2004-01-01

Tuning rules for PI and PID control of time delayed processes: some recent developments

Aidan O'Dwyer
Dublin Institute of Technology, aidan.odwyer@dit.ie

Tuning rules for PI and PID control of time delayed processes – some recent developments

Aidan O'Dwyer
School of Control Systems and Electrical Engineering,
Dublin Institute of Technology,
Kevin St., Dublin 8, Ireland
E-mail: aidan.odwyer@dit.ie

Abstract -- **The ability of PI and PID controllers to compensate many practical processes has led to their wide acceptance in industrial applications. The requirement to choose two or three controller parameters is most easily done using tuning rules. Starting with a general discussion of industrial practice, the paper will discuss, in particular, recent work in tuning rule development for processes with time delay.**

I INTRODUCTION

Proportional-Integral (PI) and Proportional-Integral-Derivative (PID) controllers are at the heart of control engineering practice for six decades. The use of the PID controller is ubiquitous in industry; it has been stated, for example, that in process control applications, more than 95% of the controllers are of PI or PID type [1-6]. Neglected by the academic research community until recently, work by K.J. Åström, T. Hägglund and F.G. Shinskey, among others, has sparked a revival of interest in the use of this "workhorse" of controller implementation. One illustrative statistic is worth quoting: over the decade 1992-2001, three hundred and eighty five publications on the use of the PI or PID controller for the compensation of processes with time delays have been recorded by the author, more than three times the number of publications in the previous five decades [7].

However, despite this development work, surveys indicating the state of the art of control industrial practice report sobering results. For example, Ender [8] states that, in his testing of thousands of control loops in hundreds of plants, it has been found that more than 30% of installed PI/PID controllers are operating in manual mode and 65% of loops operating in automatic mode produce less variance in manual than in automatic (i.e. the automatic controllers are poorly tuned). Another interesting such comment comes from literature published from Protuner UK Ltd. [9] in which they state that PI/PID controllers are sometimes deliberately detuned by operating staff for steady state operation. They quote a typical control system audit, comprising 300 loops, in which 46 controllers were operated with default tuning parameters in the controller. Literature published by Universal Dynamic Technologies [10], who are the vendors of the BrainWave predictive adaptive controller, claims that "extensive industry testing" shows that 75% of all PID based loops are out of tune. The company also quotes a recent survey of paper processing mills, in which 60% of the 36 mills surveyed stated that less than half of their control loops were well tuned (the majority of the mills reported that they had between 2000 and 4000 regulatory control loops). EnTech Control Engineering Ltd. [11] claim that only 20% of all control loops surveyed in mill audits have been found to actually reduce process variability in automatic mode over the short term. Of the problem loops, increased process variability in automatic mode could be ascribed specifically to controller tuning problems in approximately 30% of cases. Many of the points made above are re-iterated by Yu [12] (pages 1-2). The situation has not improved more recently, with Van Overschee and De Moor [13] reporting that 80% of PID controllers are badly tuned; 30% of PID controllers operate in manual with another 30% of the controlled loops increasing the short term variability of the process to be controlled (typically due to too strong integral action). The authors state that 25% of all PID controller loops use default factory settings, implying that they have not been tuned at all.

Thus, there is strong evidence that PI and PID controllers remain poorly understood and, in particular, poorly tuned in many applications. It is clear that the many controller tuning rules proposed in the literature are not having an impact on industrial practice. One reason is that the tuning rules are not very accessible, being scattered throughout the control literature; in addition, the notation used is not unified. In a recently published book [14], PI and PID controller tuning rules for processes with time delay have been bought together and summarised, using a unified notation. The present paper extends this work by detailing new tuning rules.

The structure of the paper is as follows. Section 2 summarises briefly the range of PI and PID controller structures proposed in the literature, together with the process models used to define the controller tuning rules. Section 3 details some new tuning rules for setting up PI and PID controllers (and their variations), for a number of process models. Conclusions to the paper will be drawn in Section 4.

II CONTROLLER ARCHITECTURE AND PROCESS MODELLING

A practical difficulty with PID control technology is a lack of industrial standards, which has resulted in a wide variety of PID controller architectures. Five different structures for the PI controller and some 26 different structures for the PID controller have been identified. Controller manufacturers vary in their choice of architecture; controller tuning that works well on one architecture may work poorly on another. Full details are given by O'Dwyer [14]; considering the PID controller, common architectures are:

1. The 'ideal' PID controller:

$$G_c(s) = K_c\left(1 + \frac{1}{T_i s} + T_d s\right)$$

This architecture is used, for example, on the Honeywell TDC3000 Process Manager Type A, non-interactive mode product [15].

