Advances in Industrial Control

Víctor M. Alfaro
Ramon Vilanova

# Model-Reference Robust Tuning of PID Controllers

AIC
Springer

# Advances in Industrial Control

**Series editors**

Michael J. Grimble, Glasgow, UK
Michael A. Johnson, Kidlington, UK

More information about this series at http://www.springer.com/series/1412

Víctor M. Alfaro · Ramon Vilanova

# Model-Reference Robust Tuning of PID Controllers

Springer

Víctor M. Alfaro
School of Electrical Engineering
Universidad de Costa Rica
San José
Costa Rica

Ramon Vilanova
School of Engineering
Universitat Autònoma de Barcelona
Barcelona
Spain

MATLAB® and Simulink® are registered trademarks of The MathWorks, Inc., 3 Apple Hill Drive, Natick, MA 01760-2098, USA, http://www.mathworks.com

ISSN 1430-9491 ISSN 2193-1577 (electronic)
Advances in Industrial Control
ISBN 978-3-319-28211-4 ISBN 978-3-319-28213-8 (eBook)
DOI 10.1007/978-3-319-28213-8

Library of Congress Control Number: 2016935592



Printed on acid-free paper

This Springer imprint is published by Springer Nature
The registered company is Springer International Publishing AG Switzerland

*V. M. Alfaro dedicates the book to his wife María Amanda and to his children Alejandro, Carolina, and Sebastián.*

*R. Vilanova thanks his beloved wife Rosa for her love and support during the time spent in writing the book.*

# Series Editors' Foreword

The series *Advances in Industrial Control* aims to report and encourage technology transfer in control engineering. The rapid development of control technology has an impact on all areas of the control discipline. New theory, new controllers, actuators, sensors, new industrial processes, computer methods, new applications, new philosophies,..., new challenges. Much of this development work resides in industrial reports, feasibility study papers and the reports of advanced collaborative projects. The series offers an opportunity for researchers to present an extended exposition of such new work in all aspects of industrial control for wider and rapid dissemination.

In an interesting recent journal paper [1], some survey evidence was presented that supported the premise that PID control continues to play a significant role as an industrial controller in a wide range of industries. For low-level loops the PID controller is simple to apply and sufficiently effective to justify its continued widespread popularity. The *Advances in Industrial Control* monograph series has always aimed to feature the most recent developments in this field. In 2012, the series published a wide ranging survey text *PID* Control in the *Third Millennium* edited by Ramon Vilanova and Antonio Visioli (ISBN 978-1-4471-2424-5, 2012). This volume demonstrated the breadth of ideas and applications appearing in the PID control field.

However, for the purposes of this Foreword it is sufficient to map the practical tools of the evolving science of PID control onto the two categories of "PID controller tuning" and "PID controller performance monitoring".

*PID Controller Tuning*

Controller design begins with the desired performance specification. The metrics to be achieved by the control have grown in number over recent years to take in concepts such as robustness and fragility. A subset of the design specifications are usually used by the design algorithm that can be an offline or online procedure. The particular field of automated online PID controller design is one of considerable interest to the industrial control engineer. Ultimately the PID controller design will meet only some of the possible controller specifications and then the remaining metrics can be used as controller evaluation metrics. There are a number of monographs in the *Advances in Industrial Control* series on these topics; the most recent being:

- *Control of Integral Processes* with Dead Time by Antonio Visioli and Qing-Chang Zhong (ISBN 978-0-85729-069-4, 2011);
- *Non-Parametric Tuning of PID Controllers* by Igor Boiko (ISBN 978-1-4471-4464-9, 2012); and
- *Industrial Process Identification and Control Design* by Tao Liu and Furong Gao (ISBN 978-0-85729-976-5, 2012).

*PID Controller Performance Monitoring*

Once a controller has been implemented, the question arises as to how to verify that it is retaining its desired performance over operational time. The idea of monitoring controller performance is a generic one and not just restricted to PID controllers. However, the possible situation of a large number of PID loops in a process plant has led to significant monitoring algorithm developments for the PID control field. An online automated performance monitoring routine is a very attractive economic proposition for industrial PID applications. The survey paper [1], cited above provides an excellent overview of the current state of the art of such methods. On this topic, the *Advances in Industrial Control* monograph series has titles that include:

- *Process Control Performance Assessment* edited by Andrzej W. Ordys, Damien Uduehi and Michael A. Johnson (ISBN 978-1-84628-623-0, 2007); and
- *Control Performance Management in Industrial Automation* by Mohieddine Jelali (ISBN 978-1-4471-4545-5, 2012).

This monograph, *Model-Reference Robust Tuning of PID Controllers*, by Victor M. Alfaro and Ramon Vilanova is clearly a contribution to the PID-controller-tuning literature. When dealing with controller design specifications and controller evaluation metrics, the authors discuss both time-domain and frequency-domain metrics and specifications. Subsumed into this assessment framework are both controller robustness and controller fragility. Whereas controller robustness measures the effect on performance and stability of changing process parameters, controller fragility inverts this to measure the effect of the variability of the controller parameters for a fixed given design process on performance and stability. Measures of controller fragility find application in controller commissioning where online tuning may move controller parameters away from design values with a concomitant change in closed-loop performance and stability. This aspect is discussed by the authors in Chap. 3, and is just one of the interesting topics found in this monograph.

The monograph reports the joint research of the authors. Professor Victor Alfaro is with the Department of Automation at the Universidad de Costa Rica. He has had a long and distinguished career in various industries as a practicing engineer before joining the academic community. Professor Ramon Vilanova is with the Department of Telecommunications and Systems Engineering at the Universitat Autònoma de Barcelona. He has made many contributions to the PID control literature. His industrial work has included important control research for wastewater treatment plants.

The Editors of the *Advances in Industrial Control* monograph series are very pleased to add this title to the important set of monographs in the series on all aspects of PID control.

Industrial Control Centre, Glasgow, Scotland, UK M.J. Grimble
M.A. Johnson

## Reference

1. Bauer, M., Horch, A., Xieb, L., Jelali, M., Thornhill, N.: The current state of control loop performance monitoring—A survey of application in industry, J. Process Control **38**, 1–10 (2016)

# Preface

Controllers and controller design are at the heart of industrial progress. Controllers allow to keep process variables of interest at prescribed values in order to guarantee product quality as well as better production times. The controller receives information of the actual value of the process variable of interest, the controlled variable, and of the desired value for this, the controller set-point. It compares these two values to obtain the actuating error signal. Commercial controllers with a proportional integral derivative control algorithm, PID for short, were introduced back in 1940. As has been widely reported elsewhere, 75 years later it still is a more common control algorithm used in the processes industry.

The influence on the controller output signal of each one of the control modes can be adjusted setting its corresponding adjustable parameter. Although the PID control algorithm provides to the user the opportunity of combining the information of the error signal (P), its integral over time (I), and its rate of change (D), most of the controllers in operation use only the error signal and of its integral, in a proportional integral control, or simply PI control. Among this underuse of the PID capabilities, it is a well-known fact that poor controller tuning is a common situation, bearing in mind that there are many tuning rules to allow the specification of the controller parameters. One may think that, in fact, there exists an overwhelming quantity that makes it difficult to decide and apply. In addition to this great variety of tuning approaches, even though the PID controllers are of fixed structure over the years some additional capabilities have been incorporated into them: measurement and set-point signal filters, set-point weighting factors, reset windup prevention, and other features. At the controlled process side, there is a wide range of dynamic characteristics: over- and underdamped, integrating, unstable, slow, fast, nonlinear, and their possible combinations.

The control system designer faces the task of selecting the controller control algorithm parameters to restrict the controlled process variable to perform according to certain design criteria. These criteria would include, among others, evaluation of the control system performance and relative stability. The possible combinations of control algorithms, controlled process dynamics and information, design criteria, and design approaches result in the diversity of tuning relations that have been growing over the years and reveal plenty of technical publications available in this subject.

From the existing literature on PID controller design, it is easy to see that different design approaches exist depending on the controlled process dynamics and even on the way the desired performance is stated. This clearly contributes to the confusion that prevents practitioners from the application of tuning rules. As said above, the purpose of the book is to provide a comprehensive and didactical presentation of a unifying approach for controller design (in fact when applied to PID controllers it may fit into any fixed structure controller) that deals in an explicit way with the performance/robustness trade-off as one of the key points in modern PID tuning.

The proposed controller design procedure is based on the use of closed-loop transfer functions targets (the *reference models*) to obtain robust control systems, therefore is named *Model-Reference Robust Tuning* (MoReRT). As design main considerations are, in addition to the closed-loop responses shapes, the control system relative stability, its robustness to process variations, and to obtain a smooth control effort.

This book is based on the research work the authors carried out during the past few years. It is not intended to be a research report but a unified presentation of the previously referred MoReRT methodology for PI/PID controller design. As found in references, a somewhat complete set of journals can be accessed where a deeper discussion on some control topics can be found. Also the comparison of the proposed design approach with some approaches previously existing in the literature has been excluded from the book content. These comparisons can be found on the referred journal papers as the main goal of the book is to serve as a comprehensive presentation of a design approach that, in the authors opinion, deserves some extensions and particular applications, which would be difficult to forecast just by looking at the set of *disconnected* results that journal papers usually constitute.

The book comprises a total of 11 chapters and one appendix. The contents can be structured along four parts.

The first part comprises Chaps. 1–3. These chapters provide a generic description of the control system under study as well as some particular insights into PID controllers formulations and metrics to evaluate its performance. These topics could be general to any other approach to PID controller design. Specifically, the feedback control design problem and the evolution over time of the considerations taken for PID tuning are briefly presented in Chap. 1. The two-degree-of-freedom proportional integral derivative (2DoF PID) control algorithm structures and their conversion relations are presented in Chap. 2. Parameter conversion formulas take into consideration the derivative filter constant. This chapter is innovative enough to be of interest in its own because in the PID controller literature the equivalence and conditions that make such equivalence possible are not found. Particular results for PID control are usually presented by adopting one of the multiple PID formulations. In Chap. 3 the indices used for performance, robustness, and control system fragility evaluation are presented. Control system robustness is evaluated using the maximum of the sensitivity function (maximum sensitivity).

The second part of the book contains Chaps. 4 and 5, where the methodological formulations of the MoReRT are presented. Chapter 4 describes the basis of the proposed model-reference robust tuning (MoReRT) design methodology and how the model-reference closed-loop transfer functions are selected, the cost functional stated for optimization, and the available free design parameters. This proposal is to be applied to a variety of process dynamics in order to derive the corresponding tuning rules. However, application will consider normalized models for controlled process and controller. This will ensure to satisfy the so-called time-scaling property. Therefore, before proceeding to the derivation of the robust tuning rules, normalized controlled process models and controllers equations used in the design are presented in Chap. 5. The use of normalized controlled process models and normalized control algorithms allows to obtain dimensionless controller tuning rules.

The third part of the book contains the development of tuning rules for all the considered process dynamics. It can therefore be considered the core part of the book. The MoReRT control of overdamped controlled processes is presented in Chap. 6, where controllers with 2DoF PI and PID control algorithms are used for robust control of first- and second-order controlled process models. A comparison of the achievable performance, under the same robustness, is done by using a PI or a PID controller and also by the fact that designing a controller is by using a first- or a second-order process model. The robust control of inverse response processes is described in Chap. 7. Here it is stated that the right-half plane zero position impose constrains to the achievable control system robustness. In Chap. 8 MoReRT control of first- and second-order integrating processes is presented. For first-order integrating models, the MoReRT design results in a very simple tuning for the normalized controller parameters. In Chap. 9 the MoReRT design is used to tune 2DoF PI and PID controllers for unstable processes. The unstable pole position imposes severe constraints on the achievable control system robustness. One of the detectable points in all the developments shows how MoReRT allows to face the PI/PID design problem from the same point of view.

The fourth part of the book describes potential extensions of the method as well as considerations for its practical applications. Three possible extensions of the MoReRT methodology are presented in Chap. 10. First of all, it is used in the case where the purpose is to design the profile of the manipulated variable instead of, as usual, the controlled variable. Second, a more general MoReRT design is applied in the case where the dynamics of the disturbance to the controlled variable is different from the dynamics of the manipulated variable to the controlled variable. This makes the design problem a little bit more complex, and the fact of facing multiple dynamics makes it not possible to derive general tuning rules. Instead, it is shown how MoReRT can be formulated and applied. Third, the use of the MoReRT design in robust tuning of a Smith predictor type dead-time compensating PI controller, including the predictor model parameters, is presented. As a source of practical considerations, the book ends with the description of the 2DoF PID control algorithms available in some commercial controllers, programmable logic controllers, and digital control systems. A condensed reference of the MoReRT tuning relations is also presented with its applicability ranges and constraints. A general outline for the implementation of a MoReRT design procedure and the application of the proposed tuning method to control a typical industrial process are provided in Chap. 11.

The book ends with an appendix describing a software package developed for MATLAB® in order to facilitate the implementation of the MoReRT approach. The provided routines just require the user to input the process information data and the desired controller structure. The software will perform the required optimizations and show the closed-loop responses for the obtained controller.

# Acknowledgments

The authors would like to acknowledge all the people who have contributed to this book in one way or the other, in particular, S. Alcantara, O. Arrieta, S. Dormido, C. Pedret, A. Visioli, just to name a few. Special thanks to the Series Editors M. Grimble and M. Johnson, the Editor O. Jackson and the Editorial Assistant Karin de Bie for their help during the preparation of the manuscript. Partial support of the research that originated the results presented in this book has been provided by the Spanish Ministry of Economy and Competitiveness through grants DPI2010-15230, and DPI2013-47825-C3-1-R, and by the University of Costa Rica.

# Contents

# Chapter 1
# Introduction

Generally in control systems and particularly in process control applications the *feedback control* scheme or *closed-loop control system* depicted in Fig. 1.1 is the control structure that solves most of the control problems faced.

For a given process $P'$ there is a characteristic or variable that needs to be "controlled." The information about this controlled variable is obtained with a measurement instrument, the sensor/transmitter $T$. The transmitter output signal $Y$ is sent to the controller $C$ that also receives the desired value or set-point $R$ for the controlled variable. The controller control algorithm processes these two inputs and the computed control effort or controller output $U$ is sent to the actuator or final control element $A$ in order to modify a process internal quantity, a manipulated variable, or to affect the variable of interest. The disturbances $D$ are all other process variables that affect the controlled variable in an undesired fashion.

From the *controller* point of view the actuator/process/transmitter group represents the *controlled process* $P$. The controller and the controlled process share information through the control effort ($U$) and the controlled variable ($Y$) signals. Then, for an external viewer the feedback control system has two inputs: the *set-point* ($R$) and the *disturbances* ($D$), and one output, the *controlled variable* ($Y$) as depicted in Fig. 1.2.

The controller is designed to *restrict* the controlled variable response to a change in the input signals according to the design specifications. As there are two input signals of very different kind and entering at different points of the control system, the problem of dealing with each one of them (either the disturbance $D$ that should be attenuated or the reference $R$ that must be tracked) is not trivial. The control design problem is then to adjust ("tune") the parameters of the selected controller control algorithm in order to achieve the desired controlled variable performance.

A natural way to adjust or correct the behavior over time of a controlled process output, the controlled variable, is by using an actuating input computed on the basis of the comparison of the process actual output and the measured controlled variable with its desired or set-point value; this is based in the closed-loop system feedback error. To compute the control action information about the feedback error evolution is required. Normally its current value, its past evolution, and a prediction of its future



V.M. Alfaro and R. Vilanova, *Model-Reference Robust Tuning of PID Controllers*,
Advances in Industrial Control, DOI 10.1007/978-3-319-28213-8_1

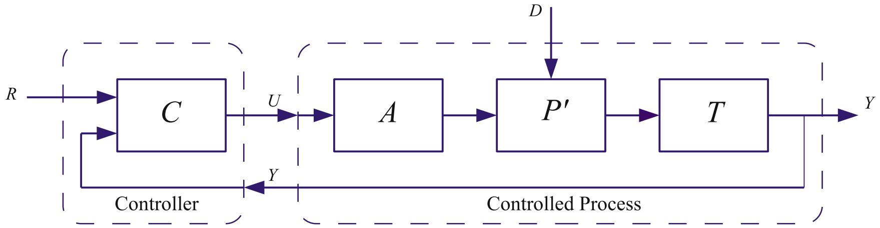
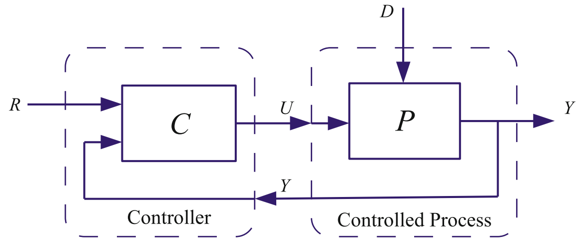



**Fig. 1.1** General feedback control system structure

**Fig. 1.2** Simple feedback control system




behavior are used. The way we use this information to deliver the control action constitutes the controller control algorithm.

The feedback control structure has been used for a long time, but if we restrict ourselves to the industrial process control area, the proportional (present error information), the integral (past error accumulated) and the derivative (future error prediction) or the PID control algorithm age starts in 1940 with the introduction of the Taylor Fullscope 100 pneumatic PID controller [1]. It was the first controller with knobs and calibrated dials for all three responses [2]. The original simple three-term PID control algorithm has evolved to the actual four- or five-term two-degree-of-freedom (2DoF) PID implementations [3].

To guarantee a stable and successful operation of the control system the controller must be matched with the controlled process, using information of the dynamic characteristics of the process usually represented by a low-order linear *model*. This matching essentially captures the process information and translates into a suitable selection for the controller parameters either by application of a direct tuning procedure (usually based on optimization or analytical derivations) or by means of a tuning rule. Tuning rules have the advantage of ease of calculation of the controller parameters (when compared to more analytical controller design methods), on the one hand; on the other hand, the use of tuning rules is a good alternative to trial and error tuning.

At the beginning, controller tuning took into consideration only the control system *performance* [4, 5], the output signal dynamic characteristics, to step changes in its inputs. Most of the developed research works that have emerged over the years, take the form of design proposals based on simple models and generally give rise to tuning rules that link the parameters of the process model, with the controller ones in a direct and simple way. The need for such simple and model-based tuning rules is also encouraged from several control engineering books; some of them specific on PID control. A common point that can be found in all of them is the need to incorporate a good understanding of the control problem and its relationship with modeling and knowledge of the process to be controlled. It was noticed that if only the performance is considered in the design it leads to control systems with very low robustness [6], this is to say low capability to deal with changes in the controlled process dynamic characteristics. Then, *robustness* was introduced into the controller design [7]. The performance/robustness trade-off in PID control system design is a well-known issue [8]. Even in case that this trade-off be resolved at the design stage [9–12], it is important to evaluate the controller *fragility*: the effect of a change in the controller parameters, at its final fine-tuning [13].

In most of the industrial process control applications, the desired value of the controlled variable, or set-point, normally remains constant and a good load disturbance rejection is required [14], which is usually known as regulatory control. However, due to variations in the process operating conditions, the controlled variable set-point may eventually need to be changed and then a good transient response to such change is required, which is known as servo-control operation. Satisfying these two operating conditions simultaneously is not possible by using a one-degree-of-freedom (1DoF) PI/PID controller, but using a two-degree-of-freedom (2DoF) PI/PID allows tuning of the controller in order to do so. The extra parameter it provides is used to improve its servo-control behavior while considering the regulatory control performance and the closed-loop control system robustness. This second degree of freedom; introduced by Araki [15–17]; is aimed at providing additional flexibility to control system design with PI/PID controllers [18, 19].

On the other hand, we have a variety of controlled processes dynamics, from the most common self-regulating overdamped to integrating and unstable processes.

Nowadays, the proportional integral and proportional integral derivative are the most used control algorithms in industry. Although, it is reported elsewhere that there are many loops with very poor performance, badly tuned or not tuned at all. Considering the huge number of PI and PID controllers in service at present in the process industry any improvement in their performance will produce a big overall revenue. Since the introduction of the seminal tuning rules of Ziegler and Nichols [20] a great number of tuning rules have been developed as revealed in [21]. Most of them take into account only one design criteria (performance, robustness) and are oriented to a specific controlled process model structure (overdamped, integrated, or other).

