, bandwidth) and robustness (stability margins).

A comparative simulation of loop performance under different operating conditions was carried out to evaluate the most appropriate controller parameters with respect to process parameter variations. This made it possible to undertake comparative analysis of the performance of each candidate solution in both the time and frequency domains to determine the most suitable set of PID controller parameters, at least in the sense of the design objectives.

Varied behaviours of industrial processes were studied. Effects of measurement noise, load disturbances and auto-manual transfer on loop performance have been investigated in this article.

Effects of Step and Load Disturbances

A step input signal was used to excite the control loop while the tuned PID controller parameters obtained under the normal conditions were used in initialising it. PID controller performance in the presence of load disturbance was investigated with a load disturbance of magnitude of unity introduced to the control loop at the 27th second for both Ziegler-Nichols open and closed loop responses as shown in Figures 2 and 3 respectively. The best compromise between performance and robustness is achieved when this method is used.

The Ziegler-Nichols open loop tuning method presented excessive oscillations in both its step and load disturbance responses. This indicates poor quality of disturbance rejection. The Ziegler-Nichols close loop tuning method has a step response, which settled within 10 seconds with just acceptable overshoot level. The relay based method adequately settled at its quiescent position within 10 seconds of load disturbance.

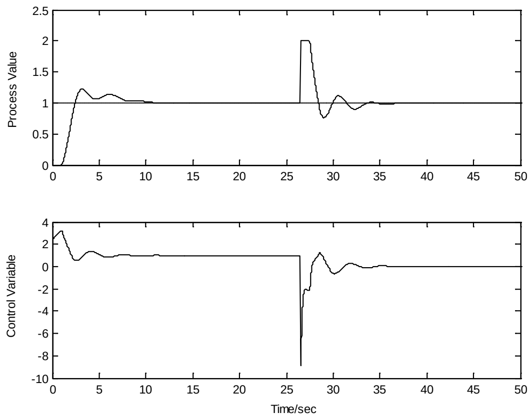

Figure 2: Step and load disturbance response with Ziegler-Nichols open loop.

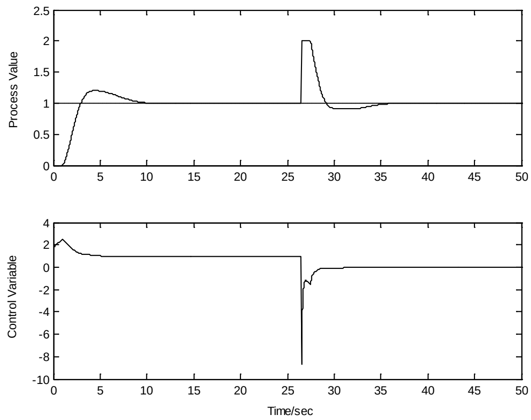

Figure 3: Step and load disturbance response with Ziegler-Nichols closed loop.

Effects of Auto-Manual Transfer

Process control loops are often required to be switched between automatic to manual control modes. In practical implementations, the PID controller ensures bumpless transfer mechanism between manual and automatic control operation. In this study, the control loop was left to run for 10 seconds at which instant the controller mode was stepped from auto to manual operating mode.

Figures 4 and 5 are results of an investigation to establish the performance of each tuning method with respect in the event of auto-manual transfer. Controllers based on the Ziegler-Nichols open loop method exhibited oscillatory CV profiles at the point of auto-manual switch, a feature which is detrimental to the life of process actuator. Corresponding responses due to the Ziegler-Nichols closed loop tuning method are without bumps, but may not be acceptable due to the excessive spikes in the control variables.

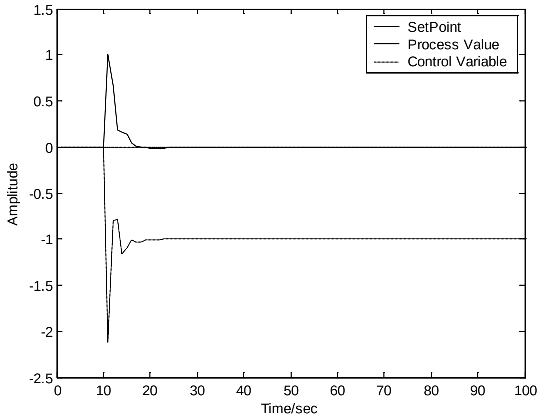

Figure 4: Auto-manual transfer response with Ziegler-Nichols open loop.