2. The 'classical' PID controller:

$$G_c(s) = K_c\left(1 + \frac{1}{T_i s}\right)\frac{1 + sT_d}{1 + s\dfrac{T_d}{N}}$$

This architecture is used, for example, on the Honeywell TDC3000 Process Manager Type A, interactive mode product [15].

3. The non-interacting controller based on the two degree of freedom structure:

$$U(s) = K_c\left(1 + \frac{1}{T_i s} + \frac{T_d s}{1 + \dfrac{T_d}{N}s}\right)E(s) - K_c\left(\alpha + \frac{\beta T_d s}{1 + \dfrac{T_d}{N}s}\right)R(s)$$

This architecture is used, for example, on the Omron E5CK digital controller with $\beta = 1$ and N=3 [15].

The most dominant PI controller architecture is the 'ideal' PI controller: $G_c(s) = K_c\left(1 + \frac{1}{T_i s}\right)$

The wide variety of controller architectures is mirrored by the wide variety of ways in which processes with time delay may be modeled. Common models are:

1. Stable first order lag plus time delay (FOLPD) model: $G_m(s) = \frac{K_m e^{-s\tau_m}}{1 + sT_m}$
2. Integral plus delay (IPD) model: $G_m(s) = \frac{K_m e^{-s\tau_m}}{s}$
3. First order lag plus integral plus delay (FOLIPD) model: $G_m(s) = \frac{K_m e^{-s\tau_m}}{s(1 + sT_m)}$
4. Second order system plus time delay (SOSPD) model:

$$G_m(s) = \frac{K_m e^{-s\tau_m}}{T_{m1}^2 s^2 + 2\xi_m T_{m1} s + 1} \text{ or}$$

$$G_m(s) = \frac{K_m e^{-s\tau_m}}{(1 + T_{m1}s)(1 + T_{m2}s)}$$

It has been shown that 89% of the PI controller tuning rules have been defined for the ideal PI controller structure, with 47% of tuning rules based on a FOLPD process model. The range of PID controller variations has lead to a less homogenous situation than for the PI controller; 44% of tuning rules have been defined for the ideal PID controller structure, with 37% of PID tuning rules based on a FOLPD process model [14].

Of course, the modeling strategy used will influence the value of the model parameters, which will in turn affect the controller values determined from the tuning rules. Twenty-six modeling strategies have been proposed to determine the parameters of the FOLPD process model, for example. Space does not permit a full discussion of this issue; further details are provided by O'Dwyer [14].

III NEW TUNING RULES FOR PI AND PID CONTROLLERS

Space considerations dictate that only some new tuning rules may be indicated; the details of all of the new tuning rules will be provided at the conference. Tuning rules are set out in tabular form (in Appendices 1a and 1b), allowing the rules to be represented compactly. The tables have four or five columns, according to whether the controller considered is of PI or PID form, respectively. The first column details the author of the rule, the method used to obtain the parameters used in the tuning rule formula (if any) and other pertinent information. The final column in all cases is labelled "Comment"; this facilitates the inclusion of information about the tuning rule that may be useful in its application. The remaining columns detail the formulae for the controller parameters.

Within each table, the tuning rules are classified further; the main subdivisions made are as follows:

(i) Tuning rules based on a measured step response (also called process reaction curve methods).

(ii) Tuning rules based on minimising an appropriate performance criterion, either for optimum regulator or optimum servo action.
(iii) Tuning rules that give a specified closed loop response (direct synthesis tuning rules). Such rules may be defined by specifying the desired poles of the closed loop response, for instance, though more generally, the desired closed loop transfer function may be specified. The definition may be expanded to cover techniques that allow the achievement of a specified gain margin and/or phase margin.
(iv) Robust tuning rules, with an explicit robust stability and robust performance criterion built in to the design process.
(v) Tuning rules based on recording appropriate parameters at the ultimate frequency (also called ultimate cycling methods).
(vi) Other tuning rules, such as tuning rules that depend on the proportional gain required to achieve a quarter decay ratio or to achieve magnitude and frequency information at a particular phase lag.