A different path is followed here. A general design procedure for 2DoF PI and PID controllers is proposed based on the specification of the corresponding control system closed-loop transfer functions that include parameters that affect the *performance/robustness trade-off*. The control system robustness requirement is controlled process dependent, and more importantly it cannot be avoided the robustness level, measured with the maximum sensitivity, is used as the design target. Besides this, the design methodology is the same for all considered processes and controllers. The specification of the closed-loop transfer functions also takes into account obtaining smooth *control effort*, controller output, signals.

One of the objectives of the work has been to develop a design methodology for robust control systems independent of the controller, PI or PID, and on the controlled process avoiding appealing to ad hoc design procedures for each particular case (controller/process combination). The controlled process model specific characteristics are incorporated only into the closed-loop target response specifications. The proposed controller design methodology denoted as *Model-Reference Robust Tuning* (MoReRT) is applied to tune 2DoF PI and PID controllers for first- and second-order overdamped, integrating, inverse response, and unstable controlled processes, being the accomplishment of the robustness target level for all the controlled process models considered (overdamped, integrated, and unstable) one of the distinctive characteristics of the proposed design method.

## References


1. Bennett, S.: The past of PID controllers. In: IFAC Digital Control: Past, Present and Future of PID Control, April 5–7, Terrassa, Spain (2000)
2. Blinkley, G.J.: Modern control started with Ziegler-Nichols tuning. Control Eng. **37**(12) (1990)
3. Åström, K., Hägglund, T.: Advanced PID Control. ISA—The Instrumentation, Systems, and Automation Society, Research Triangle Park, NC 27709, USA (2006)
4. López, A.M., Miller, J.A., Smith, C.L., Murrill, P.W.: Tuning controllers with error-integral criteria. Instrum. Technol. **14**, 57–62 (1967)
5. Rovira, A., Murrill, P.W., Smith, C.L.: Tuning controllers for setpoint changes. Instrum. Control Syst. **42**, 67–69 (1969)
6. Ingimundarson, A., Hägglund, T., Åström, K.J.: Criteria for desing of PID controllers. In: 2nd IFAC Conference Control Systems Design (CSD'03). Bratislava, Slovak Republic (2003)
7. Åström, K.J., Hägglund, T.: Automatic tuning of simple regulators with specification on phase and amplitude margins. Automatica **20**(5), 645–651 (1984)
8. Alfaro, V.M., Vilanova, R., Méndez, V., Lafuente, J.: Performance/robustness analysis of the PI/PID servo and regulatory control systems. In: IEEE International Conference on Industrial Technology (ICIT 2010). March 14–17, Viña del Mar, Chile (2010)
9. Alfaro, V.M., Méndez, V., Vilanova, R.: Robust-performance tuning of 2DoF PI/PID controllers for first- and second-order-plus-dead-time models. In: 9th IEEE International Conference on Control and Automation (ICCA), pp. 237–242. Dec 19–21, Santiago, Chile (2011)
10. Alfaro, V.M., Vilanova, R.: Optimal robust tuning for 1DoF PI/PID control unifying FOPDT/SOPDT models. In: IFAC Conference on Advances in PID Control (PID'12). 28–30 March, Brescia, Italy (2012)
11. Alfaro, V.M., Vilanova, R.: Set-point weight selection for robustly tuned PI/PID regulators for over damped processes. In: 17th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA 2012). Sept 17–21, Kraków, Poland (2012)
12. Alfaro, V.M., Vilanova, R.: Simple robust tuning of 2DoF PID controllers from a performance/robustness trade-off analysis. Asian J. Control **5**(5), 1–14 (2013)
13. Alfaro, V.M., Vilanova, R.: PID control in the third millenium—lessons learned and new approaches, chap. Fragility Evaluation of PI and PID Controllers Tuning Rules, pp. 349–380. Springer-Verlag London Limited (2012)
14. Shinskey, F.G.: Process control: as taught vs as practiced. Ind. Eng. Chem. Res. **41**, 3745–3750 (2002)

15. Araki, M.: On two-degree-of-freedom PID control system. Tech. rep., SICE research committee on modeling and control design of real systems (1984)
16. Araki, M.: Two-degree-of-freedom control system—I. Syst. Control **29**, 649–656 (1985)
17. Araki, M., Taguchi, H.: Two-degree-of-freedom PID controllers. Int. J. Control Autom. Syst. **1**(4), 401–411 (2003)
18. Taguchi, H., Araki, M.: Two-degree-of-freedom PID controllers—their functions and optimal tuning. In: IFAC Digital Control: Past, Present and Future of PID Control, April 5–7, Terrassa, Spain (2000)
19. Taguchi, H., Araki, M.: Survey of researches on two-degree-of-freedom PID controllers. In: The 4th Asian Control Conference, Sept 25–27, Singapore (2002)
20. Ziegler, J.G., Nichols, N.B.: Optimum settings for automatic controllers. ASME Trans. **64**, 759–768 (1942)
21. O'Dwyer, A.: Handbook of PI and PID Controller Tuning Rules, 3rd edn. Imperial College Press, London (2009)

# Chapter 2
# Two-Degree-of-Freedom PID Controllers Structures

As in most of the existing industrial process control applications, the desired value of the controlled variable, or set-point, normally remains constant (regulatory control or disturbance rejection operation) but needs to be changed (servo-control or set-point tracking operation) we are mainly interested in the two-degree-of-freedom (2DoF) implementation of the PID control algorithms. The extra parameter that the 2DoF control algorithm provides is used to improve their servo-control behavior while considering the regulatory control performance and the closed-loop control system robustness [1–5]. This 2DoF feature can be incorporated both into a PI or a PID control algorithm. Although all the controllers with a proportional integral (PI) control algorithm are implemented in the same way, have the same transfer function, this is not the case with commercial controllers with proportional integral derivative (PID) control algorithms.

In fact, usually, the control algorithm implementation is manufacturer dependent and not all of its variations are available in the same controller. Even more, the controllers manufacturers use different names for the same PID algorithm [6]. The diversity of the PID control algorithms is evident in [7]. In addition, it would be the case that a tuning rule of interest had been obtained using a control algorithm different from the one implemented in the controller to tune. In this case, controller parameters conversion is required that will also indicate if the pursued equivalent controller exists.

On that basis, the most widely used PID control algorithms are presented in this chapter by also providing conversion formulae that allows to convert the parameters of one algorithm from those obtained for another formulation. As it will be seen this conversion will not always be possible, showing some formulations are more general than others.

 
V.M. Alfaro and R. Vilanova, *Model-Reference Robust Tuning of PID Controllers*, Advances in Industrial Control, DOI 10.1007/978-3-319-28213-8_2

## 2.1 Proportional Integral Derivative Control Algorithm

Consider the general controller block diagram depicted in Fig. 2.1. The output or *control effort* of a proportional (P) integral (I) and derivative (D) control algorithm is given, in general, by

$$U(t) = Action\,\{U_P(t) + U_I(t) + U_D(t) + U_b\}, \tag{2.1}$$

if $0\,\% \le U(t) \le 100\,\%$, and 0 or $100\,\%$, depending on the controller action if the controller output reaches one of its limits.

In (2.1) $U_P$ is the proportional term or *proportional control action*, given by

$$U_P(t) = K_p E(t) = K_p[R(t) - Y(t)], \tag{2.2}$$

with a proportional gain $K_p$; $U_I$ is the integral term or *integral control action*, given by

$$U_I(t) = K_i \int_0^t E(\xi)\mathrm{d}\xi = K_i \int_0^t [R(\xi) - Y(\xi)]\mathrm{d}\xi, \tag{2.3}$$

with an integral gain $K_i$; and $U_D$ the derivative term or *derivative control action*, given by

$$U_D(t) = K_d \frac{\mathrm{d}E(t)}{\mathrm{d}t} = K_d \frac{\mathrm{d}[R(t) - Y(t)]}{\mathrm{d}t}, \tag{2.4}$$

with a derivative gain $K_d$.The controller output *bias* $U_b$ is usually set to $50\,\%$. In (2.1)–(2.4) controller inputs $R(t)$ and $Y(t)$, and output $U(t)$ change in the range from 0 to $100\,\%$.

The controller *Action* sign, $+1$ (Reverse) or $-1$ (Direct), must be selected equal to the controlled process gain sign to preserve the negative feedback characteristic of the control loop.

In the following, we will assume that the controller Action has been selected correctly, that all the closed-loop control variables are within their corresponding 0–$100\,\%$ range, and that the control system is initially at a steady-state stable operating

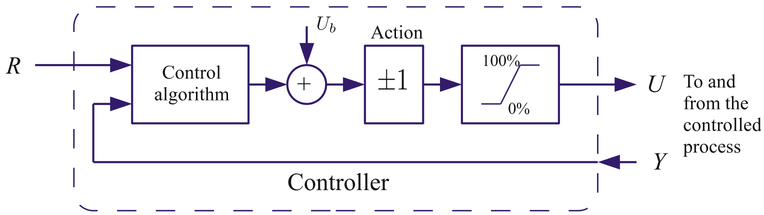



**Fig. 2.1** Controller block diagram

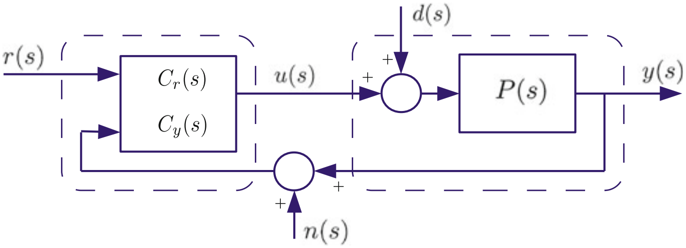



**Fig. 2.2** Closed-loop control block diagram

point given by $\{R_o, Y_o, U_o\}$. Then, we only consider deviation variables $\{r, y, u\}$ around this operating point and then the controller output bias will not be included in following controllers equations.

A linear control system is based on a linearized process model description that relates deviation variables from its operating point values. On that basis, the linearized closed-loop control system for variable deviations $r(s)$, $y(s)$, $u(s)$, and $d(s)$ is reduced as depicted in Fig. 2.2, where $P(s)$ is the transfer function of the controlled process model and $C_r(s)$ and $C_y(s)$ the controller aspects applied to the set-point and the feedback signal, respectively. The possible measurement noise $n(s)$ has been also included.

## 2.2 Two-Degree-of-Freedom (2DoF) PID Control Algorithms

The most widely used proportional integral derivative or PID control algorithms are briefly described below. Each formulation is provided by a specific notation for its parameters in order to distinguish the corresponding implementations when proceeding later on to provide the conversion equations from one algorithm to the other.

***2DoF Standard PID***

The "textbook" 2DoF proportional integral derivative control algorithm is the Standard PID whose output is given by the following [8–10]:

$$u(t) = K_p \left\{ e_p(t) + \frac{1}{T_i} \int_0^t e_i(\xi) \mathrm{d}\xi + T_d \frac{\mathrm{d}e_d(t)}{\mathrm{d}t} \right\}, \tag{2.5}$$

or

$$u(s) = K_p \left\{ e_p(s) + \frac{1}{T_i s} e_i(s) + \frac{T_d s}{\alpha T_d s + 1} e_d(s) \right\}, \tag{2.6}$$

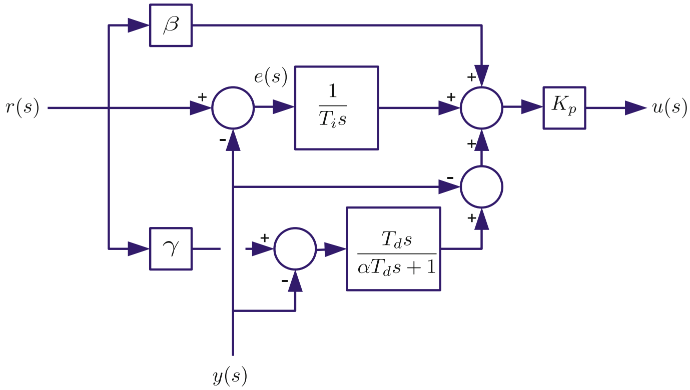



**Fig. 2.3** Two-degree-of-freedom Standard PID controller

with

$$e_p(s) = \beta r(s) - y'(s), \tag{2.7}$$

$$e_i(s) = r(s) - y'(s), \tag{2.8}$$

$$e_d(s) = \gamma r(s) - y'(s), \tag{2.9}$$

$$y'(s) = y(s) + n(s), \tag{2.10}$$

where $K_p$ is the *controller gain*, $T_i$ the *integral time constant*, $T_d$ the *derivative time constant*, $\beta$ and $\gamma$ the *set-point weights*, and $\alpha$ the *derivative filter constant*. The 2DoF PID block diagram is depicted in Fig. 2.3.

To avoid an extreme instantaneous change at the controller output signal when a set-point step change occurs normally $\gamma$ is set to zero [11, 12]. In this case (2.6) reduces to

$$u(s) = K_p \left\{ \beta r(s) - y'(s) + \frac{1}{T_i s}\left[ r(s) - y'(s) \right] - \left( \frac{T_d s}{\alpha T_d s + 1} \right) y'(s) \right\}, \tag{2.11}$$

that will be denoted as $PID_2$. In addition, in the following it is assumed that the measurement noise is filtered, then $y'(s) \approx y(s)$.

The controller output (2.11) may be rearranged, for analysis purposes, as follows:

$$u(s) = K_p \left( \beta + \frac{1}{T_i s} \right) r(s) - K_p \left( 1 + \frac{1}{T_i s} + \frac{T_d s}{\alpha T_d s + 1} \right) y(s), \tag{2.12}$$

where the $C_r(s)$ and $C_y(s)$ controller aspects read as

$$C_r(s) = K_p \left( \beta + \frac{1}{T_i s} \right), \tag{2.13}$$

$$C_y(s) = K_p\left(1 + \frac{1}{T_i s} + \frac{T_d s}{\alpha T_d s + 1}\right), \tag{2.14}$$

being the controller parameters $\theta_c = \{K_p, T_i, T_d, \alpha, \beta, \gamma = 0\}$. Although the Standard form is the classical implementation of the PID control algorithm, the following forms are also found in the control literature [10, 12, 13].

### 2DoF Parallel PID

The parallel or "independent gains" PID control algorithm is

$$u(s) = \left(\beta_p K_p + \frac{K_i}{s}\right) r(s) - \left(K_p + \frac{K_i}{s} + \frac{K_d s}{\alpha_p K_d s + 1}\right) y(s), \tag{2.15}$$

where the $C_r(s)$ and $C_y(s)$ controller aspects read as

$$C_r(s) = \left(\beta_p K_p + \frac{K_i}{s}\right), \tag{2.16}$$

$$C_y(s) = \left(K_p + \frac{K_i}{s} + \frac{K_d s}{\alpha_p K_d s + 1}\right), \tag{2.17}$$

with parameters $\theta_{cp} = \{K_p, K_i, K_d, \alpha_p, \beta_p, \gamma_p = 0\}$. $K_p$ is the proportional gain, $K_i$ the integral gain, and $K_d$ the derivative gain.

### 2DoF Series or "Industrial" PID

The 2DoF version of the series "interacting" implementation of the PID algorithm is

$$u(s) = K_p'\left(\beta' + \frac{1}{T_i' s}\right) r(s) - K_p'\left(1 + \frac{1}{T_i' s}\right)\left(\frac{T_d' s + 1}{\alpha' T_d' s + 1}\right) y(s), \tag{2.18}$$

where the $C_r(s)$ and $C_y(s)$ controller aspects read as

$$C_r(s) = K_p'\left(\beta' + \frac{1}{T_i' s}\right), \tag{2.19}$$

$$C_y(s) = K_p'\left(1 + \frac{1}{T_i' s}\right)\left(\frac{T_d' s + 1}{\alpha' T_d' s + 1}\right), \tag{2.20}$$

with parameters $\theta_c' = \{K_p', T_i', T_d', \alpha', \beta', \gamma' = 0\}$.

### 2DoF Ideal PID with Filter

A commonly used PID implementation in Internal Model Control (IMC)-based controller design is given by the following:

$$u(s) = K_p^*\left(\beta^* + \frac{1}{T_i^* s}\right) r(s) - K_p^*\left(1 + \frac{1}{T_i^* s} + T_d^* s\right)\left(\frac{1}{T_f s + 1}\right) y(s), \tag{2.21}$$

where the $C_r(s)$ and $C_y(s)$ controller aspects read as

$$C_r(s) = K_p^*\left(\beta^* + \frac{1}{T_i^* s}\right), \tag{2.22}$$

$$C_y(s) = K_p^*\left(1 + \frac{1}{T_i^* s} + T_d^* s\right)\left(\frac{1}{T_f s + 1}\right), \tag{2.23}$$

with parameters $\theta_c^* = \left\{K_p^*, T_i^*, T_d^*, T_f, \beta^*, \gamma^* = 0\right\}$. $T_f$ is the controller intput filter time constant.

## 2.3 PID Control Algorithms Conversion Relations

As it can be observed from the presented PID forms, whereas the reference controller aspect takes the same form in all formulations, it is the feedback part the one that prevents a direct translation of the controller parameters from one formulation to another. This is important because some of the existing tuning rules have been conceived for a specific PID formulation. As an example, the derivations of the celebrated SIMC tuning [14] are with the Series or Industrial formulation in mind, whereas much of the other proposals are based on the Standard one. Due to the possibility that the control PID algorithm of the controller to tune be different to the one considered by the tuning rule to use it is necessary to have conversion relations to obtain "equivalent" parameters between two or more of them [15]. In what follows, we present conversion formulae to get the controller parameters for one specific PID formulation starting from the parameters got for another different one.

***Conversion from a 2DoF Parallel PID to a Standard PID***

A $PID_2$ controller (2.11) equivalent to the 2DoF Parallel PID (2.15) is found using the following relations:

$$K_p = K_p, \tag{2.24}$$

$$T_i = \frac{K_p}{K_i}, \tag{2.25}$$

$$T_d = \frac{K_d}{K_p}, \tag{2.26}$$

$$\alpha = \alpha_p K_p, \tag{2.27}$$

$$\beta = \beta_p, \tag{2.28}$$

$$\gamma = \gamma_p = 0. \tag{2.29}$$

There is a direct relation between the Standard and Parallel PID algorithms then this last one will not be further considered.

### *Conversion from a 2DoF Series PID to a Standard PID*

It is possible to obtain a Standard 2DoF PID controller (2.11) equivalent to the 2DoF Series PID (2.18) using the following relations:

$$K_p = F_c' K_p', \tag{2.30}$$

$$T_i = F_c' T_i', \tag{2.31}$$

$$T_d = \frac{(1 - \alpha' F_c') T_d'}{F_c'}, \tag{2.32}$$

$$\alpha = \frac{F_c' \alpha'}{1 - \alpha' F_c'}, \ \ \alpha' < 1 + \frac{T_i'}{T_d'}, \tag{2.33}$$

$$\beta = \frac{\beta'}{F_c'}, \tag{2.34}$$

$$\gamma = \gamma' = 0, \tag{2.35}$$

$$F_c' = 1 + \frac{(1 - \alpha') T_d'}{T_i'}. \tag{2.36}$$

where $F_c'$ (2.36) is the $PID_{2s}$ to $PID_2$ *conversion factor*. It takes into account the derivative filter constant $\alpha'$.

The conversion constraint in (2.33) usually holds then we may say that there is a Standard PID equivalent to a Series one.

### *Conversion from a 2DoF Ideal PID with Filter to a Standard PID*

A Standard 2DoF PID controller (2.11) equivalent to the Ideal PID with filter (2.21), denoted by $PID_{2F}$, can be obtained using the following relations:

$$K_p = F_c^* K_p^*, \tag{2.37}$$

$$T_i = F_c^* T_i^*, \tag{2.38}$$

$$T_d = \frac{T_d^*}{F_c^*} - T_f, \ \ T_d^* > F_c^* T_f, \tag{2.39}$$

$$\alpha = \frac{F_c^* T_f}{T_d^* - F_c^* T_f}, \tag{2.40}$$

$$\beta = \frac{\beta^*}{F_c^*}, \tag{2.41}$$

$$\gamma = \gamma^* = 0, \tag{2.42}$$

$$F_c^* = 1 - \frac{T_f}{T_i^*}, \tag{2.43}$$

$$T_f < T_i^*, \ \text{for PI} \ T_f = 0. \tag{2.44}$$

where $F_c^*$ (2.44) is the $PID_{2F}$ to $PID_2$ *conversion factor*.