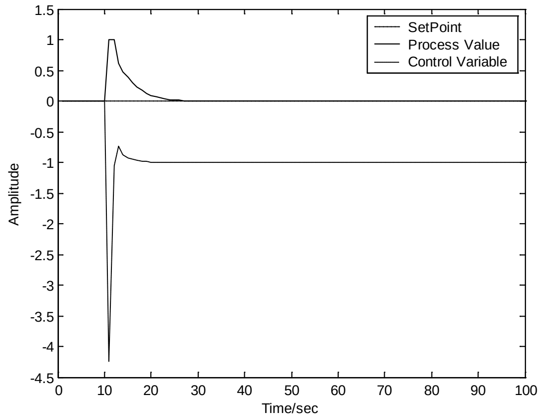

Figure 5: Auto-manual transfer response with Ziegler-Nichols closed loop.

Bode Diagrams of Sensitivity Functions of the PID Controller

Figures 6 and 7 present Bode diagrams of the sensitivity function with different PID structures. Both figures have presented a good closed-loop stability margin. This is in agreement with the time response descriptions of the process under investigation regardless of the method used in the controller synthesis.

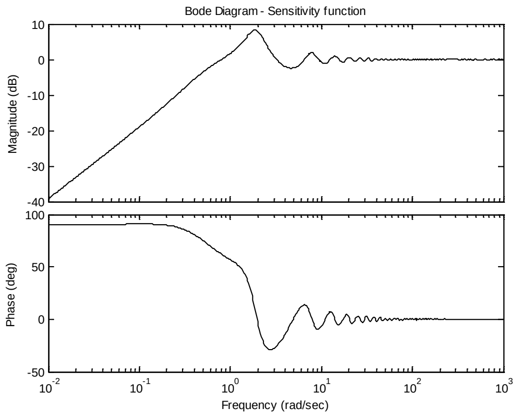

Figure 6: Bode diagram of the sensitivity function with Ziegler-Nichols open loop.

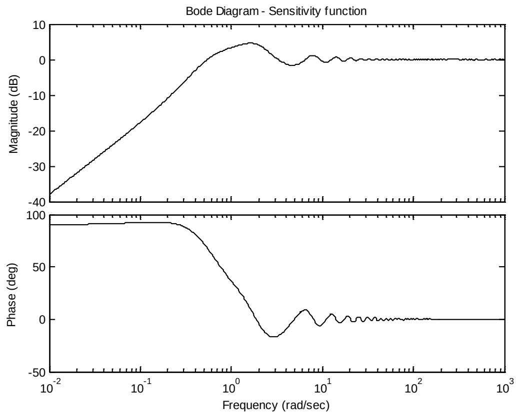

Figure 7: Bode diagram of the sensitivity function with Ziegler-Nichols closed loop.

CONCLUSIONS

Control academics and practitioners remain interested in the use of the PID controller with an understandable resistance to move away from the well-established algorithm [19]. However, most PID control loops are poorly tuned due to a lack of understanding of process dynamics, a lack of understanding of the PID algorithm or a lack of knowledge regarding effective tuning procedures. The suggestion is that if a PID can be properly tuned, there is much scope to improve its operational performance.

This article provides a self-taught guide to control practitioners and engineering students, such that they should be capable of approaching a loop tuning problem in a competent and efficient manner and have sufficient knowledge to effectively tune a PID control algorithm.

The article contributes towards the expected knowledge, attitudes and skills for graduates and their ability to fit into the requirements of the new national and global economy. Several scenarios were investigated. Firstly, comparison of the performance of PI and PID controllers was undertaken in absence of load disturbances and measurement noise. Investigation was conducted also on disturbance rejection and auto-manual transfer. It was found that if premium is put on speed of response the most preferable controller tuning method is that based on Ziegler-Nichols open loop. If peak overshoot is to be minimised then the most preferable tuning method is the Ziegler-Nichols relay technique.