Some tuning rules could be considered to belong to more than one subdivision, so the subdivisions cannot be considered to be mutually exclusive; nevertheless, they provide a convenient way to classify the rules. In the tables, all symbols used are defined in Appendix 2.

Forty-four new PI controller tuning rules have been specified. The total number of PI controller tuning rules that have been identified, from the work reported in this paper and previous work [14], is 263. Seventy-three new PID controller tuning rules have been specified. The total number of PID controller tuning rules that have been identified, from the work reported in this paper and previous work [14], is 454.

IV CONCLUSIONS

Control academics and practitioners remain interested in the use of PI and PID controllers to compensate processes with time delay. This paper summarises recent work in tuning rule development for such processes, updating the information provided by O'Dwyer [14]. The most startling statistic to emerge from the work is the quantity of tuning rules proposed; 263 PI tuning rules and 454 PID tuning rules, a total of 717 separate rules. Recent years have seen an acceleration in the accumulation of tuning rules. In general, there is a lack of comparative analysis regarding the performance and robustness of closed loop systems compensated with controllers whose parameters are chosen using the tuning rules; associated with this is the lack of benchmark processes, at least until the recent suggestions of Åström and Hägglund [16], on which to base such analysis. The main priority for future research in the area should be a critical analysis of available tuning rules, rather than the proposal of further tuning rules.

REFERENCES

[1] K.J. Åström and T. Hägglund. *PID Controllers: Theory, Design and Tuning*. Instrument Society of America, Research Triangle Park, North Carolina, 2nd Edition, 1995.

[2] H.N. Koivo and J.T. Tanttu. "Tuning of PID Controllers: Survey of SISO and MIMO techniques". Proc. *IFAC Intelligent Tuning and Adaptive Control Symposium*, 75-80, Singapore, 1991.

[3] W.L. Bialkowski in *The Control Handbook*, ed. W.S. Levine. CRC/IEEE Press, Boca Raton, Florida, 1996, 1219-1242.

[4] W.L. Luyben and M.L. Luyben. *Essentials of process control*. McGraw-Hill International Edition, 1997.

[5] M.A. Hersh and M.A. Johnson. "A study of advanced control systems in the workplace". *Control Eng. Practice*, **5**(6):771-778, 1997.

[6] H. Takatsu and T. Itoh. "Future needs for control theory in industry – report of the control technology survey of Japanese industry". *IEEE Trans. Control Syst. Tech*., **7**(3):298-305, 1999.

[7] A. O'Dwyer. "PID compensation of time delayed processes: a full survey", *Technical Report AOD-00-05*, Version 3. School of Control Systems, Dublin Institute of Technology, Ireland (January 2004).

[8] D.B. Ender. *Control Engineering*. 180-190, September 1993.

[9] Protuner U.K. Ltd. "The state of art in control". Sales literature, 1997.

[10] Universal Dynamic Technologies. "Brainwave: the new concept in process control". Sales literature, 1998.

[11] Entech Control Engineering Ltd. *Competency in Process Control - Industry Guidelines*. Version 1.0, 3/94, 1994

[12] C.-C. Yu. *Autotuning of PID controllers*. Advances in Industrial Control Series, Springer-Verlag London Ltd., 1999.

[13] P. Van Overschee and B. De Moor. "RaPID: the end of heuristic PID tuning". *Preprints Proc. PID '00: IFAC Workshop on digital control*, 687-692, Terrassa, Spain, 2000.

[14] A. O'Dwyer. *Handbook of PI and PID controller tuning rules*. Imperial College Press, London, 2003.

[15] ISMC. *RAPID: Robust Advanced PID Control Manual*. Intelligent System Modeling and Control nv, Belgium, 1999.

[16] K.J. Åström and T. Hägglund. "Benchmark systems for PID control". *Preprints Proc. PID '00: IFAC Workshop on digital control*, 181-182, Terrassa, Spain, 2000.