In this case, an equivalent $PID_2$ controller cannot always be obtained as shown in (2.39) and (2.44).

Using the conversion factors presented above, exact equivalent feedback ($C_y(s)$) and set-point ($C_r(s)$) controllers transfer functions for a $PID_2$ (2.12) may be obtained for 2DoF PID controllers given by (2.15), (2.18), and (2.21).

Exact equivalent controllers guarantee to obtain the same control system performance and robustness in case a 2DoF PID controller is replaced with a PID controller with a different 2DoF algorithm.

***Conversion from a 2DoF Standard PID to a Series PID***

In the other direction a 2DoF Series PID controller equivalent to a 2DoF Standard one can be found using the following relations:

$$K_p' = F_c K_p, \tag{2.45}$$

$$T_i' = F_c T_i, \tag{2.46}$$

$$T_d' = \frac{(1+\alpha)T_d}{F_c}, \tag{2.47}$$

$$\alpha' = \frac{\alpha F_c}{1+\alpha}, \tag{2.48}$$

$$\beta' = \frac{\beta}{F_c}, \tag{2.49}$$

$$\gamma' = \gamma = 0, \tag{2.50}$$

$$F_c = 0.5\left[1 + \frac{\alpha T_d}{T_i} + \sqrt{1 - \frac{(4+2\alpha)T_d}{T_i} + \frac{\alpha^2 T_d^2}{T_i^2}}\right]. \tag{2.51}$$

Due to the constraint imposed by (2.51) there will not always exist a Series PID equivalent to a Standard PID. It will only exist if

$$\alpha^2 \left(\frac{T_d}{T_i}\right)^2 - (4+2\alpha)\left(\frac{T_d}{T_i}\right) + 1 > 0. \tag{2.52}$$

If the $PID_2$ derivative filter constant is taken as $\alpha = 0.1$ there is a Series equivalent PID controller only if $T_i > 4.20\ T_d$. Figure 2.4 shows that this constrain increases as $\alpha$ increases.

As can be seen in same figure quadratic inequality (2.52) can be approximated by the following straight line for $0 \le \alpha \le 1.0$:

$$\frac{T_i}{T_d} > 4.05 + 1.80\ \alpha. \tag{2.53}$$

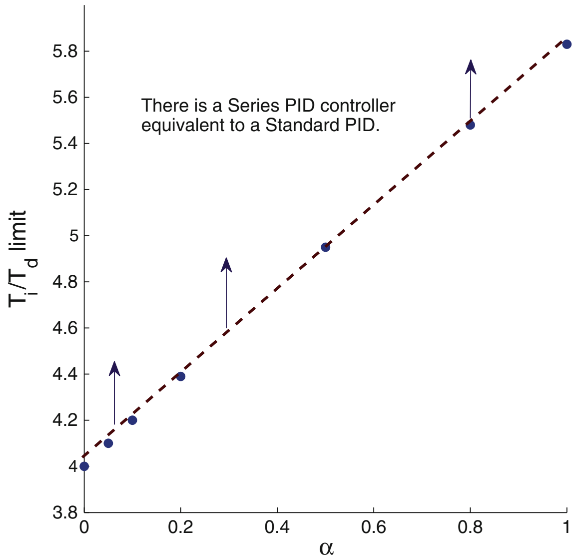



**Fig. 2.4** $T_i/T_d$ condition to obtain a Series PID equivalent to a Standard PID

***Conversion from a 2DoF Standard PID to an Ideal PID with Filter***

The $PID_{2F}$ is a more general control algorithm and, as indicated above, not always an equivalent $PID_2$ controller may be obtained from the $PID_{2F}$ but it is always possible to obtain a $PID_{2F}$ control algorithm equivalent to the $PID_2$ using the following relations:

$$K_p^* = F_{cf} K_p, \tag{2.54}$$

$$T_i^* = F_{cf} T_i, \tag{2.55}$$

$$T_d^* = \left(\frac{1+\alpha}{F_{cf}}\right) T_d, \tag{2.56}$$

$$T_f = \alpha T_d, \tag{2.57}$$

$$\beta^* = \frac{\beta}{F_{cf}}, \tag{2.58}$$

$$\gamma^* = 0, \tag{2.59}$$

$$F_{cf} = 1 + \frac{\alpha T_d}{T_i}. \tag{2.60}$$

where $F_{cf}$ (2.60) is the $PID_2$ to $PID_{2F}$ *conversion factor*.

Considering the above we may say that in the 2DoF PID controllers parametric space $\theta_c' \subset \theta_c \subset \theta_c^*$. Controller parameters conversion equations show that the derivative filter constant $(\alpha, \alpha_p, \alpha')$ must be take into account to obtain an equivalent controller with a different control algorithm from a given one.

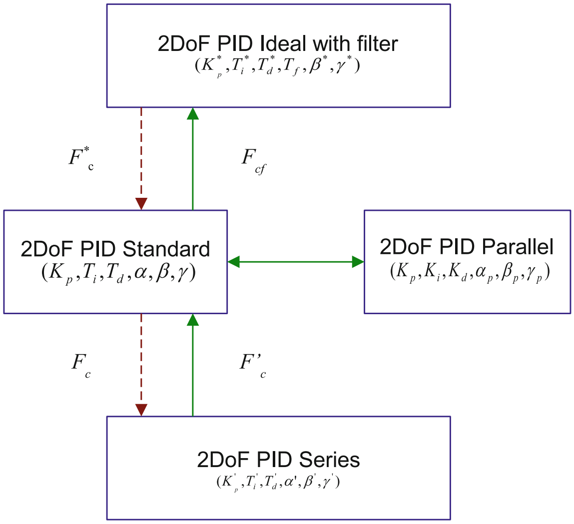



**Fig. 2.5** 2DoF PID controllers conversion

To summarize the above relations a 2DoF PID controllers conversion chart is shown in Fig. 2.5. The solid arrows indicate directions on where there are always equivalent controllers and the dashed arrows the directions on where there are constraints to obtain equivalent controllers. As can be seen in this chart the 2DoF Ideal PID with filter is the most general proportional integral derivative control algorithm.

## 2.4 PID Controller with Two Input Filters

The different signals that enter the PID controller are normally filtered in different ways before they enter the controller. However, as pointed out in [16], a proper choice of these filters can improve the performance of the feedback loop considerably. Therefore, it is important to keep these filters in mind during the design procedures. In order to include into the controller design the measurement noise filter and also to have more freedom for the servo-control design, the control algorithm may be aggregated with two input filters as depicted in Fig. 2.6 [16, 17]. These filters should be considered as an integral part of the design procedure.

The control algorithm is of independent gains (ideal parallel PID implementation) whose output signal is given by [8]:

$$u(s) = K_p \left[r'(s) - y'(s)\right] + \frac{K_i}{s}\left[r'(s) - y'(s)\right] - K_d s y'(s), \qquad (2.61)$$

where $K_p$ is the controller *proportional gain*, $K_i$ the *integral gain*, and $K_d$ the *derivative gain* ($\gamma = 0$).

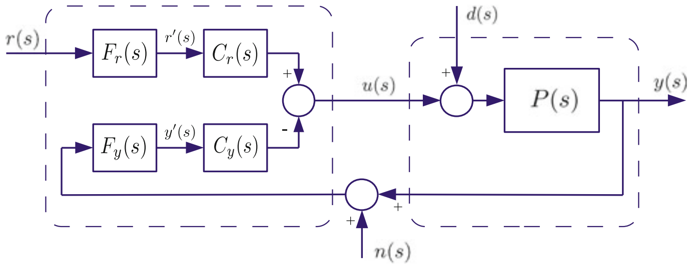



**Fig. 2.6** Closed-loop control system of a controller with two input filters

The set-point $r$ and feedback $y$ signals are filtered before they enter the controller. Then $r'$ and $y'$ in (2.61) are given by

$$r'(s) = F_r(s)r(s), \;\; y'(s) = F_y(s)\left[y(s) + n(s)\right]. \tag{2.62}$$

Using (2.62) into (2.61), it is obtained that

$$u(s) = \left(K_p + \frac{K_i}{s}\right) F_r(s)r(s) - \left(K_p + \frac{K_i}{s} + K_d s\right) F_y(s)\left[y(s) + n(s)\right]. \tag{2.63}$$

In a compact form (2.63) is expressed as

$$u(s) = C_r(s)F_r(s)r(s) - C_y(s)F_y(s)\left[y(s) + n(s)\right]. \tag{2.64}$$

The *set-point filter* $F_r(s)$ is selected strictly proper and given by the transfer function

$$F_r(s) = \frac{\sigma T_r s + 1}{(T_r s + 1)^2}, \tag{2.65}$$

where $T_r$ is its time constant and $\sigma$ an adjustable parameter. Filter (2.65) avoids to have a step change in the controller output when a set-point step change is made.

The *feedback filter* ("noise filter") $F_y(s)$ is selected of first order for PI controllers, given by

$$F_y(s) = \frac{1}{D_{fy}(s)} = \frac{1}{T_f s + 1}, \tag{2.66}$$

with time constant $T_f$, and of second order for PID controllers, given by

$$F_y(s) = \frac{1}{D_{fy}(s)} = \frac{1}{T_f^2/2s^2 + T_f s + 1}, \tag{2.67}$$

to provide high-frequency roll-off (measurement noise attenuation) with either controllers.

Input filters transfer function gains are constrained to be equal, $\lim_{s\to 0} F_r(s) = \lim_{s\to 0} F_y(s)$, to ensure that in steady state the controller integral action operates on the error signal.

Considering $F_r$ and $F_y$ as part of the "controller" be designed the selectable parameters of the set-point controller are $\theta_{cr} = \left\{K_p, K_i, T_r, \sigma, \gamma = 0\right\}$, and corresponding to the feedback controller $\theta_{cy} = \left\{K_p, K_i, K_d, T_f\right\}$. Then, parameters of the controller as a whole are $\theta_c \doteq \theta_{cr} \bigcup \theta_{cy} = \left\{K_p, K_i, K_d, T_f, T_r, \sigma, \gamma = 0\right\}$.

The set-point and feedback signal filters combination with the PID control algorithm is denoted as $PID_{2IF}$ controller. For tuning rules comparison, in addition to the quantitative performance and robustness indices and the responses shapes, the process control-oriented characteristics of the $PID_{2IF}$ controllers must bring to the front.

With the $PID_{2IF}$ controllers there is not any abrupt change at the controller output when a step change is made on the set-point. To mimic this characteristic with a 2DoF PID controller its proportional set-point weight $\beta$ must be made zero. With this, the second degree of freedom is lost and the servo-control response delayed.

The other important characteristic of the $PID_{2IF}$ controllers is their frequency response roll-off. It is normal that in process control applications the feedback signal be corrupted with high-frequency measurement noise. If this noise is not properly filtered it will generate high control signal variations resulting in a deterioration of the final control element. If a measurement noise filter is added to a Standard PID controller *after* its tuning the filter dynamics will affect the control system robustness and performance. Then, it is essential that both these characteristics be part of the controller design from the beginning.

**Chapter Remarks**

The (2DoF) PID algorithm implementations are presented as well as the conversion relations between their parameters.

From the presented PID algorithms the Ideal PID with filter is the more general one.

The aggregation of the deal PID control algorithm with two input filters allows to include two important industrial features: high-frequency roll-off and lack of a control effort abrupt change on a step set-point modification.

## References

1. Araki, M.: On two-degree-of-freedom PID control system. Tech. rep., SICE research committee on modeling and control design of real systems (1984)
2. Araki, M.: Two-degree-of-freedom control system—I. Syst. Control **29**, 649–656 (1985)
3. Araki, M., Taguchi, H.: Two-degree-of-freedom PID controllers. Int. J. Control Autom. Syst. **1**(4), 401–411 (2003)
4. Taguchi, H., Araki, M.: Two-degree-of-freedom PID controllers—their functions and optimal tuning. In: IFAC Digital Control: Past, Present and Future of PID Control, April 5–7. Terrassa, Spain (2000)

5. Taguchi, H., Araki, M.: Survey of researches on two-degree-of-freedom PID controllers. In: The 4th Asian Control Conference, Sept 25–27, Singapore (2002)
6. Gerry, J.P.: A comparison of PID algorithms. Control Eng. **34**(3), 102–105 (1987)
7. O'Dwyer, A.: Handbook of PI and PID Controller Tuning Rules, 3rd edn. Imperial College Press, London (2009)
8. Åström, K.J., Hägglund, T.: PID Controllers: Theory, Design and Tuning. Instrument Society of America, Research Triangle Park, NC 27709, USA (1995)
9. Seborg, D.E., Edgar, T.F., Mellichamp, D.A.: Process Dynamics and Control, 2nd edn. Wiley, Hoboken, NJ 07030, USA (2004)
10. Sung, S.W., Lee, I.B.: Process Identification and PID Control. John Wiley & Sons (Asia) Pte Ltd, Singapore (2009)
11. Tan, W., Yuan, Y., Niu, Y.: Tuning of PID controller for unstable process. In: IEEE International Conference on Control Applications. Kohala Cost-Island of Hawai'i, Hawai'i, USA, Aug 22–27 (1999)
12. Visioli, A.: Practical PID Control. Springer Verlag London Limited (2006)
13. Johnson, M.A.: PID Control—New Identification and Design Methods, chap. PID Control Technology, pp. 1–46. Springer-Verlag London Ltd., U.K. (2005)
14. Skogestad, S.: Simple analytic rules for model reduction and PID controller tuning. J. Process Control **13**, 291–309 (2003)
15. Alfaro, V.M., Vilanova, R.: Conversion formulae and performance capabilities of two-degree-of-freedom PID control algorithms. In: 17th IEEE International Conference on Emerging Technlogies and Factory Automation (ETFA 2012) (2012). Sept 17–21, Kraków, Poland
16. Hägglund, T.: A unified discussion on signal filter in PID control. Control Eng. Pract. **21**, 994–1006 (2013)
17. Alfaro, V.M., Vilanova, R.: Performance and robustness considerations for tuning of proportional integral/proportional integral derivative controllers with two input filters. Ind. Eng. Chem. Res. **52**, 18287–18302 (2013)

# Chapter 3
# Control System Evaluation Metrics

It is clear that when we are faced with a design problem, some measures for the evaluation of such design should be clearly stated. For what matters on a closed-loop control system, there are different aspects that need to be considered. It is necessary to take into account the resulting control system *performance* to set-point and disturbance step changes, the *control effort use*, and its *robustness* to changes in the controlled process dynamics. A review of research history on PID controller design reveals that different performance and robustness metrics have been used. However, in the past years there seems to be a de facto agreement with the use of the integrated absolute error (IAE) because of its direct relationship with economic implications.

Regarding robustness, the ideas emerged from the development of robust control have been particularized to the case of PID control. This permeabilization has led to different approaches of what we can call robust PID. Even the different measures for the closed-loop system robustness, the idea spread today to a common use of the maximum of the sensitivity function (commonly called $M_S$) as a reasonable robustness measure.

There is, however, another consideration that must be taken into account in the control system design process: the effect of the variation of the controller parameters over the control system stability and performance, known as the controller *fragility*. If the control system robustness is an indication of the margin of variation of the process characteristics with a fixed controller, then the controller fragility has a similar meaning but in terms of the variation of the controller parameters considering a fixed controlled process. Even this is not considered as an evaluation to be introduced at the design stage, it is introduced here because the relevance it is gaining among PID researchers.



V.M. Alfaro and R. Vilanova, *Model-Reference Robust Tuning of PID Controllers*,
Advances in Industrial Control, DOI 10.1007/978-3-319-28213-8_3

## 3.1 Closed-Loop Control System

Consider the general closed-loop control system block diagram depicted in Fig. 2.2. In this system we have the following relations:

- Controlled variable feedback signal

$$y(s) = P(s)\,[u(s) + d(s)]\,. \tag{3.1}$$

- Controller output signal

$$u(s) = C_r(s)r(s) - C_y(s)\,[y(s) + n(s)]\,. \tag{3.2}$$

- The error signal

$$e(s) = r(s) - y(s). \tag{3.3}$$

Combining (3.1) and (3.2) it is obtained that the controlled variable is given by the following relation:

$$y(s) = \frac{C_r(s)P(s)}{1 + C_y(s)P(s)}r(s) + \frac{P(s)}{1 + C_y(s)P(s)}d(s) - \frac{C_y(s)P(s)}{1 + C_y(s)P(s)}n(s), \tag{3.4}$$

that can be expressed as

$$y(s) = M_{yr}(s)r(s) + M_{yd}(s)d(s) + M_{yn}(s)n(s). \tag{3.5}$$

## 3.2 Control System Performance Evaluation

The performance of the closed-loop control system may be evaluated with diverse indices, such as those related to the integrated error (difference between the controlled variable set-point and its measured value) as the integrated absolute error (IAE), the integrated time-weighted absolute error (ITAE), or the integrated squared error (ISE), given in general by the following:

$$J_{eg} \doteq \int_0^{\infty} t^p\,|e(t)|^q\,\mathrm{d}t, \tag{3.6}$$

or with other characteristics of the time response to a set-point or to a load-disturbance step change, such as the overshoot, rise or settling time, peak error, or decay ratio [1].

For the controller performance evaluation, it is desirable to use a performance indicator that takes into account economic considerations (an economic performance measure) as the integrated error (IE) does [2]. Taking this into account, to avoid the cancelation of positive and negative errors, we select, as the main control system performance measure, the integrated absolute error (IAE), $p = 0$, $q = 1$ in (3.6), given by the following:

$$J_e \doteq \int_0^\infty |e(t)|\,\mathrm{d}t = \int_0^\infty |r(t) - y(t)|\,\mathrm{d}t. \tag{3.7}$$

The performance measure (3.7) will be evaluated for set-point and load-disturbance changes, $J_{er}$ and $J_{ed}$ depending on the source of the error.

As the controllers will have a proportional integral (PI) or a proportional integral derivative (PID) control algorithm, the steady-state error to both step inputs (set-point and disturbances) will be zero.

## 3.3 Control Effort Use Evaluation

Controller design problems are stated in terms of the controlled variable (usually the process output). Depending on how this problem is stated and solved, this may generate controller settings that produce command signals that are either undesirable or not realistic. It is therefore always needed to evaluate the control signal and take care of the controller bandwidth. This is usually related to the variation of the control signal moves as a measure of its *smoothness*. For the evaluation of the *control effort* required using a tuning rule the control signal total variation $TV_u$ given by

$$TV_u \doteq \sum_{k=1}^{\infty} |u_{k+1} - u_k|\,, \tag{3.8}$$

is used as main indication of the *smoothness* of the control action for both input changes, $TV_{ur}$ and $TV_{ud}$.

As complementary measurements of the control effort use we consider the controller output instant change to a set-point step change (the “proportional kick”) given by

$$\Delta u_0 \doteq \beta K_p \Delta r, \tag{3.9}$$

and the maximum control effort required, $U_{max}$.

## 3.4 Control System Robustness Evaluation

The sole optimization of the control system performance normally results in systems with very poor robustness, and then it is essential to consider the controlled process robustness constrains [3].

There are several quantitative measures of the control system *relative stability* that may be used for the robustness evaluation, such as the classical *Gain Margin* and *Phase Margin* ($A_m$, $\phi_m$) [4], which provide an indication of the distance from the open-loop transfer function, $L(\mathrm{j}\omega)$, frequency response, or Nyquist curve, to the critical point $(-1, 0)$ on the open-loop polar graph. There is also the parametric *Gain Ratio* and the *Delay Ratio* of the *robustness plot* [5], which defines a parametric robustness region.