REFERENCES

1. National Univerities Comission. Benchmark Minimum Academic Standards for Undergraduate Programmes in Nigerian Universities: Engineering & Technology. Abuja: National Universities Commission Press, 1-21 (2007).

2. Nagrath, I. and Gopal, M., *Control Systems Engineering*. (5th Edn), New Delhi: New Age International Ltd, 477-482 (1975).
3. Kuo, B.C., *Digital Control Systems*. New York: Oxford University Press, Inc., 375-401 (1995).
4. Marlin, T.E., *Process Control: Designing Processes and Control Systems for Dynamic Performance*. (International Edition 2nd Edn). Singapore: McGraw-Hill Inc., 211-219 (2000).
5. Ogata, K., *Modern Control Systems.* University of Minnesota: Prentice Hall Inc., 288-296 (1987).
6. Morari, M. and Zafiriou, E., *Robust Process Control*. (International Student Edn). New Jersey: Prentice Hall Inc., 162-183 (1989).
7. EnTech, Automatic Controller Dynamic Specification: Summary of Version 1.0, 11/93. Austin: Emerson Process Management, 1-5 (1993).
8. Bappah, A.S., Strategies for improving engineering students' efficacy in PID control laboratory. *Energy Educ. Science and Technol. Part B: Social and Educational Studies*. 5, **1**, 227-236 (2013).
9. Daley, S. and Liu, G., Optimal PID tuning using direct search algorithms. *Computing & Control Engng J.,* 10, **2**, 51-56 (1999).
10. Knospe, C., PID control. *IEEE Control Systems Magazine.* 26, **1**, 30-31 (2006).
11. O'Dwyer, A., PID compensation of time delayed processes: a survey. *Proc. Irish Signals and Systems Conf.*. Dublin, Ireland, 5-12 (2000).
12. Leva, A., PID autotuning algorithm based on relay feedback. *IEE Proc. D: Control Theory and Applications.* 140, **5**, 328-338 (1993).
13. Hang, C., Astrom, K. and Wang, Q., Relay feedback auto-tuning of process controllers - a tutorial review. *J. of Process Control.* 12, **1**, 143-162 (2002).
14. Nishikawa, Y., Sannomiya, N., Ohta, T. and Tanaka, H., A method for auto-tuning of PID control parameters. *Automatica.* 20, **3**, 321-332 (1984).
15. Atherton, D.P., PID controller tuning. *Computing & Control Engng. J.,* 10, 2, 44-50 (1999).
16. Atherton, D.P. and Majhi, S., Limitations of PID controllers. *IEEE 1999 Proc. American Control Conf.*. San Diego, USA, 3843-3847 (1999).
17. Ruano, A., Fleming, P. and Jones, D., Connectionist approach to PID autotuning. *IEE Proc. D: Control Theory and Applications.* 139, **3**, 279-285 (1992).
18. Shardt, Y., Zhao, Y., Qi, F., Lee, K., Yu, X., Huang, B. and Shah, S., Determining the state of a process control system: current trends and future challenges. *The Canadian J. of Chemical Engng.*, 90, **2**, 217-245 (2012).
19. Daley, S., Developing controllers through a process of natural evolution. *The IEE Computing and Control Engng. J.*, 10, **2**, 42-43 (1999).

BIOGRAPHY

Abubakar Sadiq Bappah is an immediate past Head of Department and Associate Professor of Technical and Engineering Education at the School of Technology Education, Abubakar Tafawa Balewa University, Bauchi, Nigeria. He holds a Bachelor degree (1994); MEng (2000) and PhD (2006) in Technical Education and Engineering, respectively. The author is a member of the Institute of Electrical Electronics Engineers (MIEEE); a member of the World Institute for Engineering and Technology Education (WIETE) and the Nigerian Association of Teachers of Technology (MNATT) as National Vice-President (North-East). His research interests include innovation strategies, system modelling and simulation, and technical, engineering education and training. Dr Bappah is happily married with children.