APPENDIX 1a: NEW PI CONTROLLER TUNING RULES

Table 1: FOLPD model $G_m(s) = \dfrac{K_m e^{-s\tau_m}}{1+sT_m}$; Ideal PI controller $G_c(s) = K_c\left(1+\dfrac{1}{T_i s}\right)$

Rule	K_c	T_i	Comment
Direct synthesis	**Time domain criteria**		
Skogestad [17]	$\dfrac{0.5T_m}{K_m\tau_m}$	$\min(T_m, 8\tau_m)$	Model parameters derived assuming higher order process parameters known
Gorez [18] *Model parameters assumed known*	[1] $\dfrac{T_m}{\chi K_m\tau_m}$	T_m	$0 \le \dfrac{\tau_m}{\tau_m+T_m} \le \tau_m$
	$K_c^{(1)}$ [2]	$(1-v)\tau_m + T_m$	$\tau_m \le \dfrac{\tau_m}{\tau_m+T_m} \le 1$

Table 2: IPD model $G_m(s) = \dfrac{K_m e^{-s\tau_m}}{s}$; Ideal PI controller $G_c(s) = K_c\left(1+\dfrac{1}{T_i s}\right)$

Rule	K_c	T_i	Comment
Direct synthesis	**Time domain criteria**		
Chidambaram and Sree [19]	$\dfrac{1.1111}{K_m\tau_m}$	$4.5\tau_m$	Model parameters assumed known
Skogestad [17]	$\dfrac{0.5}{K_m\tau_m}$	$8\tau_m$	Model parameters assumed known

APPENDIX 1b: NEW PID CONTROLLER TUNING RULES

Table 3: FOLPD model $G_m(s) = \dfrac{K_m e^{-s\tau_m}}{1+sT_m}$; ideal PID controller $G_c(s) = K_c\left(1+\dfrac{1}{T_i s}+T_d s\right)$

Rule	K_c	T_i	T_d	Comment
	Other tuning: performance index minimisation			
Servo or regulator tuning – minimum IAE – Huang and Jeng [20] *Model parameters identified using a relay feedback method*	$\dfrac{0.36+0.76\dfrac{T_m}{\tau_m}}{K_m}$	$0.47\tau_m + T_m$	$\dfrac{0.47T_m\tau_m}{0.47\tau_m+T_m}$	$\dfrac{\tau_m}{T_m} < 0.33$
	$K_c^{(2)}$ [3]	$T_i^{(2)}$	$T_d^{(2)}$	$\dfrac{\tau_m}{T_m} \ge 0.33$

---

[1] $$\chi = \frac{1}{M_s(M_s-1)} + \frac{1.5M_s^2-2}{0.32M_s^2(M_s-1)}\left(\frac{\tau_m}{\tau_m+T_m}\right)^2\left(3-\frac{5\tau_m}{(\tau_m+T_m)\sqrt{M_s}}\right) - \frac{0.5}{M_s(M_s-1)}\left(1-\frac{2.5\tau_m}{(\tau_m+T_m)\sqrt{M_s}}\right)^{2(M_s^2-1)}$$

[2] $$K_c^{(1)} = \frac{(1-v)\tau_m+T_m}{\chi K_m\tau_m} \text{ with } v = 1-0.5\left(\frac{\dfrac{\tau_m}{\tau_m+T_m}-0.4\sqrt{M_s}}{1-0.4\sqrt{M_s}}\right)^2,\ \chi = \frac{0.1M_s}{M_s-1}\left[7.5-\left(\frac{\dfrac{\tau_m}{\tau_m+T_m}-0.4\sqrt{M_s}}{1-0.4\sqrt{M_s}}\right)^2\right]$$

[3] $$K_c^{(2)} = \frac{0.8194\,T_m + 0.2773\,\tau_m}{K_m\tau_m^{0.9738}T_m^{0.0262}},\ T_i^{(2)} = 1.0297T_m + 0.3484\tau_m,\ T_d^{(2)} = \left(\frac{0.4575\,T_m+0.0302\,\tau_m}{1.0297\,T_m+0.3484\,\tau_m}\right)\tau_m$$

Table 4: IPD model $G_m(s)=\frac{K_m e^{-s\tau_m}}{s}$; ideal PID controller $G_c(s)=K_c\left(1+\frac{1}{T_i s}+T_d s\right)$

Rule	K_c	T_i	T_d	Comment
	Direct synthesis			
Chidambaram and Sree [19]	$\frac{1.2346}{K_m \tau_m}$	$4.5\tau_m$	$0.45\tau_m$	Model parameters assumed known
	Robust			
Bequette [21] – *page 300.*	$\frac{2}{K_m(\lambda+0.5\tau_m)}$	$2\lambda+\tau_m$	$\frac{\lambda+0.25\tau_m}{2\lambda+\tau_m}\tau_m$	Model method not specified