Another way to express the system robustness is by using the *Stability Margin* ($S_m$) [6], which is the shortest distance from the Nyquist curve to the critical point

$$S_m = \min_{\omega} \left|1 + C_y(\mathrm{j}\omega)P(\mathrm{j}\omega)\right|. \tag{3.10}$$

This distance is the reciprocal of the maximum peak of the sensitivity function, or *Maximum Sensitivity* ($M_S$) [7], defined as follows:

$$M_S \doteq \max_{\omega} |S(\mathrm{j}\omega)| = \max_{\omega} \left|\frac{1}{1 + C_y(\mathrm{j}\omega)P(\mathrm{j}\omega)}\right|. \tag{3.11}$$

Lower values of $M_S$ do imply closed-loop systems with a loop transfer function that is far from the critical point, and therefore more robust. The use of the maximum sensitivity as a robustness measure has the advantage that imposes lower bonds to the gain and phase margins can be assured according to the following [7]:

$$A_m > \frac{M_S}{M_S - 1}, \;\; \phi_m > 2\sin^{-1}\left(\frac{1}{2M_S}\right). \tag{3.12}$$

For the controller robustness evaluation, we will use the maximum sensitivity, $M_S$, as the indication of the closed-loop control system robustness. The normal range for the maximum sensitivity is from 1.2 to 2.0 then; $M_S = 2.0$ will be considered as the minimum allowed robustness. It is important here to highlight the fact of verifying the robustness level achieved by a particular design. In most cases, the robustness is just specified as design specification, without a posterior checking. Here, all the proposed tunings will be developed bearing in mind the accomplishment of the specified robustness.

## 3.5 Controller Fragility Evaluation

There is, however, another consideration that must be taken into account in the control system design process: the effect of the variation of the controller parameters over the control system stability and performance, known as the controller fragility. If the control system robustness is an indication of the margin of variation of the process characteristics with a fixed controller, then the controller fragility has a similar meaning but in terms of the variation of the controller parameters considering a fixed controlled process.

The controller must allow variations of its parameters around their design values, making it easy to fine-tune the controller when the control loop is placed in service. This is the most probable cause of major variations in the controller parameters from their design, or nominal, values. Effectively, most of the tuning approaches, either based on tuning rules or on optimization methods, provide precise values for the controller parameters, but due to the inaccuracies associated with the controlled process model used as part of the tuning procedure, normally these parameters should be taken only as a first approximation, and such final fine tuning of the controller is normally required.

For the tuning rule fragility evaluation (the losses of robustness and/or performance when the controller parameters are changed), the Delta 20 robustness- and performance-fragility indices can be used [8, 9].

***Robustness Fragility***

The controller *Delta-20 Robustness-Fragility Index* relates the control system loss of robustness to its nominal robustness and is given by the following:

$$RFI_{\Delta 20} \doteq \frac{M^m_{S\Delta 20}}{M^o_S} - 1, \tag{3.13}$$

where $M^m_{S\Delta 20}$ represents the highest loss of robustness of the control system when all the parameters $\theta_{cy}$ of the feedback controller have been perturbed 20 % from their nominal values, $\theta^o_{cy}$, considering all the possible combinations of the perturbed parameters, and $M^o_S$ is the control system nominal maximum sensitivity. The robustness fragility definition above considers a 10 % reduction in the control system robustness as marginal and a 50 % reduction as the maximum allowed limit because it turns a highly robust system, with $M_S$ lower than 1.4, into one with a minimally acceptable robustness, $M_S$ of approximately 2.0.

A controller is considered *robustness-fragile* if the control system loses more than 50 % of its robustness when all its parameters change up to 20 %; otherwise, it is *robustness-non-fragile*. In addition, a controller is *robustness-resilient* if the control system does not lose more than 10 % of its robustness when its parameters change up to 20 %. A controller with a low robustness-fragility will allow final fine tuning without a significant reduction in the control system robustness.

The relative influence of a $\delta 20$ change in the controller parameter $p_i$ over its robustness fragility can be obtained with the *Parametric Delta-20 Robustness-Fragility Index* given by the following:

$$RFI^{p_i}_{\delta 20} \doteq \frac{M^{p_i}_{S\delta 20}}{M^o_S} - 1 = \frac{\max\{M_S((1 \pm \delta 20)p_i, \theta^o_{cy})\}}{M_S(\theta^o_{cy})} - 1. \tag{3.14}$$

***Performance Fragility***

The controller *Delta-20 Performance-Fragility Index* relates the maximum loss of the control system performance when a change of up to 20 % occurs in one or more of the nominal controller parameters to its nominal performance and is given by the following:

$$PFI_{\Delta 20} \doteq \frac{J^m_{e\Delta 20}}{J^o_e} - 1, \quad (3.15)$$

were $J^m_{e\Delta 20}$ and $J^o_e$ are the extreme and the nominal performance, respectively, measured by the integrated absolute error (3.7). A controller is considered *performance-fragile* if the control system loses more than 50 % of its performance when all its parameters change up to 20 %; otherwise, it is *performance-non-fragile*. In addition, a controller is *performance-resilient* if the control system does not lose more than 10 % of its performance when its parameters change up to 20 %. The controller performance fragility must be evaluated for the servo-control response, $PFI_{r\Delta 20}$, and for the regulatory control response, $PFI_{d\Delta 20}$.

The relative influence of a $\delta 20$ change in the controller parameter $p_i$ over its performance fragility can be obtained with the *Parametric Delta-20 Performance-Fragility Index*, given by the following:

$$PFI^{p_i}_{\delta 20} \doteq \frac{J^{p_i}_{e\delta 20}}{J^o_e} - 1 = \frac{\max\{J_e((1 \pm \delta 20)p_i, \theta^o_c)\}}{J_e(\theta^o_c)} - 1. \quad (3.16)$$

***Fragility Balance***

To define when a controller is or is not a robustness- or performance-fragility-balanced controller, we must obtain first its *average parametric delta-epsilon-robustness-fragility index*:

$$RFI^a_{\delta 20} \doteq \frac{1}{w}\sum_{i=1}^{w} RFI^{p_i}_{\delta 20}, \quad (3.17)$$

and its *average parametric delta-epsilon-performance-fragility index*:

$$PFI^a_{\delta 20} \doteq \frac{1}{w}\sum_{i=1}^{w} PFI^{p_i}_{\delta 20}, \quad (3.18)$$

where the number of parameters is two for a PI ($w = 2$), and three for a PID ($w = 3$).

Based on the parametric delta-epsilon-fragility indices a robustness (performance)-fragility-balanced PID controller is one in which all its parametric robustness (performance) delta-epsilon-fragility indices are within a selected

**Table 3.1** Control systems evaluation metrics.

| Metric | Equation |
|---|---|
| Performance (integrated absolute error) | $J_e \doteq \int_0^\infty \lvert e(t)\rvert \, \mathrm{d}t = \int_0^\infty \lvert r(t) - y(t)\rvert \, \mathrm{d}t$ (3.7) |
| Control effort (total variation) | $TV_u \doteq \sum_{k=1}^{\infty} \lvert u_{k+1} - u_k\rvert$ (3.8) |
| (instant change) | $\Delta u_0 \doteq \beta K_p \Delta r$ (3.9) |
| Robustness (maximum sensitivity) | $M_S \doteq \max_\omega \left\lvert \frac{1}{1+C_y(\mathrm{j}\omega)P(\mathrm{j}\omega)} \right\rvert$ (3.11) |
| Fragility (robustness fragility) | $RFI_{\Delta 20} \doteq \frac{M^m_{S\Delta 20}}{M^o_S} - 1$ (3.14) |
| (performance fragility) | $PFI_{\Delta 20} \doteq \frac{J^m_{e\Delta 20}}{J^o_e} - 1$ (3.15) |

$\pm\sigma\,\%$ band (usually $\pm 25\,\%$) centered on its average parametric delta-epsilon-robustness (performance)-fragility index, $RFI^a_{\delta 20}$ ($PFI^a_{\delta 20}$); otherwise, it is a robustness (performance)-fragility-unbalanced controller.

The robustness- or performance-fragility unbalance of a controller is caused by the controller parameter with the highest parametric robustness- or performance-fragility index.

**Chapter Remarks**

The metrics used to evaluate the control system performance, control effort use, control system robustness, and controller fragility are presented. These will allow to highlight the main characteristics of a tuning procedure.

In order to have a fair comparison between tuning rules their evaluation must be made using the same "base-line." At this respect a control system robustness level can be used as the "equalizer" parameter.

## 3.6 Evaluation Metrics Summary

The metrics used for control systems evaluation are summarized in Table .
In this table:

- $\beta$—controller proportional set-point weight,
- $\Delta r$—set-point step change,
- $\Delta u_0$—controller output instant change,
- $C_y(s)$—feedback controller transfer function,
- $e$—error signal,
- $PFI_{\Delta 20}$—delta-20 performance fragility,
- $J_e$—integrated absolute error (IAE),
- $J^o_e$—nominal IAE performance,
- $J^m_{e\Delta 20}$—delta-20 IAE performance (worst case),
- $K_p$—controller gain,
- $M_S$—maximum sensitivity,

- $M_S^o$—nominal maximum sensitivity,
- $M_{S\Delta 20}^m$—delta-20 maximum sensitivity (worst case),
- $P(s)$—controlled process model transfer function,
- $RFI_{\Delta 20}$—delta-20 robustness fragility,
- $r$—set-point,
- $TV_u$—control effort total variation,
- $u$—control effort (controller output signal),
- $y$—controlled variable feedback signal.

## References


1. Swanda, A.P., Seborg, D.E.: Controller performance assessment based on setpoint response data. Proceedings of the American Control Conference pp. 3863–3867: San Diego. California, USA (1999)
2. Shinskey, F.G.: Process control: as taught vs as practiced. Ind. Eng. Chem. Res. **41**, 3745–3750 (2002)
3. Ingimundarson, A., Hägglund, T., Åström, K.J.: Criteria for design of PID controllers. In: 2nd IFAC Conference Control Systems Design (CSD'03). Bratislava, Slovak Republic (2003)
4. Goodwin, G.C., Graebe, S.F., Salgado, M.E.: Control System Design. Prentice Hall Inc, NJ (2001)
5. Gerry, J.P., Hansen, P.D.: Choosing the right controller. Chem. Eng. 65–68 (1987)
6. Åström, K.J., Murray, R.: Feedback Systems: An Introduction for Scientists and Engineers. Princenton University Press, Oxfordshire (2008)
7. Åström, K.J., Hägglund, T.: PID Controllers: Theory, Design and Tuning. Instrument Society of America, Research Triangle Park, NC 27709, USA (1995)
8. Alfaro, V.M.: PID controllers' fragility. ISA Trans. **46**, 555–559 (2007)
9. Alfaro, V.M., Vilanova, R.: PID Control in the Third Millenium: Lessons Learned and New Approaches, chap. Fragility Evaluation of PI and PID Controllers Tuning Rules, pp. 349–380. Springer (2012)

# Chapter 4
# Model-Reference Robust Tuning Design Methodology

Over the years, the design of controllers with PID control algorithms has been faced with different approaches. As shown in [1] controller tuning rules may be classified using different criteria: based on the controlled process information used (model order and structure, critical information), on the control algorithm to tune (P, PD, PI, PID, one or two degrees of freedom), and on the controller design criteria (performance, robustness, or a combination of both) using analytical or optimization procedures. All the history of research work on PID controller design leads us with a really wide *choice-menu*. However, one of the aspects that is shared by all modern approaches is that of including robustness considerations as a must. Therefore, it is the way in through which the design approach is able to combine the time response, or performance aspects, with the stability (robustness) ones that determines the utility, versatility, etc.

A possible control system design approach is by specifying the desired targets for the closed loop. Regarding the time responses or the performance of the closed-loop control system, a possible design approach is by specifying the closed-loop poles and zeros location. However, instead of doing this by the usual pole-placement approach, say the algebraic way, where the closed-loop transfer functions are to have those desired pole locations, the desired control system closed-loop transfer functions are stated (as a target or model reference). From these target transfer functions, an optimization procedure is used in order to obtain the controller parameters that provide the best match to the desired responses. When the optimization is pursued a target closed-loop control system robustness is also ensured.

Based on the previous statement, this chapter presents the proposed controller design procedure that is based on the use of closed-loop transfer functions targets (the *reference models*) to obtain robust control systems, therefore named *Model-Reference Robust Tuning* (MoReRT), as outlined in [2].



V.M. Alfaro and R. Vilanova, *Model-Reference Robust Tuning of PID Controllers*,
Advances in Industrial Control, DOI 10.1007/978-3-319-28213-8_4

## 4.1 Introduction

Consider the general two-degree-of-freedom closed-loop control system depicted in Fig. 4.1 where $P(s)$ and $\{C_r(s), C_y(s)\}$ are the controlled process model and the controller transfer functions, respectively. In this system, $r(s)$ is the set-point, $u(s)$ the controller output signal, $d(s)$ the disturbance, $y(s)$ the process controlled variable, and $n(s)$ the measurement noise. It is assumed that the disturbance enters at the process input (load-disturbance). Controlled process model parameters are $\theta_p$.

As derived in Sect. 3.1 the closed-loop control system output $y(s)$ as a function of its inputs $r(s)$, $d(s)$, and $n(s)$ is

$$y(s) = M_{yr}(s)r(s) + M_{yd}(s)d(s) + M_{yn}(s)n(s), \tag{4.1}$$

where

$$M_{yr}(s) \doteq \frac{C_r(s)P(s)}{1 + C_y(s)P(s)}, \tag{4.2}$$

is the servo-control closed-loop transfer function,

$$M_{yd}(s) \doteq \frac{P(s)}{1 + C_y(s)P(s)}, \tag{4.3}$$

the regulatory control closed-loop transfer function, and

$$M_{yn}(s) \doteq \frac{-C_y(s)P(s)}{1 + C_y(s)P(s)}, \tag{4.4}$$

the measurement noise sensitivity function.

The regulatory control main objective is *load-disturbance rejection*; this is, to return the controlled variable to its set-point in the event a disturbance enters to the control system. For the servo-control, it is intended to *follow a changing set-point*; this is, to bring the controlled variable to its new desired value. Controller tuning for above operations must take also into account to not amplified the measurement noise, if any.

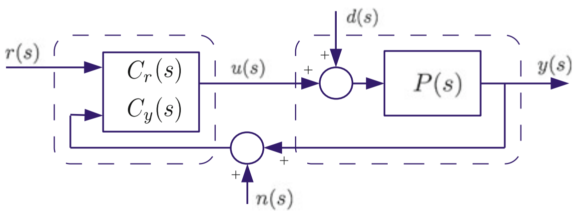



**Fig. 4.1** Two-degree-of-freedom closed-loop control system

Bearing in mind the regulatory and servo operations, assume the regulatory control target response is stated using a target model transfer function $M^t_{yd}(s)$ as

$$y^t_d(s) = M^t_{yd}(s)d(s), \tag{4.5}$$

and along the same lines, the servo-control target transfer function, $M^t_{yr}(s)$, that does generate the desired reference following response as

$$y^t_r(s) = M^t_{yr}(s)d(s). \tag{4.6}$$

Then, using (4.5) and (4.6) the global control system noise free output target, $y^t(s)$, is computed as follows:

$$y^t(s) = M^t_{yr}(s)r(s) + M^t_{yd}(s)d(s), \tag{4.7}$$

and in the time domain as follows:

$$y^t(t) = y^t_r(t) + y^t_d(t). \tag{4.8}$$

The design goal will be to find a controller that generates a global system response that matches the target one (4.8) as much as possible. Next section states the optimization problem that will formally specify this closeness to the desired target responses.

## 4.2 Optimization Cost Functionals

To obtain the controller parameters, that best match the target response (4.8) in the *least-squares sense*, a minimization procedure is used based on the differences between the target responses and the actual ones. One of the advantages of the proposed design framework is that similar approaches can be worked out, but using a different metric for measuring the distance between the target and closed-loop responses.

For the regulatory control model-reference response optimization the cost functional to be optimized is defined as follows:

$$J_d(\theta_p, \theta_{cy}, \theta_d) \doteq \int_0^{\infty} \left[y^t_d(\theta_p, \theta_{cy}, \theta_d, t) - y_d(\theta_p, \theta_{cy}, t)\right]^2 \mathrm{d}t, \tag{4.9}$$

where $y^t_d(\theta_p, \theta_{cy}, \theta_d, t)$ is the regulatory control closed-loop step response target and $y_d(\theta_p, \theta_{cy}, t)$ is that of the regulatory control system with controlled process model $P(s)$ and the feedback controller $C_y(s)$.

In a similar way, the servo-control cost functional to be optimized is defined as follows:

$$J_r(\theta_p, \theta_c, \theta_d) \doteq \int_0^{\infty} \left[y_r^t(\theta_p, \theta_c, \theta_d, t) - y_r(\theta_p, \theta_c, t)\right]^2 \mathrm{d}t, \tag{4.10}$$

where $y_r^t(\theta_p, \theta_c, \theta_d, t)$ is the step response of the servo-control closed-loop transfer function target and $y_r(\theta_p, \theta_c, t)$ is that of the servo-control system with the controlled process $P(s)$ and the controller $C(s)$.

For the 2DoF controllers design, the following overall cost functional is optimized:

$$J_T(\theta_p, \theta_c, \theta_d) \doteq J_r(\theta_p, \theta_c, \theta_d) + J_d(\theta_p, \theta_{cy}, \theta_d). \tag{4.11}$$

Using (4.11) the controller parameters $\theta_c^o$ are obtained such that

$$J_T^o \doteq J_T(\theta_p, \theta_c^o, \theta_d) = \min_{\theta_c} J_T(\theta_p, \theta_c, \theta_d), \tag{4.12}$$

for design parameters $\theta_d$ selected in such a way that the control system robustness matches a target value (*robust design*) measured using the maximum sensitivity, $M_S$.

The performance/robustness trade-off in PID controller design is a well-known issue and the $M_S$ has become the de facto robustness measure [3]. Robustness requirements are related with the expected changes in the controlled process dynamics from the nominal model used for controller tuning. An *a priori* evaluation of this requirement is needed to state the minimum robustness level desired for the designed control system. Then, if robustness is a must the design is focused on the attainable performance and/or control effort characteristics, under the selected metrics, for the stated robustness.

The optimization of (4.11) is a *closed-loop model matching problem*, instead of a control system performance optimization problem as in the traditional PI and PID controllers optimization procedures. In the usual optimization approach to PI and PID controller design, the goal is to minimize some integral cost function related to the feedback error. In this case, instead, the control system behavior is stated by the closed-loop regulatory and servo-control transfer functions targets and the design parameters $\theta_d$. The optimization procedure searches for the controller parameters $\theta_c$ that achieve the best match between the actual overall control system response and the target one.

The complete set of controller parameters $\theta_c$ are obtained when considering the regulatory control response and the servo-control response at once.

A target robustness $M_S^t$ is used as a *soft constraint*. For a given controlled process model the resulting control system robustness depends on the design parameters then, it is evaluated after each closed-loop model-reference response optimization and the design parameters adjusted to meet this target. Therefore, there is a need to accommodate the inclusion of this constraint into the optimization stage. For what matters to the optimizations carried out in this book, a direct search Nelder-Mead [4] simplex-based algorithm [5, 6] is used. On the other hand, the target transfer functions (as will be seen in the subsequent chapters) are parameterized in such a way that depend on a single parameter that is iteratively adjusted in order to match the desired robustness target. More details about this iterative procedure and the practical implementation of the MoReRT design approach are provided on the last chapter.

## 4.3 Closed-Loop Reference Models

In (4.2)–(4.4) $C_r(s)$ is the 2DoF controller aspect that operates on the set-point $r(s)$, the *set-point controller* with parameters $\theta_{cr}$, and $C_y(s)$ the controller aspect that operates on the controlled variable $y(s)$, the *feedback controller* with parameters $\theta_{cy}$, then $C(s) = \{C_{cr}(s), C_{cy}(s)\}$.

The controlled process model and the feedback controller transfer functions are expressed as a quotient of polynomials in $s$ as

$$P(s) = \frac{N_p^-(s)N_p^+(s)}{D_p(s)}, \tag{4.13}$$

and

$$C_y(s) = \frac{N_{cy}(s)}{D_{cy}(s)}, \tag{4.14}$$

where $N_p^+(s)$ is the controlled process model non-invertible part (dead-time and/or right-half plane zeros).

Replacing $P(s)$ and $C_y(s)$ in (4.3) by (4.13) and (4.14) the regulatory control closed-loop transfer function can be expressed by

$$M_{yd}(s) = \frac{D_{cy}(s)N_p^-(s)N_p^+(s)}{D_{cy}(s)D_p(s) + N_{cy}(s)N_p^-(s)N_p^+(s)} = \frac{D_{cy}(s)N_p^-(s)N_p^+(s)}{D_M(s)}. \tag{4.15}$$

where $D_M(s)$ stands for the closed-loop denominador common to both the servo and regulatory closed-loop transfer functions. The feedback controller $C_y(s)$ design methodology will use a regulatory control closed-loop transfer function target for (4.15), $M_{yd}^t(s)$, that depends on the controlled process model non-minimum phase components, the rest of model parameters, the feedback controller parameters, and the control system *design parameters*, $\theta_d$, with the following general form:

$$M_{yd}^t(s) = \mathscr{M}_d(N_p^+(s), \theta_p, \theta_{cy}, \theta_d, s). \tag{4.16}$$

Then, the regulatory control response target is

$$y_d^t(s) = M_{yd}^t(s)d(s). \tag{4.17}$$

Using (4.16) in (4.2) the servo-control closed-loop transfer function target is given by the following general function:

$$M_{yr}^{t}(s) = \mathscr{M}_r(N_p^{+}(s), \theta_p, \theta_c, \theta_d, s) = C_r(s)M_{yd}^{t}(s), \tag{4.18}$$

where the controller parameters $\theta_c = \theta_{cr} \cup \theta_{cy}$.

In the next chapters, when facing the controller design for the considered process dynamics, once the expressions for the process model transfer function and the selected controller are got, the target dynamics for the servo and regulatory control performance are specified. As it can be expected, quite different possibilities do exists regarding these target dynamics. For what matter to the extent of this book, all desired dynamics have been chosen on the basis of specifying a desired non-oscilatory dynamics for the closed loop. This may not be the case on some specific cases. In such situation, the proposed design framework can be easily accommodated by introducing the specific target transfer functions.

### Chapter Remarks

In this chapter, the groundings of the proposed model-reference robust tuning (MoReRT) method have been presented. It starts with the selection of regulatory control and servo-control closed-loop transfer functions targets $M_{yd}^{t}(s)$ and $M_{yr}^{t}(s)$, respectively. These transfer functions state the desired shapes for the disturbance and set-point step responses $y_d^t(t)$ and $y_r^t(t)$.

The time responses target specification must take into account the controlled process model characteristics, particularly its non-minimum phase components and order. They depend also on the design parameters $\theta_d$ that affect the control system performance, control effort use, and robustness.

Controller parameters $\theta_c$ are obtained by optimizing a cost functional that evaluates the difference between the target total response (regulatory control response plus servo-control response) and the corresponding obtained with the controller to tune. Design parameters are adjusted to obtain a robustness level target, $M_S^t$.

## References

1. O'Dwyer, A.: Handbook of PI and PID Controller Tuning Rules, 3rd edn. Imperial College Press, London (2009)
2. Alfaro, V.M., Vilanova, R.: Performance/robustness trade-off design framework for 2DoF PI controllers. Stud. Inf. Control **21**(1), 73–83 (2012)
3. Vilanova, R., Alfaro, V.M.: Control PID robusto: una visión panorámica. Revista Iberoamericana de Automática e Informatica Industrial **8**, 141–158 (2011)
4. Lagarias, J.C., Reeds, J.A., Wright, M.H., Wright, P.E.: Convergence properties of the Nelder-Mead simplex algorithm in low dimensions. SIAM J. Optim. **9**, 112–147 (1998)
5. The MathWorks, Inc.: Optimization Toolbox User's Guide (R2011a). The MathWorks, Inc., Natick, MA, USA (2011)
6. Mathews, J.H., Fink, K.D.: Numerical Methods Using MATLAB, 4th edn. Prentice-Hall, Inc., NJ (2004)

# Chapter 5
# Normalized Controlled Process Models and Controllers

The design methodology outlined in Chap. 4 is applied to controlled processes represented by stable overdamped and inverse response models, and as well by integrating and unstable models. For control system performance analysis and controller tuning it is convenient to work with dimensionless parameters to make it nondependent on the controlled process timescale and gain. Therefore, in the next sections the different processes and controllers transfer functions to be considered will be presented as well as their normalization in terms of dimensionless parameters.

## 5.1 Timescaling and Consistent Controller Design

The timescale property is of special use in control systems because it facilitates the analysis task and allows the use of normalized forms for processes and controllers, making the design statement independent of the time properties of the system at hand. It is customary to plot Bode diagrams as well as set-point responses for normalized first- and scond-order transfer functions. Even this may sound quite clear, it is not since the original work of Atherton and Boz [1] that attention is put on this issue regarding PID controller tuning. The fact is that there do exist several PI/PID tuning rules that are not consistent with the time scale property. Therefore, these tuning rules will not ensure the same response properties for different timescaled transfer functions. As an example, the Ziegler–Nichols first tuning rule [2] does satisfy this timescale property. On the other hand the Chien et al. [3] tuning rule for set-point response does not satisfy the timescaling property.

The referred timescaling property is stated in [1] for first-order plus dead-time transfer functions

$$P(s) = \frac{Ke^{-Ls}}{Ts + 1},$$



V.M. Alfaro and R. Vilanova, *Model-Reference Robust Tuning of PID Controllers*,
Advances in Industrial Control, DOI 10.1007/978-3-319-28213-8_5

that can be timescaled by replacing $Ts$ by $\hat{s}$ in order to give the normalized process transfer function

$$P_n(s) = \frac{Ke^{-\tau_L \hat{s}}}{\hat{s}+1},$$

where $\tau_L = L/T$. Now, if the process is controlled by an ideal PID controller $K_p[1 + sT_d + 1/(sT_i)]$, the same kind of time responses will be obtained for the original, non-timescaled, transfer function, provided the design is consistent. This is specified in [1] as the controller parameters to obey to the form $K_p = f_1(\tau_L)/K$, $T_i = Tf_2(\tau_L)$ and $T_d = Tf_3(\tau_L)$. As an example of a suitable choice for those $f_i(\tau_L)$ is $f_1(\tau_L) = k_1/\tau_L$, $f_2(\tau_L) = k_2\tau_L$ and $f_3(\tau_L) = k_3\tau_L$. That corresponds to the case of the Ziegler and Nichols [2] rules with $k_1 = 1.2$, $k_2 = 2$ and $k_3 = 0.5$.

As this book is devoted to the development of PI/PID tuning rules for a wide range of processes dynamics, generality is gained by working with normalized transfer functions (both of process model and controller) from the very beginning. Therefore, in this book, all the results are based on normalized transfer function models as well as the corresponding normalized controller parameters. In this way, we ensure that controller design is consistent from the point of view of being applicable to all transfer function models equivalent to the normalized one but to a timescaling. An additional advantage is that the number of process model transfer function does have one parameter less.

In the original [1] paper, just first-order and second-order plus dead-time were considered. Also, the ideal PID controller was used for the derivation of the corresponding relations and designs. In the rest of this chapter, we apply this normalization to all controller formulations considered as well as process transfer function models. Therefore, the results provided here can be seen of utility in its own.

## 5.2 Controlled Process Normalized Parameters Models

Regarding the considered process models, four different dynamics are considered but are represented by six model transfer functions:

1. Overdamped dynamics, both first and second order. Therefore will have the overdamped first-order plus dead-time (FOPDT) and second-order plus dead-time (SOPDT) process models.
2. Inverse response dynamics, represented by a second-order plus a right-half plane zero (SOPRHPZ) process models.
3. Integrating dynamics represented by integrating plus dead-time (IPDT) and second-order plus dead-time (ISOPDT) process models.
4. Unstable dynamics represented by the unstable first-order plus dead-time (UFOPDT) process model.

In what follows, each one of the previous dynamics is presented and the corresponding process model transfer function is specified and normalized.

1. The *overdamped* controlled process (first- and second-order) are represented by a linear model given by the transfer function

$$P(s) = \frac{K\mathrm{e}^{-Ls}}{(Ts+1)(aTs+1)}, \ \theta_p = \{K, T, a, L\}, \tag{5.1}$$

where $K$ is the model gain, $T$ the main time constant, $a$ the ratio of the two time constants ($0 \leq a \leq 1.0$), and $L$ the dead-time.
Using the controlled process model gain $K$, and time constant $T$, as well as the transformation $\hat{s} \doteq Ts$, the controlled process (5.1) can be expressed in normalized form as follows:

$$\hat{P}(\hat{s}) = \frac{\mathrm{e}^{-\tau_L \hat{s}}}{(\hat{s}+1)(a\hat{s}+1)}, \ \tau_L \doteq \frac{L}{T}, \tag{5.2}$$

where $\tau_L$ is the normalized (dimensionless) dead-time.
The overdamped second-order plus dead-time (SOPDT) (5.2) model has two normalized parameters, $\hat{\theta}_p = \{a, \tau_L\}$. For the particular case of the first-order plus dead-time (FOPDT) model ($a = 0$) it has only one, $\hat{\theta}_p = \tau_L$. Using the same procedure normalized models are obtained for the other processes.
2. The controlled process with *inverse response* characteristics, i.e., the controlled process initial response to a step change is in the opposite direction to that the steady-state direction, are represented by a second-order plus a right-half plane zero (SOPRHPZ) model given by the transfer function

$$P(s) = \frac{K(-bTs+1)}{(Ts+1)(aTs+1)}, \ \theta_p = \{K, T, a, b\}, \tag{5.3}$$

where $K$ is the model gain, $T$ the main time constant, $a$ the ratio of the two main poles time constants ($0.1 \leq a \leq 1.0$), and $b$ the relative position of the right-half plane zero with respect to the main time constant.
The controlled process (5.3) can be expressed in normalized form as follows:

$$\hat{P}(\hat{s}) = \frac{-b\hat{s}+1}{(\hat{s}+1)(a\hat{s}+1)}, \ \hat{\theta}_p = \{a, b\}. \tag{5.4}$$

3. Even though most of the controlled processes found in the process industry are self-regulating, i.e., the process output seeks a stable point under a constant input, there are others that under a constant input their output is unbounded, rise, or decrease without limit. These nonself regulated processes are named integrating or unstable if their model transfer function has poles at the complex plane origin or at the right-half plane, respectively.

*Integrating* second-order plus dead-time (ISOPDT) models given by the transfer function

$$P(s) = \frac{K\mathrm{e}^{-Ls}}{s(Ts+1)}, \;\; \theta_p = \{K, T, L\}, \tag{5.5}$$

can be expressed in normalized form as:

$$\hat{P}(\hat{s}) = \frac{\mathrm{e}^{-\tau_L \hat{s}}}{\hat{s}(\hat{s}+1)}, \;\hat{\theta}_p = \tau_L. \tag{5.6}$$

Integrating as well as stable processes with very long time constants can be represented by an integrating plus dead-time (IPDT) model given by the following:

$$P(s) = \frac{K\mathrm{e}^{-Ls}}{s}, \;\; \theta_p = \{K, L\}, \tag{5.7}$$

where K is the gain and $L$ the dead-time. In this case using the transformation $\tilde{s} \doteq Ls$, the controlled process (5.7) can be expressed in normalized form as follows:

$$\tilde{P}(\tilde{s}) = \frac{\mathrm{e}^{-\tilde{s}}}{\tilde{s}}. \tag{5.8}$$

The normalized controlled process IPDT model (5.8) has no parameters ($\tilde{\theta}_p = \emptyset$).

4. We consider the *unstable* first-order plus dead-time (UFOPDT) model given by the transfer function

$$P(s) = \frac{K\mathrm{e}^{-Ls}}{Ts-1}, \;\; \theta_p = \{K, T, L\}\,, \tag{5.9}$$

that can be expressed in normalized form as follows:

$$\hat{P}(\hat{s}) = \frac{\mathrm{e}^{-\tau_L \hat{s}}}{\hat{s}-1}, \;\hat{\theta}_p = \tau_L. \tag{5.10}$$

From (5.2), (5.4), (5.6), (5.8), and (5.10) it can be seen that the normalized models can have from zero to two dimensionless parameters.

## 5.3 Normalized Controllers Parameters

Regarding the controller, to consider the control transfer function alone does not make sense. It has to be considered in conjunction with the process model transfer function to be controlled. To work with controlled process normalized models, (F)SOPDT (5.2), SOPRHPZ (5.4), UFOPDT (5.10), ISOPDT (5.6), and IPDT (5.8), a normalized controller is required. Therefore, according to the normalization of the process model, the controlled transfer function has also to be scaled. This will define the normalized controller parameters.

Next, we consider the normalization of the *Standard 2DoF PID controller* $PID_2$ from where the normalized parameters of other 2DoF PID control algorithms can be found using the conversion relations stated in Sect. 2.3. Also, the normalized expressions for the filtered *PID Parallel controller* is considered.

1. For example, the output equation of the normalized version of the *Standard 2DoF PID controller* $PID_2$ (2.11), with the $\hat{s} \doteq Ts$ transformation, is given by

$$u(\hat{s}) = \kappa_p \left\{ \beta r(\hat{s}) - y(\hat{s}) + \frac{1}{\tau_i \hat{s}} \left[ r(\hat{s}) - y(\hat{s}) \right] - \left( \frac{\tau_d \hat{s}}{\alpha \tau_d \hat{s} + 1} \right) y(\hat{s}) \right\}, \quad (5.11)$$

with parameters $\hat{\theta}_c = \left\{ \kappa_p, \tau_i, \tau_d, \alpha, \beta \right\}$.

- For *overdamped first-* and *second-order plus dead-time*, *second-order plus right-half plane zero*, and *unstable first-order plus dead-time* models, using the corresponding model parameters the associated $PID_2$ ($PID_{2F}$) controllers parameters can be expressed in normalized form as follows:

$$\kappa_p \doteq KK_p, \;\; \tau_i \doteq \frac{T_i}{T}, \;\; \tau_d \doteq \frac{T_d}{T}, \;\; \tau_f \doteq \frac{T_f}{T}, \quad (5.12)$$

- For the *integrating second-order plus dead-time* model the normalized controller parameters are

$$\kappa_p \doteq KK_pT, \;\; \tau_i \doteq \frac{T_i}{T}, \;\; \tau_d \doteq \frac{T_d}{T}, \;\; \tau_f \doteq \frac{T_f}{T}. \quad (5.13)$$

  For these models the new control system timescale is $\hat{\tau} \doteq t/T$.

- Now, for the particular case of the *integrating plus dead-time* model the $\tilde{s} \doteq Ls$ transformation is used and the controller parameters can be expressed in normalized form as follows:

$$\kappa_p \doteq KK_pL, \;\; \tau_i \doteq \frac{T_i}{L}, \;\; \tau_d \doteq \frac{T_d}{L}, \;\; \tau_f \doteq \frac{T_f}{L}, \quad (5.14)$$

  and the new control system timescale in this case is $\tilde{\tau} \doteq t/L$.

2. The normalized parameters of the *PID Parallel controller* (2.63) *aggregated with a set-point input filter* (2.65) *and a feedback input filter* (2.66) or (2.67) ($PID_{2IF}$) are:

- For *overdamped first-* and *second-order plus dead-time*, *second-order plus right-half plane zero*, and *unstable first-order plus dead-time* models are

$$\kappa_p \doteq KK_p,\ \kappa_i \doteq KK_iT,\ \kappa_d \doteq \frac{KK_d}{T},\ \tau_f \doteq \frac{T_f}{T},\ \tau_r \doteq \frac{T_r}{T}. \tag{5.15}$$

- For *integrated second-order plus dead time* models the corresponding $PID_{2IF}$ controller normalized parameters are

$$\kappa_p \doteq KK_pT,\ \kappa_i \doteq KK_iT^2,\ \kappa_d \doteq KK_d,\ \tau_f \doteq \frac{T_f}{T},\ \tau_r \doteq \frac{T_r}{T}. \tag{5.16}$$

- Finally, for the *integrated plus dead-time* models $PID_{2IF}$ controller normalized parameters are

$$\kappa_p \doteq KK_pL,\ \kappa_i \doteq KK_iL^2,\ \kappa_d \doteq KK_d,\ \tau_f \doteq \frac{T_f}{L},\ \tau_r \doteq \frac{T_r}{L}. \tag{5.17}$$

### Chapter Remarks

This chapter has introduced the idea of timescaling into the formulation of the controller by considering the corresponding previous normalization of the process model to be controlled. This will allow to obtain later normalized (dimensionless) controller tuning rules, in terms of the defined normalized controlled process models and controller parameters.

It is important to note that there are normalized controlled process models with no parameters (IPDT), with only one parameter (FOPDT, UFOPDT, and ISOPDT), and with up to two parameters (SOPDT and SOPRHPZ). This will have an impact on the tuning rule simplicity.

## References

1. Atherton, D., Boz, A.: Time scaling in PID controller tuning. Trans. Inst. Meas. Control **31**(5), 425–433 (2009)
2. Ziegler, J.G., Nichols, N.B.: Optimum settings for automatic controllers. ASME Trans. **64**, 759–768 (1942)
3. Chien, I., Hrones, J., Reswick, J.: On the automatic control of generalized passive systems. ASME Trans. **74**, 175–185 (1952)

# Chapter 6
# MoReRT Control of Overdamped Processes

In Chap. 4, the proposed model-reference robust tuning design methodology was described. In the following, this procedure is applied to obtain robust PI and PID controllers to control stable overdamped processes. First of all the generic SOPDT process model transfer function will be presented followed by the design of PI and PID controllers on a separate way. The application of the MoReRT design is done by showing first the particularization of the design framework and specification of target models. Following the obtention of the optimization results, the verification of the results regarding the robustness and evaluation of the servo and regulatory performance.

## 6.1 Introduction

Most of the industrial controlled processes are self-regulated (stable) and also overdamped (their open-loop response has no oscillation). Then it is not a surprise that more attention has been place to obtain controller tuning rules for this type of processes.

To represent the dynamic behavior of these processes the general SOPDT model (5.1) is used, rewrite here

$$P(s) = \frac{K\mathrm{e}^{-Ls}}{(Ts+1)(aTs+1)}, \quad \theta_p = \{K, T, a, L\}\,. \tag{6.1}$$

This model covers first-order plus dead-time models with $a = 0$, dual-pole plus dead-time models with $a = 1$, and any overdamped plus dead-time model in between ($0 < a < 1$). Model (6.1) may be obtained from a open or closed-loop test data.

The *control relevant* controlled process model used for controller tuning is process-dependent but, as shown elsewhere, if the process dynamic is of high order



V.M. Alfaro and R. Vilanova, *Model-Reference Robust Tuning of PID Controllers*, Advances in Industrial Control, DOI 10.1007/978-3-319-28213-8_6

usually the use of a second-order model ($a \neq 0$) for tuning allows the resulting control system to provide more performance than the resulting one using a simpler first-order model ($a = 0$).

## 6.2 Proportional Integral Control

First consider the design of a two-degree-of-freedom proportional integral (PI) control algorithm to control (6.1).

### 6.2.1 Control System Framework

Using $T_d = 0$ in (2.11), the control algorithm output reduces to a 2DoF PI:

$$u(s) = K_p \left\{ \beta r(s) - y(s) + \frac{1}{T_i s} \left[ r(s) - y(s) \right] \right\}. \tag{6.2}$$

The controller parameters to tune are $\theta_c = \left\{ K_p, T_i, \beta \right\}$.

For the PI controller (6.2), the regulatory control closed-loop transfer function target (4.16) is

$$M_{yd}^t(s) = \frac{(T_i/K_p) s N_p^+(s)}{D_M(\theta_p, \theta_{cy}, \theta_d, s)}, \tag{6.3}$$

where $D_M(\theta_p, \theta_{cy}, \theta_d, s)$ is the denominator of all the control system closed-loop transfer functions with $D_M(s = 0) = 1$. The number of closed-loop poles depends on the controlled process model order.

Using the controller (6.2) aspect applied to the set-point and (6.3) into (4.18), the servo-control transfer function target is

$$M_{yr}^t(s) = \frac{(\beta T_i s + 1) N_p^+(s)}{D_M(\theta_p, \theta_{cy}, \theta_d, s)}. \tag{6.4}$$

The application of the 2DoF PI MoReRT methodology to overdamped models is introduced in [1] and fully explained in [2].

For a second-order controlled process and a PI controller, the closed-loop control system is of third-order. Then a target third-order closed-loop denominator $D_M(\theta_p, \theta_{cy}, \theta_d, s)$ must be specified.