Table 5: IPD model $G_m(s)=\frac{K_m e^{-s\tau_m}}{s}$; non-interacting controller based on the two degree of freedom structure $U(s)=K_c\left(1+\frac{1}{T_i s}+\frac{T_d s}{1+\frac{T_d}{N}s}\right)E(s)-K_c\left(\alpha+\frac{\beta T_d s}{1+\frac{T_d}{N}s}\right)R(s)$

Rule	K_c	T_i	T_d	Comment
	Direct synthesis			
Chidambaram and Sree [19] *Model parameters assumed known*	$\frac{1.2346}{K_m \tau_m}$	$4.5\tau_m$	$0.45\tau_m$	N = 0; $\beta=0$; α = 0.6

Table 6: IPD model $G_m(s)=\frac{K_m e^{-s\tau_m}}{s}$ – controller $U(s)=\frac{K_c}{T_i s}E(s)-K_c(1+T_d s)Y(s)$.

Rule	K_c	T_i	T_d	Comment
	Regulator tuning: Minimum performance index			
Minimum ISE– Arvanitis *et al*. [22]	$\frac{1.4394}{K_m \tau_m}$	$2.4569\,\tau_m$	$0.3982\,\tau_m$	Model parameters assumed known
[4]Minimum performance index – Arvanitis *et al*. [22]	$\frac{1.2986}{K_m \tau_m}$	$3.2616\,\tau_m$	$0.4234\,\tau_m$	Model parameters assumed known
[5]Minimum performance index – Arvanitis *et al*. [22]	$\frac{1.1259}{K_m \tau_m}$	$6.7092\,\tau_m$	$0.4627\,\tau_m$	Model parameters assumed known

REFERENCES FOR APPENDIX 1a AND 1b

[17] S. Skogestad. "Simple analytic rules for model reduction and PID controller tuning". *Journal of Process Control*, 13:291-309, 2003.

[18] R. Gorez. "New design relations for 2-DOF PID-like control systems". *Automatica*, 39:901-908, 2003.

[19] M. Chidambaram and R.P. Sree. "A simple method of tuning PID controllers for integrator/dead-time processes". *Computers and Chemical Eng*., 27:211-215, 2003.

[20] H.-P. Huang and J.-C. Jeng. "Identification for monitoring and autotuning of PID controllers". *Journal of Chemical Eng. of Japan*, 36(3):284-296, 2003.

[4] Performance index = $\int_0^\infty \left[e^2(t)+K_m{}^2u^2(t)\right]dt$

[5] Performance index = $\int_0^\infty \left[e^2(t)+K_m{}^2\left(\frac{du}{dt}\right)^2\right]dt$

[21] B.W. Bequette. *Process Control: Modeling, Design and Simulation*. Prentice-Hall International Series in the Physical and Chemical Sciences, Upper Saddle River, New Jersey 07458, USA, 2003.
[22] K.G. Arvanitis, G. Syrkos, I.Z. Stellas and N.A. Sigrimis. "Controller tuning for integrating processes with time delay. Part I: IPDT processes and the pseudo-derivative feedback control configuration". *Proc. 11th Mediterranean Conf. On Control and Automation*, Rodos, Greece, Paper T7-040, 2003.

APPENDIX 2: GLOSSARY OF SYMBOLS USED

du/dt = time derivative of the manipulated variable (time domain)

e(t) = desired variable, r(t), minus controlled variable, y(t) (time domain).

E(s) = Desired variable, R(s), minus controlled variable, Y(s)

FOLPD model = First Order Lag Plus time Delay model

FOLIPD model = First Order Lag plus Integral Plus time Delay model

$G_c(s)$ = PID controller transfer function

ISE = integral of squared error = $\int_0^\infty e^2(t)dt$

K_c = Proportional gain of the controller

K_m = Gain of the process model

M_s = closed loop sensitivity

N = parameter that determines the amount of filtering on the derivative term on some PID controller structures

PI controller = proportional integral controller

PID controller = proportional integral derivative controller

R(s) = Desired variable (Laplace domain)

s = Laplace variable

SOSPD model = Second Order System Plus time Delay model

T_d = Derivative time of the controller

T_i = Integral time of the controller

T_m = Time constant of the FOLPD process model

T_{m1}, T_{m2} = Time constants of second order process model

u(t) = manipulated variable (time domain).

U(s) = manipulated variable (Laplace domain)

α, β = weighting factors in some PI or PID controller structures

λ = Parameter that determines robustness of compensated system.

ξ_m = damping factor of an underdamped process model

τ_m = time delay of the process model