### 6.2.2 PI Tuning for Overdamped Closed-Loop Response Target

In order to obtain a non-oscillatory controlled variable and, as a side effect, a smooth controller output the closed-loop transfer functions targets (6.3) and (6.4) are stated such that the global control system output target $y^t(s)$ (4.7) is as follows:

$$y^t(s) = \frac{e^{-Ls}}{(\tau_c Ts + 1)(a\tau_c Ts + 1)} r(s) + \frac{(T_i/K_p)s e^{-Ls}}{(\tau_c Ts + 1)^2(a\tau_c Ts + 1)} d(s), \tag{6.5}$$

where $\tau_c$ is a dimensionless design parameter, which is an indication of the closed-loop system response speed in relation to the controlled process speed. This parameter is the one that will be used as a *soft-constraint* as mentioned. The parameter will be literately adjusted in order to get the desired robustness target. As the formulation of the target transfer function guarantees a non-oscillatory response, the purpose of adjusting $\tau_c$ will be that of getting the fastest target model that allows the desired level of robustness.

Particular cases of (6.5) are, for a first-order model ($a = 0$)

$$y^t(s) = \frac{e^{-Ls}}{\tau_c Ts + 1} r(s) + \frac{(T_i/K_p)s e^{-Ls}}{(\tau_c Ts + 1)^2} d(s), \tag{6.6}$$

and for a second-order dual-pole model ($a = 1$)

$$y^t(s) = \frac{e^{-Ls}}{(\tau_c Ts + 1)^2} r(s) + \frac{(T_i/K_p)s e^{-Ls}}{(\tau_c Ts + 1)^3} d(s). \tag{6.7}$$

Target overdamped response (6.5) combines the closed-loop characteristics considered in [3] for servo-control and the ones in [4] for regulatory control.

Usually, the design of two-degree-of-freedom controllers is performed in two steps. First, the feedback controller parameters are selected to obtain the desired regulatory control response with a target control system robustness level, and then the free parameter in the set-point controller is used to improve the servo-control response.

Here an overall cost functional that takes into account, at the same time, the regulatory control and the servo-control target responses are used for the controller design.

Using (6.5) with (4.9) and (4.10) for optimizing (4.11) the closed-loop response relative speed $\tau_c$ is adjusted to obtain a target robustness level $M_S^t$. The resulting controller normalized parameters[1] can be expressed as functions of the (6.1) controlled

---

[1] See Sect. 5.3 for the controller normalized parameters expressions.

process normalized model (5.2) parameters and the design robustness target $M_S^t$ as follows:

$$\kappa_p = \frac{a_0 + a_1\tau_L}{a_2 + a_3\tau_L + a_4{\tau_L}^2 + a_5{\tau_L}^3}, \tag{6.8}$$

$$\tau_i = \frac{b_0 + b_1\tau_L}{b_2 + b_3\tau_L + b_4{\tau_L}^2 + b_5{\tau_L}^3 + b_6{\tau_L}^4}, \tag{6.9}$$

$$\beta = c_0 + c_1\tau_L + c_2{\tau_L}^2 + c_3{\tau_L}^3. \tag{6.10}$$

The $a_i$, $b_i$, and $c_i$ constants in expressions (6.8)–(6.10), are listed in Tables 6.1, 6.2, 6.3, 6.4, 6.5, and 6.6 [2] for four robustness target levels, $M_S^t \in \{1.4,\ 1.6,\ 1.8,\ 2.0\}$; and models with six time constants ratios, $a \in \{0.0,\ 0.1,\ 0.25,\ 0.50, 0.75,\ 1.0\}$ and normalized dead-time $\tau_L$ in the range from 0.1 to 2.0.

For first-order ($a = 0$) and dual-pole ($a = 1$) models (6.8)–(6.10) provide directly the controller parameters. For second-order models with $a \notin \{0.1, 0.25, 0.50, 0.75\}$, controller parameters must be obtained by a interpolation between the two sets of parameters obtained with the two adjacent $a$ values provided.

Constants in (6.8)–(6.10) were obtained from the optimization process data using a curve fitting tool.

**Table 6.1** MoReRT constants, FOPDT models ($a = 0.0$)

| $M_S^t$ | 1.4 | 1.6 | 1.8 | 2.0 |
|---|---|---|---|---|
| $a_0$ | 0.7253 | 0.4441 | 0.5249 | 0.5930 |
| $a_1$ | 0.6505 | 0.1745 | 0.2281 | 0.2658 |
| $a_2$ | 0.002337 | 0 | 0 | 0 |
| $a_3$ | 2.143 | 1 | 1 | 1 |
| $a_4$ | 1 | 0 | 0 | 0 |
| $a_5$ | 0 | 0 | 0 | 0 |
| $b_0$ | −0.1606 | −0.09742 | 0.1530 | 0.6088 |
| $b_1$ | 47.67 | 83.72 | 115.5 | 154.9 |
| $b_2$ | 4.166 | 10.71 | 18.67 | 29.32 |
| $b_3$ | 30.23 | 51.35 | 68.28 | 88.39 |
| $b_4$ | 7.973 | 3.948 | −0.4553 | −4.346 |
| $b_5$ | −4.738 | −5.369 | −4.952 | −4.659 |
| $b_6$ | 1 | 1 | 1 | 1 |
| $c_0$ | 0.5049 | 0.4759 | 0.4706 | 0.4758 |
| $c_1$ | 0.8330 | 0.5924 | 0.4360 | 0.3267 |
| $c_2$ | −0.1034 | −0.1278 | −0.09808 | −0.07063 |
| $c_3$ | 0 | 0 | 0 | 0 |

**Table 6.2** MoReRT constants, SOPDT models with $a = 0.10$

| $M_S^t$ | 1.4 | 1.6 | 1.8 | 2.0 |
|---|---|---|---|---|
| $a_0$ | 4.264 | 5.026 | 10.54 | 12.28 |
| $a_1$ | 3.008 | 2.912 | 6.250 | 7.795 |
| $a_2$ | 0.7672 | 0.6431 | 1.058 | 1.017 |
| $a_3$ | 13.52 | 11.99 | 21.47 | 22.57 |
| $a_4$ | 2.816 | 1 | 1 | 1 |
| $a_5$ | 1 | 0 | 0 | 0 |
| $b_0$ | 2.268 | 75.12 | 17.21 | 11.33 |
| $b_1$ | 39.41 | 1426 | 265.2 | 151.1 |
| $b_2$ | 3.965 | 165.9 | 41.16 | 28.29 |
| $b_3$ | 27.77 | 1028 | 178.6 | 96.67 |
| $b_4$ | 5.123 | −110.40 | −25.83 | −16.01 |
| $b_5$ | −3.507 | 1 | 1 | 1 |
| $b_6$ | 1 | 0 | 0 | 0 |
| $c_0$ | 0.5565 | 0.5243 | 0.5123 | 0.5139 |
| $c_1$ | 0.9507 | 0.6265 | 0.4547 | 0.3259 |
| $c_2$ | −0.3226 | −0.2313 | −0.1689 | −0.1036 |
| $c_3$ | 0.0872 | 0.03721 | 0.02538 | 0.01162 |

**Table 6.3** MoReRT constants, SOPDT models with $a = 0.25$

| $M_S^t$ | 1.4 | 1.6 | 1.8 | 2.0 |
|---|---|---|---|---|
| $a_0$ | 2.533 | 6.240 | 16.12 | 14.67 |
| $a_1$ | −0.1547 | 3.418 | 9.223 | 9.476 |
| $a_2$ | 0.8599 | 1.441 | 2.857 | 2.084 |
| $a_3$ | 7.432 | 15.02 | 33.12 | 27.52 |
| $a_4$ | −2.820 | 1 | 1 | 1 |
| $a_5$ | 1 | 0 | 0 | 0 |
| $b_0$ | 2.166 | 154.5 | 17.72 | 10.72 |
| $b_1$ | 11.19 | 2042 | 203.4 | 109.2 |
| $b_2$ | 2.230 | 196.9 | 24.89 | 15.86 |
| $b_3$ | 6.897 | 1480 | 139.4 | 72.02 |
| $b_4$ | 4.012 | −152.1 | −20.97 | −12.87 |
| $b_5$ | −3.089 | 1 | 1 | 1 |
| $b_6$ | 1 | 0 | 0 | 0 |
| $c_0$ | 0.5796 | 0.5406 | 0.5151 | 0.5057 |
| $c_1$ | 1.024 | 0.6162 | 0.4748 | 0.3758 |
| $c_2$ | −0.4927 | −0.2497 | −0.2081 | −0.1633 |
| $c_3$ | 0.1773 | 0.04321 | 0.03662 | 0.02808 |

**Table 6.4** MoReRT constants, SOPDT models with $a = 0.50$

| $M_S^t$ | 1.4 | 1.6 | 1.8 | 2.0 |
|---|---|---|---|---|
| $a_0$ | 3.998 | 5.072 | 31.07 | 13.96 |
| $a_1$ | −1.784 | 2.772 | 15.29 | 8.546 |
| $a_2$ | 1.974 | 1.588 | 7.564 | 2.664 |
| $a_3$ | 9.781 | 11.72 | 58.82 | 24.52 |
| $a_4$ | −6.350 | 1 | 1 | 1 |
| $a_5$ | 1 | 0 | 0 | 0 |
| $b_0$ | 16.33 | 188.6 | 174.4 | 33.74 |
| $b_1$ | −7.025 | 2668 | 1767 | 314.9 |
| $b_2$ | 12.46 | 174.9 | 173.0 | 35.50 |
| $b_3$ | −7.889 | 1779 | 1096 | 187.3 |
| $b_4$ | 5.904 | −144.1 | −128.5 | −26.56 |
| $b_5$ | −4.141 | 1 | 1 | 1 |
| $b_6$ | 1 | 0 | 0 | 0 |
| $c_0$ | 0.4262 | 0.5252 | 0.4937 | 0.4777 |
| $c_1$ | 1.994 | 0.5520 | 0.4335 | 0.3619 |
| $c_2$ | −2.060 | −0.2216 | −0.1896 | −0.1616 |
| $c_3$ | 0.8367 | 0.03796 | 0.0330 | 0.02835 |

**Table 6.5** MoReRT constants, SOPDT models with $a = 0.75$

| $M_S^t$ | 1.4 | 1.6 | 1.8 | 2.0 |
|---|---|---|---|---|
| $a_0$ | 5.774 | 13.09 | 780.7 | 1586 |
| $a_1$ | −2.612 | 4.900 | 304.2 | 671.0 |
| $a_2$ | 3.256 | 4.764 | 214.0 | 350.6 |
| $a_3$ | 12.27 | 25.71 | 1290 | 2340 |
| $a_4$ | −7.671 | 1 | 1 | 1 |
| $a_5$ | 1 | 0 | 0 | 0 |
| $b_0$ | 20.03 | 435.4 | 136.7 | 225.6 |
| $b_1$ | −8.585 | 4154 | 1262 | 1593 |
| $b_2$ | 13.27 | 323.1 | 111.8 | 190.7 |
| $b_3$ | −7.615 | 2425 | 693.7 | 820.4 |
| $b_4$ | 5.483 | −144.7 | −70.57 | −92.06 |
| $b_5$ | −4.049 | 1 | 1 | 1 |
| $b_6$ | 1 | 0 | 0 | 0 |
| $c_0$ | 0.4223 | 0.4967 | 0.4631 | 0.4472 |
| $c_1$ | 1.705 | 0.4609 | 0.3698 | 0.3115 |
| $c_2$ | −1.759 | −0.1704 | −0.1510 | −0.1286 |
| $c_3$ | 0.7198 | 0.02748 | 0.02498 | 0.0231 |

**Table 6.6** MoReRT constants, SOPDT models with $a = 1.0$

| $M_S^t$ | 1.4 | 1.6 | 1.8 | 2.0 |
|---|---|---|---|---|
| $a_0$ | 7.163 | 26.71 | 18.65 | 521.4 |
| $a_1$ | −2.794 | 8.032 | 7.737 | 199.0 |
| $a_2$ | 4.118 | 10.02 | 5.215 | 117.7 |
| $a_3$ | 13.68 | 45.76 | 27.97 | 684.5 |
| $a_4$ | −7.551 | 1 | 1 | 1 |
| $a_5$ | 1 | 0 | 0 | 0 |
| $b_0$ | 24.23 | 778.0 | 422.3 | 531.5 |
| $b_1$ | −7.143 | 41.60 | 261.7 | 3139 |
| $b_2$ | 14.18 | 490.5 | 292.4 | 390.7 |
| $b_3$ | −6.404 | 2093 | 1242 | 1414 |
| $b_4$ | 5.820 | −88.86 | −102.4 | −135.1 |
| $b_5$ | −4.059 | 1 | 1 | 1 |
| $b_6$ | 1 | 0 | 0 | 0 |
| $c_0$ | 0.4986 | 0.4617 | 0.4307 | 0.4155 |
| $c_1$ | 0.7797 | 0.3840 | 0.3130 | 0.2716 |
| $c_2$ | −0.4881 | −0.1314 | −0.1198 | −0.1078 |
| $c_3$ | 0.1978 | 0.02011 | 0.01928 | 0.01791 |

### *Control System Robustness Verification*

One of the characteristics of the MoReRT tuning is to achieve the design control system robustness target level. The robustness obtained with (6.8) and (6.9) for first-order plus dead-time models are shown in Fig. 6.1. This figure shows for every process model normalized time constant, the controller designed to match as much as possible the specified overdamped closed-loop targets, also do achieve the specified robustness level. Here just the case for $a = 0$ is shown as an example, whereas robustness accomplishment for other models is presented in [2].

### *Evaluation of the MoReRT Tuning*

Consider the fourth-order controlled processes proposed as benchmarks in [5] and given by the transfer function:

$$P_\alpha(s) = \frac{1}{\prod_{n=0}^{3}(\alpha^n s + 1)}, \tag{6.11}$$

with $\alpha \in \{0.1, 0.25, 0.5, 1.0\}$.

Using the three-point identification procedure *123c* [6] FOPDT and SOPDT models were obtained whose parameters are listed in Table 6.7. The fact of obtaining models of different order is done because this will allow to compare the performance that can be achieved if a higher order model is used for design. This is an interesting

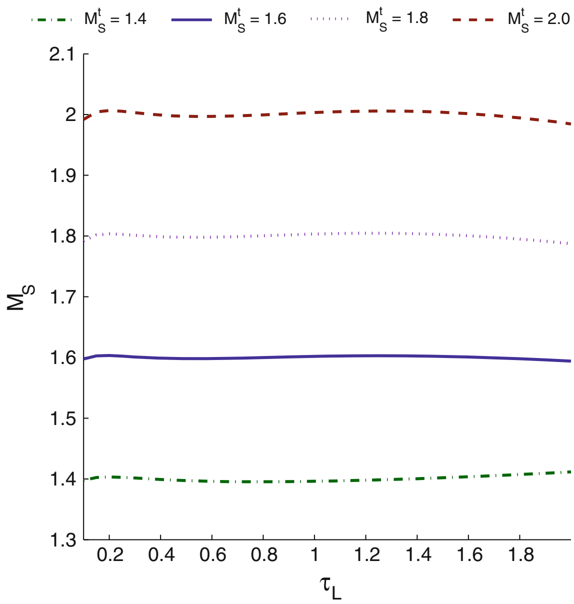



**Fig. 6.1** MoReRT PI controllers robustness accomplishment for first-order plus dead-time models

**Table 6.7** Example—$P_\alpha(s)$ FOPDT and SOPDT models

| $\alpha$ | $K$ | $T$ | $a$ | $L$ | $\tau_L$ |
|---|---|---|---|---|---|
| 0.10 | 1 | 1.003 | 0 | 0.112 | 0.112 |
| 0.25 | 1 | 1.049 | 0 | 0.298 | 0.284 |
| 0.50 | 1 | 1.247 | 0 | 0.691 | 0.554 |
| 1.0 | 1 | 2.343 | 0 | 1.860 | 0.794 |
| 0.25 | 1 | 0.987 | 0.254 | 0.086 | 0.087 |
| 0.50 | 1 | 0.876 | 0.821 | 0.277 | 0.316 |
| 1.0 | 1 | 1.487 | 1.0 | 1.110 | 0.747 |

point because the PI control literature almost concentrates on the use of FOPDT models, whereas the existence of tuning rules based on more elaborated models will allow the control system to show higher performance (while satisfying the same robustness level).

Tables 6.8 and 6.9 list the MoReRT controllers parameters for these models that result from application of tuning equations (6.8)–(6.10).

Please note that the proportional set-point weight factor $\beta$ may take values higher than 1.0, the arbitrary upper limit imposed by some manufacturers of 2DoF controllers, see Sect. 11.1. The advantages of using proportional set-point weights $\beta > 1.0$ to improve the servo-control performance when a high robust tuned control system required are shown in [7].

**Table 6.8** Example—$PI_2$ from FOPDT models

| $M_S^t$ | 1.4 | 1.6 | 1.8 | 2.0 |
|---|---|---|---|---|
| $\alpha = 0.10$ | | | | |
| $K_p$ | 3.140 | 4.152 | 4.929 | 5.576 |
| $T_i$ | 0.678 | 0.563 | 0.498 | 0.459 |
| $\beta$ | 0.597 | 0.541 | 0.518 | 0.514 |
| $\alpha = 0.50$ | | | | |
| $K_p$ | 0.725 | 0.976 | 1.175 | 1.336 |
| $T_i$ | 1.445 | 1.458 | 1.438 | 1.413 |
| $\beta$ | 0.935 | 0.765 | 0.682 | 0.635 |
| $\alpha = 1.0$ | | | | |
| $K_p$ | 0.532 | 0.734 | 0.889 | 1.013 |
| $T_i$ | 2.828 | 3.009 | 3.052 | 3.054 |
| $\beta$ | 1.101 | 0.866 | 0.755 | 0.691 |

**Table 6.9** Example—$PI_2$ from SOPDT models

| $M_S^t$ | 1.4 | 1.6 | 1.8 | 2.0 |
|---|---|---|---|---|
| $\alpha = 0.25$ | | | | |
| $K_p$ | 1.690 | 2.366 | 2.937 | 3.445 |
| $T_i$ | 1.088 | 1.015 | 0.953 | 0.911 |
| $\beta$ | 0.664 | 0.592 | 0.555 | 0.537 |
| $\alpha = 0.50$ | | | | |
| $K_p$ | 0.785 | 1.145 | 1.423 | 1.677 |
| $T_i$ | 1.395 | 1.475 | 1.497 | 1.497 |
| $\beta$ | 0.778 | 0.589 | 0.552 | 0.522 |
| $\alpha = 1.0$ | | | | |
| $K_p$ | 0.482 | 0.731 | 0.917 | 1.065 |
| $T_i$ | 2.494 | 2.882 | 3.038 | 3.117 |
| $\beta$ | 0.891 | 0.684 | 0.606 | 0.566 |

### Regulatory Control

Table 6.10 lists the control system robustness and the regulatory control performance obtained with the controllers tuned using the FOPDT models. The robustness levels ($M_S$) were obtained using the models, while the performance indices were evaluated with the original controlled processes (6.11).

Table 6.11 lists the control system robustness and the regulatory control performance obtained with the controllers tuned using the SOPDT models.

It can be seen from Table 6.10 that the robustness targets were perfectly achieved with the MoReRT controllers for all the FOPDT models.

From the standpoint of performance, the existence of a trade-off with robustness is clear. For a given model, if the control system robustness level is increased,

**Table 6.10** Example—Robustness and performance, $PI_2$ tuned from FOPDT models

| $M_S^t$ | 1.4 | 1.6 | 1.8 | 2.0 |
|---|---|---|---|---|
| $\alpha = 0.10$ | | | | |
| $M_S$ | 1.40 | 1.60 | 1.80 | 2.00 |
| $J_{ed}$ | 0.216 | 0.136 | 0.101 | 0.082 |
| $TV_{ud}$ | 1.171 | 1.312 | 1.419 | 1.514 |
| $\alpha = 0.50$ | | | | |
| $M_S$ | 1.40 | 1.60 | 1.80 | 2.00 |
| $J_{ed}$ | 2.007 | 1.495 | 1.224 | 1.058 |
| $TV_{ud}$ | 1.000 | 1.115 | 1.297 | 1.475 |
| $\alpha = 1.0$ | | | | |
| $M_S$ | 1.40 | 1.60 | 1.80 | 2.00 |
| $J_{ed}$ | 5.313 | 4.099 | 3.433 | 3.112 |
| $TV_{ud}$ | 1.000 | 1.123 | 1.343 | 1.556 |

**Table 6.11** Example—Robustness and performance, $PI_2$ tuned from SOPDT models

| $M_S^t$ | 1.4 | 1.6 | 1.8 | 2.0 |
|---|---|---|---|---|
| $\alpha = 0.25$ | | | | |
| $M_S$ | 1.40 | 1.60 | 1.80 | 2.00 |
| $J_{ed}$ | 0.644 | 0.429 | 0.325 | 0.264 |
| $TV_{ud}$ | 1.091 | 1.330 | 1.568 | 1.797 |
| $\alpha = 0.50$ | | | | |
| $M_S$ | 1.40 | 1.60 | 1.80 | 2.00 |
| $J_{ed}$ | 1.777 | 1.288 | 1.052 | 0.907 |
| $TV_{ud}$ | 1.015 | 1.248 | 1.506 | 1.795 |
| $\alpha = 1.0$ | | | | |
| $M_S$ | 1.40 | 1.60 | 1.80 | 2.00 |
| $J_{ed}$ | 5.173 | 3.942 | 3.315 | 3.051 |
| $TV_{ud}$ | 1.000 | 1.151 | 1.395 | 1.642 |

the performance decreases, in that $J_{ed}$ increases. On the controller output side, a robustness increment results in more smoothness, in that $TV_{ud}$ decreases.

For the SOPDT models, Table 6.11 reveals the perfect achievement of the design robustness level with the MoReRT controllers for all models.

For controlled processes than can be approximated with both, first- and second-order models ($\alpha \in \{0.50,\ 1.0\}$), it can be seen from Tables 6.10 and 6.11 that more regulatory control performance is obtained, for all robustness levels, if the PI controller is tuned using the second-order model.

**Table 6.12** Example—Servo-control performance, $PI_2$ tuned from FOPDT models

| $M_S^t$ | 1.4 | 1.6 | 1.8 | 2.0 |
|---|---|---|---|---|
| $\alpha = 0.10$ | | | | |
| $J_{er}$ | 0.489 | 0.394 | 0.341 | 0.305 |
| $TV_{ur}$ | 1.435 | 2.131 | 2.839 | 3.562 |
| $\alpha = 0.50$ | | | | |
| $J_{er}$ | 2.102 | 1.838 | 1.681 | 1.586 |
| $TV_{ur}$ | 0.482 | 0.733 | 1.008 | 1.286 |
| $\alpha = 1.0$ | | | | |
| $J_{er}$ | 5.029 | 4.502 | 4.191 | 4.111 |
| $TV_{ur}$ | 0.502 | 0.719 | 0.974 | 1.241 |

**Table 6.13** Example—Servo-control performance, $PI_2$ tuned from SOPDT models

| $M_S^t$ | 1.4 | 1.6 | 1.8 | 2.0 |
|---|---|---|---|---|
| $\alpha = 0.25$ | | | | |
| $J_{er}$ | 1.009 | 0.843 | 0.713 | 0.686 |
| $TV_{ur}$ | 0.829 | 1.493 | 2.229 | 3.045 |
| $\alpha = 0.50$ | | | | |
| $J_{er}$ | 2.087 | 1.894 | 1.723 | 1.612 |
| $TV_{ur}$ | 0.557 | 0.879 | 1.263 | 1.728 |
| $\alpha = 1.0$ | | | | |
| $J_{er}$ | 5.446 | 4.853 | 4.510 | 4.319 |
| $TV_{ur}$ | 0.573 | 0.734 | 0.981 | 1.252 |

### Servo-Control

The servo-control performance obtained with the controllers tuned using the FOPDT models are listed in Table 6.12. From these data the trade-off between performance and robustness, $J_{er}$ versus $M_S$, is evident for all $\alpha$'s considered. Also note that as in the regulatory control case, the higher robustness systems have smoother controller outputs.

The servo-control performance obtained with the controllers tuned using the SOPDT models are listed in Table 6.13. From these data the trade-off between performance and robustness is also evident, in that $J_{er}$ increases as the design robustness decreases (or $M_S^t$ increases).

Considering the above, for processes (6.11) with $\alpha \in \{0.25, 0.50, 1.0\}$ it is better to use the SOPDT models for the design of the 2DoF PI control system.

Figure 6.2 shows the MoReRT $PI_2$ (tuned with FOPDT model data) closed-loop responses to a 20 % set-point step change followed by a 10 % disturbance change for the controlled process with $\alpha = 0.50$.

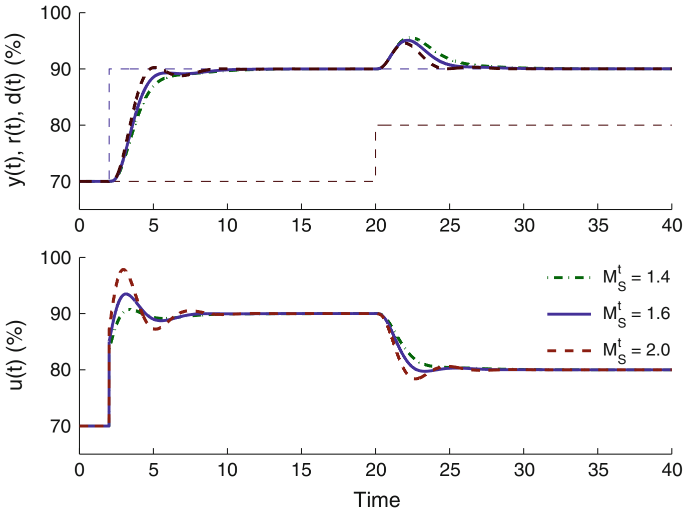



**Fig. 6.2** MoReRT PI closed-loop responses for the $\alpha = 0.50$ process

### *6.2.3 PI Tuning for Under-Damped Closed-Loop Response Target*

In order to analyze if it is possible to modify the control system performance to a load-disturbance and set-point step changes without affecting its robustness, a new global control system output target $y^t(s)$ with two under-damped dominant poles is selected and computed as [8]

$$y^t(s) = \frac{(\tau_c Ts + 1)\mathrm{e}^{-Ls}}{(\tau_c^2 T^2 s^2 + 2\zeta \tau_c Ts + 1)(a\tau_c Ts + 1)} r(s) + \frac{(T_i/K_p)s\mathrm{e}^{-Ls}}{(\tau_c^2 T^2 s^2 + 2\zeta \tau_c Ts + 1)(a\tau_c Ts + 1)} d(s). \tag{6.12}$$

Now the control system has two design parameters, the *closed-loop poles relative speed* $\tau_c$ and their *damping ratio* $\zeta$. The effect of the damping ratio $\zeta$ over the closed-loop control systems performance and control effort characteristics is analyzed in [8]. Again, $\tau_c$ will be adjusted with the purpose of matching the desired robustness.

The regulatory control response indices evaluated are the integrated absolute error ($J_{ed}$), the controller output total variation ($TV_{ud}$), the maximum error ($E_{max}$), the time to reach the maximum error ($t_{max}$), and the settling time ($t_{5\,\%Emax}$). For the servo-control response the indices evaluated are the integrated absolute error ($J_{er}$), the controller output total variation ($TV_{ur}$), the rise time ($t_r$), the control effort maximum value ($U_{maxr}$), the controller output instant change ($\Delta U_{0r}$), and the settling time ($t_{5\,\%\Delta y}$). These indices are defined in [8].

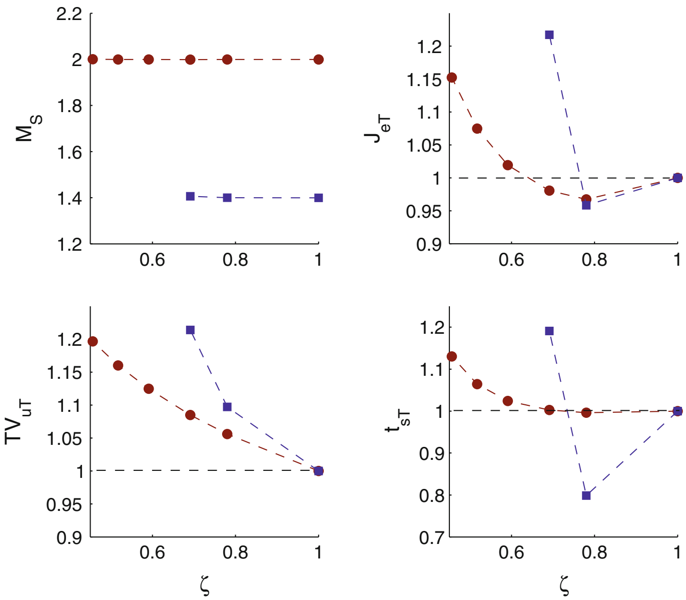



**Fig. 6.3** Damping ratio $\zeta$ effect over total performance, control effort variation, and settling time for model with $a = 0.75$ and $\tau_L = 0.50$

The analysis made use of damping ratios $\zeta$ in the range from 1.0 to 0.456 and robustness levels $M_S$ from 2.0 to 1.40. Figure 6.3 shows the indices obtained for the model with ($a = 0.75$, $\tau_L = 0.5$). Case analysis is presented in [8].

The analysis shows that, for same robustness, all the controllers obtained with the original non-oscillatory response target ($\zeta = 1.0$) provide the smoothest control efforts with an integrated absolute error performance similar to the lower obtainable value for the corresponding robustness level target.

An improvement in the control system performance (integrated absolute error and settling time), specially for the servo-control, may be obtained if the closed-loop transfer function poles design damping ratio is selected in the range from 0.7 to 0.8 but adversely affecting the control effort characteristics.

On the basis of the analysis for overdamped controlled processes, it is recommended to use $\zeta = 0.8$ for MoReRT 2DoF PI design when under-damped target responses are specified.

***Evaluation of MoReRT Controllers***

As an example of the effect of the damping ratio $\zeta$ over the control systems performance and control effort characteristics, consider the second-order plus dead-time normalized model given by

$$P(s) = \frac{e^{-0.5s}}{(s+1)(0.75s+1)}. \tag{6.13}$$

**Table 6.14** Overdamped model example, $M_S^t = 2.0$, effect of $\zeta$ over the control system performance and control effort

| $\zeta$ | 1.0 | 0.78 | 0.591 | 0.456 |
|---|---|---|---|---|
| $J_{ed}$ | 1.406 | 1.410 | 1.495 | 1.697 |
| $TV_{ud}$ | 1.627 | 1.648 | 1.683 | 1.737 |
| $J_{er}$ | 2.158 | 2.038 | 2.139 | 2.411 |
| $TV_{ur}$ | 2.111 | 2.345 | 2.590 | 2.774 |
| $U_{maxr}$ | 1.327 | 1.442 | 1.560 | 1.642 |
| $\Delta U_{0r}$ | 0.726 | 0.811 | 0.885 | 0.907 |
| $J_{eT}$ | 3.564 | 3.448 | 3.634 | 4.108 |
| $TV_{uT}$ | 3.738 | 3.993 | 4.273 | 4.511 |

The regulatory and servo-control performance $J_e$ (measured with the integrated absolute error) and control effort total variation $TV_u$, maximum value $U_{max}$, and instant change $\Delta U_0$ for a robustness level target $M_S^t = 2.0$ and four damping ratios are listed in Table 6.14.

The table data confirms that, keeping the control system robustness constant, the servo-control performance can be improved, obtaining lower $J_{er}$, if the closed-loop poles damping ratio is decreased a bit, but affecting the control effort characteristics, $TV_{ur}$, $U_{maxr}$, and $\Delta U_{0r}$. It also shows that very low closed-loop poles damping ratios deteriorate all control system characteristics.

Control system responses for $M_s^t = 2.0$ and four damping ratios are shown in Fig. 6.4.

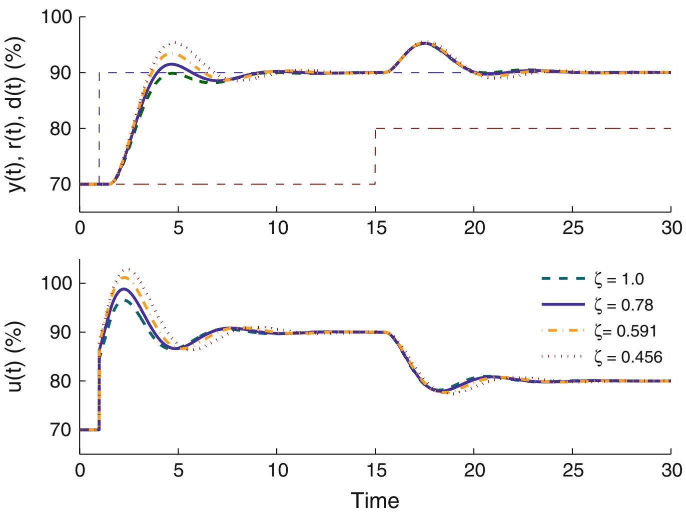



**Fig. 6.4** Overdamped model example, control system responses ($M_S^t = 2.0$)

Further evaluations of MoreRT tuning (6.8)–(6.10) and comparison with other robust tuning methods for overdamped models are presented in [2].

## 6.3 Proportional Integral Derivative Control

The MoReRT design methodology is used here for tuning proportional integral derivative (PID) controllers to control overdamped processes. When moving to PID controllers, we face the problem of choosing the PID configuration. In this case, the filtered PID implementations will be used in order to derive the tuning relations. The Standard 2DoF PID will not be considered because when doing the optimizations, it turns out that the PID tends to a filtered ideal PID.

In fact, first a Standard 2DoF PID (2.12) was considered. The $PID_2$ controller parameters are $\theta_c = \{K_p, T_i, T_d, \alpha, \beta, \gamma = 0\}$. To take the derivative filter constant $\alpha$ as a tunable parameter, a case study is made for process with $0.1 \leq \tau_L \leq 2.0$ and robustness levels $M_S^t \in \{2.0, 1.6, 1.4\}$. What it was found during this study is that in all cases, ideally, $\alpha \to 0$, this is, the Standard PID (with filter) tends to be an Ideal PID (with a non-proper transfer function). The controller high frequency gain is $1/\alpha$ then, if $\alpha$ is very low the measurement noise, if any, will be amplified. It is also found that the control system performance is not very sensitive to $\alpha$ if it is restricted to the values usually found in commercial controllers ($0.05 \leq \alpha \leq 0.20$). Considered the above the use of the default $\alpha = 0.1$ is recommended in this case.

### 6.3.1 2DoF Ideal PID with Filter

It is shown in Sect. 2.3 that the two-degree-of-freedom proportional integral derivative controller with filter $PID_{2F}$ given by (2.21) is a more general control algorithm than the Standard PID. Then to control an overdamped controlled process represented by a first-order plus dead-time model a $PID_{2F}$ controller is used [9].

To take into account the $PID_{2F}$ filter, the global control system target response (4.7) is under-damped and stated as follows:

$$y^t(s) = \frac{(\beta^* T_i^* s + 1)(T_f s + 1)\mathrm{e}^{-Ls}}{\tau_c^2 T^2 s^2 + 2\zeta \tau_c T s + 1} r(s) + \frac{(T_i^*/K_p^*) s (T_f s + 1)\mathrm{e}^{-Ls}}{\tau_c^2 T^2 s^2 + 2\zeta \tau_c T s + 1} d(s), \tag{6.14}$$

where $\zeta$ and $\tau_c$ are the design parameters, being $\tau_c$ adjusted for robustness purposes.

To select the target closed-loop poles damping ratio $\zeta$, a performance (integrated absolute error) to control effort total variation trade-off analysis is made. It is found that $\zeta = 0.70$ provides a good balance between these two indices [9]. Therefore, this parameter becomes fixed and the only parameter that is adjusted and directly related to robustness is $\tau_c$.

For the design FOPDT models with normalized dead-time $\tau_L$ between 0.1 and 2.0 and closed-loop target robustness $M_S^t \in \{1.4, 1.6, 1.8, 2.0\}$ are used.

The $PID_{2F}$ controller parameters $\theta_c^* = \{K_p^*, T_i^*, T_d^*, T_f, \beta^*, \gamma^* = 0\}$ are obtained optimizing the overall cost functional defined in Sect. 4.2

$$J_T^o \doteq J_T(\theta_p, \theta_c^o, \theta_d) = \min_{\theta_c} J_T(\theta_p, \theta_c^*, \theta_d). \tag{6.15}$$

The design parameter $\tau_c$ ($\theta_d = \{\tau_c, \zeta = 0.70\}$) is selected in such a way that the control system robustness matches a maximum sensitivity target value $M_S^t$.

The optimized controllers parameters are used to fit the tuning equations for the four robustness levels considered given by

$$\kappa_p^* = a_0 + a_1 \tau_L^{a_2}, \tag{6.16}$$

$$\tau_i^* = b_0 + b_1 \tau_L^{b_2}, \tag{6.17}$$

$$\tau_d^* = c_1 \tau_L^{c_2}, \tag{6.18}$$

$$\tau_f^* = d_0 + d_1 \tau_L^{d_2}, \tag{6.19}$$

$$\beta^* = 0, \tag{6.20}$$

$$\gamma^* = 0. \tag{6.21}$$

In this section just the case $a = 0$ has been considered. To deal with SOPTD process models can be done exactly along the same lines as here. All the cases are not reproduced here just because not being excessively repetitive. When dealing in the next section with a more generic formulation of PID controller, say the one that includes two input filters, application to specific SOPTD process models will be provided, showing the advantage that can be achieved if higher order models are allowed for controller design.

Constants $a_i$, $b_i$, $c_i$, and $d_i$ for (6.16)–(6.19) are listed in Table 6.15 [9].

***Evaluation of MoReRT Controllers***

Consider the fourth-order controlled process proposed as benchmark in [5] and given by the transfer function:

$$P_e(s) = \frac{1}{(s+1)(0.5s+1)(0.25s+1)(0.125s+1)}. \tag{6.22}$$

Using a two-point identification procedure, the parameters of a FOPDT model approximation of (6.22) are: $K = 1$, $T = 1.247$, and $L = 0.691$ ($\tau_L = 0.554$).

Controllers parameters, robustness ($M_S^m$, with the model), performance (measured with the integrated absolute error, $J_{ed}$ and $J_{er}$), and control effort total variation ($TV_{ud}$ and $TV_{ur}$) to unitary step changes of the control systems obtained with proposed tuning are listed in Table 6.16.

**Table 6.15** Constants for tuning relations (6.16)–(6.19)

| $M_S^t$ | 2.0 | 1.8 | 1.6 | 1.4 |
|---|---|---|---|---|
| $a_0$ | 0.3745 | 0.3067 | 0.2458 | 0.1047 |
| $a_1$ | 0.6237 | 0.6719 | 0.5753 | 0.4553 |
| $a_2$ | −0.8575 | −0.9161 | −0.8997 | −0.7672 |
| $b_0$ | 0.0825 | −0.336 | −0.4981 | 0.4383 |
| $b_1$ | 1.199 | 1.639 | 1.874 | 0.9669 |
| $b_2$ | 0.5727 | 0.3507 | 0.2744 | 0.5062 |
| $c_1$ | 0.2948 | 0.3091 | 0.3114 | 0.2997 |
| $c_2$ | 0.9495 | 0.9471 | 0.9327 | 0.9075 |
| $d_0$ | 0.06395 | 0 | 0.01938 | 0.1074 |
| $d_1$ | 0.2104 | 0.08934 | 0.0864 | 0.1868 |
| $d_2$ | 1.137 | 0.4708 | 0.6525 | 1.641 |

**Table 6.16** MoReRT controller parameters and performance

| $M_S^t$ | 2.0 | 1.6 | 1.4 |
|---|---|---|---|
| $K_p^*$ | 1.409 | 1.225 | 0.821 |
| $T_i^*$ | 1.169 | 1.366 | 1.440 |
| $T_d^*$ | 0.210 | 0.224 | 0.219 |
| $T_f$ | 0.214 | 0.098 | 0.222 |
| $\beta^*$ | 0 | 0 | 0 |
| $\gamma^*$ | 0 | 0 | 0 |
| $M_S^m$ | 2.001 | 1.599 | 1.397 |
| $J_{ed}$ | 1.113 | 1.116 | 1.754 |
| $TV_{ud}$ | 1.742 | 1.173 | 1.021 |
| $J_{er}$ | 2.171 | 2.383 | 2.972 |
| $TV_{ur}$ | 2.117 | 1.209 | 1.008 |
| $K^\infty$ | 1.383 | 2.813 | 0.8075 |

In Table 6.16 $K^\infty$ is the $PID_{2F}$ controller high frequency gain given by

$$K^\infty = \frac{K_p K T_d^*}{T_f}. \tag{6.23}$$

Control system outputs to a 20 % step set-point change followed by a 10 % step load-disturbance change are shown in Fig. 6.5.

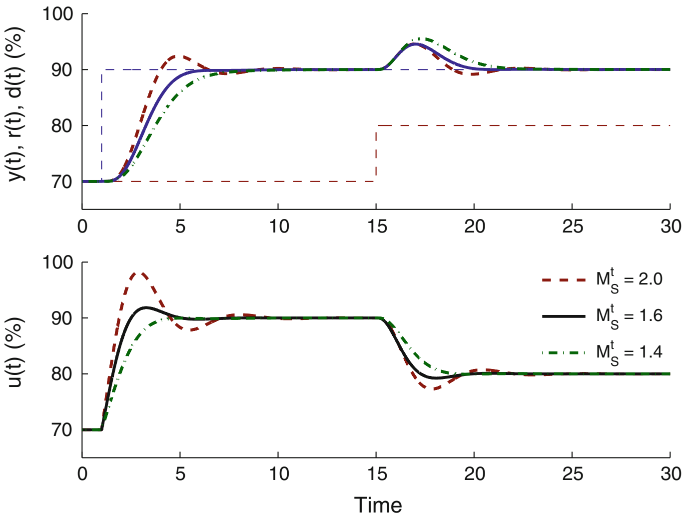



**Fig. 6.5** MoReRT example—Four-order overdamped process, $PID_{2F}$ responses

### 6.3.2 2DoF Ideal Parallel PID with Two Input Filters

Now a two-degree-of-freedom ideal parallel PID controller is aggregated with two input filters as described in Sect. 2.4.

From (2.63) the controller output signal is

$$u(s) = \left(K_p + \frac{K_i}{s}\right) F_r(s)r(s) - \left(K_p + \frac{K_i}{s} + K_d s\right) F_y(s)y(s), \tag{6.24}$$

where the filters transfer functions are

$$F_r(s) = \frac{\sigma T_r s + 1}{(T_r s + 1)^2}, \tag{6.25}$$

and

$$F_{yPI}(s) = \frac{1}{D_{fy}(s)} = \frac{1}{T_f s + 1}, \tag{6.26}$$

$$F_{yPID}(s) = \frac{1}{D_{fy}(s)} = \frac{1}{T_f^2/2s^2 + T_f s + 1}. \tag{6.27}$$

In this case the controller design is made in two steps.

First to obtain the feedback controller and filter parameters, the regulatory control target closed-loop transfer function (4.15) is expressed as

$$M_{yd}^t(s) = \frac{(1/K_i)sD_{fy}(s)N_p^+(s)}{D_{cy}(s)D_p(s) + N_{cy}(s)N_p^-(s)N_p^+(s)}, \quad (6.28)$$

and the regulatory control model-reference response cost functional

$$J_d(\theta_p, \theta_{cy}, \theta_d) \doteq \int_0^\infty \left[y_d^t(\theta_p, \theta_{cy}, \theta_d, t) - y_d(\theta_p, \theta_{cy}, t)\right]^2 \mathrm{d}t, \quad (6.29)$$

is optimized.

Second once the feedback controller and filter are obtained, the set-point filter parameters are designed using the servo-control closed-loop transfer function

$$M_{yr}(s) = \frac{(\sigma T_r s + 1)[(K_p^o/K_i^o)s + 1]D_{fy}^o(s)N_p^+(s)}{(T_r s + 1)^2[D_{cy}^o(s)D_p(s) + N_{cy}^o(s)N_p^-(s)N_p^+(s)]}. \quad (6.30)$$

The optimality of the servo-control step response may be obtained by selecting the set-point filter parameters ($\sigma$, $T_r$) to optimize the integrated absolute error (IAE) given by the cost functional

$$J_{er} \doteq \int_0^\infty |r(t) - y_r(t)|\,\mathrm{d}t, \quad (6.31)$$

the integrated absolute control effort (IAU) given by

$$J_{ur} \doteq \int_0^\infty |u_r(t) - u_r(\infty)|\,\mathrm{d}t, \quad (6.32)$$

or other suitable cost functional.

It is also possible to design the set-point filter selecting its time constant $T_r = K_p^o/T_i^o$ in order to made a zero pole cancelation reducing (6.30) to

$$M_{yr}(s) = \frac{(\sigma K_p^o/T_i^o s + 1)D_{fy}^o(s)N_p^+(s)}{(K_p^o/T_i^o s + 1)[D_{cy}^o(s)D_p(s) + N_{cy}^o(s)N_p^-(s)N_p^+(s)]}, \quad (6.33)$$

and obtaining the remaining filter parameter optimizing (6.31) or (6.32).

Note that in this case the shift of paradigm regarding the design statement, instead of considering a joint functional that takes everything into account in a single shot, here a two-step design is formulated. The reason is that having the two input filters there are more degrees of freedom that affect the control system. On this respect, primary interest is of determining the feedback properties, regulation, and robustness. Therefore the regulatory functional including the process output filter are first considered. Once the closed-loop properties are ensured, we can proceed with the adjustment of the reference filter in order to *design* de command signal that enters the feedback loop in such a way that a secondary goal is targeted. Although usually of secondary interest for process control, in the event of a sporadic set-point step change, it is important to have a fast response with low overshoot and without an abrupt change or high variations of the controller output signal. On this respect, we can manipulate the command signal in order to take care of the control signal or the tracking error. As an example, in the next section, this two-step procedure is conducted by showing the final tuning relations that arise in case of optimizing the set-point filter parameters for (6.31) or (6.32).

### *Performance Robustness Analysis*

In order to reduce the number of design parameters, the influence of the closed-loop poles damping ratio over the regulatory control performance is analyzed [10]. Remember that one of the goals when defining the target transfer functions for the closed-loop dynamics, was that of getting non-oscilatory responses for a smooth control action. More complex target transfer function models allow more rich dynamics with small and smooth oscillations. the effect of allowing such oscillations was analyzed in [10]. From the analysis, it is found that allowing some small oscillation at the control system output is possible to improve the regulatory control performance but with a deterioration of the control effort smoothness. A good balance between the regulatory control performance and the controller output total variation is obtained for damping ratios in the range from 0.70 to 0.80. Then for PI controllers, tuning $\zeta = 0.8$ is selected and $\zeta = 0.70$ for the PID.

It is also found that for a given controlled process model and same target robustness level, the PID controllers outperforms the PI in all aspects but the control effort variation.

### *$PID_{2IF}$ Controller Robust Tuning*

Due to normalized overdamped controlled process models with $a = 0$ (FOPDT) has only one parameter ($\tau_L$), controller normalized parameters can be found as function of $\tau_L$.

For first-order plus dead-time models with $0.1 \le \tau_L \le 2.0$ the PID controller and feedback filter parameters for an intermediate robustness ($M_S^t = 1.6$) can be obtained with following relations:

$$\kappa_p = 0.2954 + 0.5065\ \tau_L^{-0.9805}, \tag{6.34}$$

$$\kappa_i = \frac{-0.04612 + 0.6407\ \tau_L}{0.005159 - 0.13885\ \tau_L + \tau_L^2}, \tag{6.35}$$

$$\kappa_d = 0.2874 + 0.04719\ \tau_L^{0.9134}, \tag{6.36}$$

$$\tau_f = -0.01613 + 0.1233\ \tau_L^{0.4922}, \tag{6.37}$$

$$\gamma = 0. \tag{6.38}$$

The set-point filter parameters are given by the relations:

$$\tau_r = \frac{-0.3141 + 18.66\ \tau_L}{2.347 + 18.77\ \tau_L - 6.415\ \tau_L^2 + \tau_L^3}, \qquad (6.39)$$
$$\sigma = 1.042 + 0.3809\ \tau_L^{0.4615},$$

if the integrated absolute error (6.31) is optimized, or by following relations:

$$\tau_r = \frac{-0.3141 + 18.66\ \tau_L}{2.347 + 18.77\ \tau_L - 6.415\ \tau_L^2 + \tau_L^3}, \qquad (6.40)$$
$$\sigma = -0.7676 + 1.828\ \tau_L^{0.2382},$$

if is of interest to optimize the integrated absolute control effort (6.32).

For second-order plus dead-time models ($a \neq 0$) the controller normalized parameters depend on the two normalized model parameters and on the robustness level ($\hat{\theta}_c = \{h_i(a, \tau_L; M_S^t)\}$); therefore several sets of tuning equations are needed to consider different model time constants ratio $a$ and target $M_S^t$ robustness levels.

For these models the direct application of the proposed MoReRT procedure is recommended. At this point a word must be given to the fact of tuning a controller by a direct application of the methodology for each specific and concrete case or, instead, to have a tuning rule. When dealing with higher order processes and controllers with four or five parameters (as the case of the $PID_{2IF}$) constructing the tuning rules on the basis of fitting the controller parameters behavior can be a problem because of the numerical sensitivity of the controller parameters value. Instead, the possibility of doing efficient and low time consuming optimizations, allow for getting the controller parameters *on-demand*.

### 6.3.3 Evaluation of MoReRT Controllers

For comparison purposes consider the fourth-order controlled process given by the benchmark [5] transfer function:

$$P(s) = \frac{1}{(s+1)(0.5s+1)(0.25s+1)(0.125s+1)}. \qquad (6.41)$$

For controller tuning from $P(s)$ reaction curve following FOPDT and SOPDT models are obtained

$$P_1(s) = \frac{e^{-0.691s}}{1.247s+1},\ \tau_L = 0.554, \qquad (6.42)$$

**Table 6.17** $PI(D)_{2IF}$ controllers parameters

| $M_S^t$ | $PI_{2IF}$ (FOPDT) | $PI_{2IF}$ (SOPDT) | $PID_{2IF}$ (FOPDT) | $PID_{2IF}$ (SOPDT) | $PID_{2IF}$ (SOPDT) |
|---|---|---|---|---|---|
| | 1.6 | 1.6 | 1.6 | 1.6 | 2.0 |
| $K_p$ | 0.802 | 0.916 | 1.197 | 2.345 | 3.374 |
| $K_i$ | 0.629 | 0.672 | 1.034 | 1.841 | 2.873 |
| $K_d$ | 0.0 | 0.0 | 0.392 | 0.958 | 1.387 |
| $T_f$ | 0.117 | 0.093 | 0.096 | 0.067 | 0.054 |
| $T_r$[a] | 1.275 | 1.363 | 1.158 | 1.274 | 1.174 |
| $\sigma^{e}$[b] | 1.581 | 1.542 | 1.343 | 1.212 | 1.155 |
| $\sigma^{u}$[c] | 1.146 | 1.201 | 0.836 | 0.647 | 0.446 |
| $\gamma$ | 0 | 0 | 0 | 0 | 0 |

[a] $T_r = K_p/K_i$, [b] using $J_{er}$ (6.31), [c] using $J_{ur}$ (6.32)

$$P_2(s) = \frac{e^{-0.277s}}{(0.876s + 1)(0.719s + 1)}, \quad a = 0.821, \, \tau_L = 0.316. \tag{6.43}$$

The $PI_{2IF}$ and $PID_{2IF}$ controllers parameters using the FOPDT and the SOPDT models for a robustness level $M_S^t = 1.6$ are listed in Table 6.17. The obtained regulatory and servo-control performance indices are listed in Table 6.18. These tables also include $PID_{2IF}$ controller parameters and performance for $M_S^t = 2.0$.

From Table 6.18 it is seen that the $PID_{2IF}$ tuned with the SOPDT model outperforms all other controllers.

It also can be seen that, as designed, the $PI_{2IF}$ and $PID_{2IF}$ controllers have no abrupt controller output changes ($\Delta u_0 = 0$) and high frequency gain $K^\infty = 0$.

Figure 6.6 shows the control system responses to a 20 % set-point step change followed by a 10 % load-disturbance step change.

## Chapter Remarks

The model-reference robust tuning (MoReRT) design is applied to obtain tuning relations for 2DoF PI controllers ($PI_2$) and models representing overdamped dynamics found in industrial processes.

For stable processes represented by first- and second-order plus dead-time (FOPDT, SOPDT) models tuning relations are obtained using closed-loop overdamped responses targets. These tuning relations guarantee the accomplishment of four robustness levels for models with normalized dead-times $\tau_L$ in the range from 0.1 to 2.0.

The four robustness design levels range from the normal minimum corresponding to $M_S^t = 2.0$ to a high robustness design with $M_S^t = 1.4$. Performance comparison analysis shows that for a high-order controlled process more performance is obtained with $PI_2$ controllers if they are tuned using a process SOPDT model approximation in lieu of a simpler FOPDT model.

**Table 6.18** $PI(D)_{2IF}$ controllers performance

| | $PI_{2IF}$ (FOPDT) | $PI_{2IF}$ (SOPDT) | $PID_{2IF}$ (FOPDT) | $PID_{2IF}$ (SOPDT) | $PID_{2IF}$ (SOPDT) |
|---|---|---|---|---|---|
| $M_S$ | 1.60 | 1.60 | 1.60 | 1.60 | 2.00 |
| $K^{\infty}$ | 0 | 0 | 0 | 0 | 0 |
| $J_{ed}$ | 1.590 | 1.488 | 1.095 | 0.574 | 0.371 |
| $TV_{ud}$ | 1.172 | 1.207 | 1.223 | 1.260 | 1.577 |
| $E_{max}$ | 0.561 | 0.538 | 0.437 | 0.302 | 0.240 |
| $t_{Emax}$ | 2.290 | 2.220 | 2.000 | 1.610 | 1.415 |
| $t_{s5\,\%Emax}$ | 5.485 | 5.340 | 6.700 | 3.825 | 3.160 |
| $J_{er}$ [a] | 2.352 | 2.266 | 2.215 | 1.813 | 1.569 |
| $TV_{ur}$ [a] | 1.499 | 1.519 | 1.728 | 1.923 | 2.596 |
| $U_{max}$ [a] | 1.229 | 1.229 | 1.338 | 1.455 | 1.769 |
| $t_r$ [a] | 2.080 | 2.050 | 1.750 | 1.520 | 1.265 |
| $t_{s5\,\%\Delta y}$ [a] | 5.740 | 5.115 | 5.890 | 4.750 | 4.080 |
| $\Delta u_0$ | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| $J_{er}$ [b] | 2.562 | 2.484 | 2.366 | 2.200 | 2.119 |
| $TV_{ur}$ [b] | 1.211 | 1.263 | 1.320 | 1.174 | 1.272 |
| $U_{max}$ [b] | 1.084 | 1.099 | 1.148 | 1.083 | 1.126 |
| $t_r$ [b] | 2.650 | 2.530 | 2.295 | 2.455 | 2.440 |
| $t_{s5\,\%\Delta y}$ | 4.315 | 4.160 | 3.745 | 4.045 | 4.025 |

[a]Using $\sigma^e$, [b]using $\sigma^u$

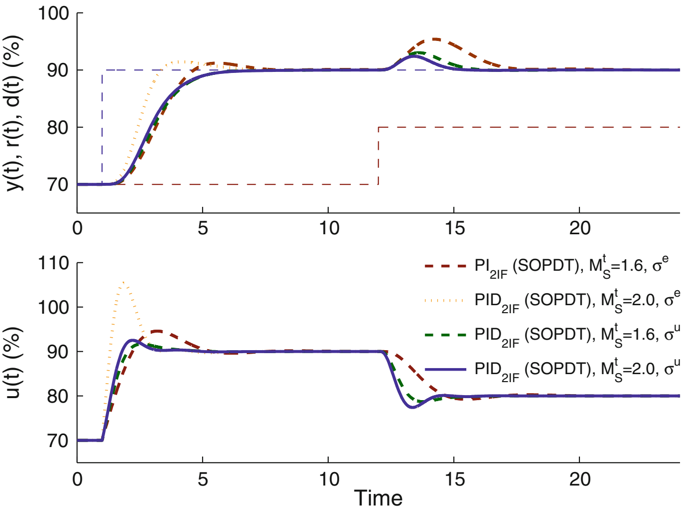



**Fig. 6.6** MoReRT $PI(D)_{2IF}$ control systems responses

It is also shown that an improvement of the control system performance, measured with the integrated absolute error, can be obtained if the design is made using closed-loop transfer functions targets with two dominant under-damped poles with damping rations in the range from 0.7 to 0.8. However, the reduction of the dominant poles damping ratio increases the control effort total variation.

The use of ideal parallel PID controllers aggregated with two input filters allows to obtain control systems with two additional desirable characteristics: no controller output abrupt changes and high frequency roll-off (noise attenuation).

## References


1. Alfaro, V.M., Vilanova, R.: Performance/robustness trade-off design framework for 2DoF PI controllers. Stud. Inf. Control **21**(1), 73–83 (2012)
2. Alfaro, V.M., Vilanova, R.: Model-reference robust tuning of 2DoF PI controllers for first- and second-order plus dead-time processes. J. Process Control **22**, 359–374 (2012)
3. Martin, J., Corripio, A.B., Smith, C.L.: Controller tuning from simple process models. Instrum. Technol. **22**(12), 39–44 (1975)
4. Chen, D., Seborg, D.: PI/PID controller desing based on direct synthesis and disturbance rejection. Ind. Eng. Chem. Res. **41**, 4807–4822 (2002)
5. Åström, K.J., Hägglund, T.: Benchmark systems for PID control. In: IFAC Digital Control: Pas, Present and Future of PID Control (PID'00), 5–7 April, Terrasa, Spain (2000)
6. Alfaro, V.M.: Low-order models identification from the process reaction curve. Ciencia y Tecnología (Costa Rica) **24(2)**, 197–216 (2006) (in Spanish). http://revista-ciencia-tecnologia.ucr.ac.cr/index.php/ciencia-tecnologia/article/view/35
7. Alfaro, V.M., Vilanova, R., Arrieta, O.: Considerations on set-point weight choice for 2-DoF PID controllers. In: IFAC International Symposium on Advanced Control on Chemical Process (ADCHEM 2009), 12–15 July, Istanbul, Turkey (2009)
8. Alfaro, V.M., Vilanova, R.: Performance analysis of model-reference robust tuned 2DoF PID controllers for over damped processes. In: IEEE 20th Mediterranean Conference on Control and Automation (MED 2012), July 3–6, Barcelona, Spain (2012)
9. Alfaro, V.M., Vilanova, R.: Model reference based robust tuning of five-parameter 2DoF PID controllers for first-order plus dead-time models. In: European Control Conference (ECC 2013), July 17–19, Zürich, Switzerland (2013)
10. Alfaro, V.M., Vilanova, R.: Performance and robustness considerations for tuning of proportional integral/proportional integral derivative controllers with two input filters. Ind. Eng. Chem. Res. **52**, 18287–18302 (2013)

# Chapter 7
# MoReRT Control of Inverse Response Processes

The inverse response characteristic, that is, the controlled process initial response to a step change is in the opposite direction to that of the steady-state direction, originated from two parallel competing dynamics can be found in industrial processes such as distillation columns and chemical reactors. This non-minimum phase characteristics impose severe limits to the achievable closed-loop control system robustness.

The application of the proposed MoReRT tuning procedure to inverse response processes is presented in [1–3].

## 7.1 Introduction

The transfer function of these kinds of processes has one zero or an odd number of zeros in the open right-half plane (RHP). It is common in the literature of process control to assume a second-order transfer function with one RHP-zero. This is the so-called second-order plus right-half plane zero (SOPRHPZ) model (5.3) rewritten here as

$$P(s) = \frac{K(-bTs + 1)}{(Ts + 1)(aTs + 1)}, \ \theta_p = \{K, T, a, b\}\,. \tag{7.1}$$

The reasons of using this model are that it contains the essential characteristics of inverse response processes and it can be adopted to model higher order processes. A process that can be described by a SOPRHPZ model constitutes one example of a non-minimum-phase system (a process with dead-time is another example). The essential characteristic of such processes is that they do not have the smallest phase lag that is possible for processes with the same gain. This non-minimum-phase characteristic introduces essential limitations in terms of achievable output performance. The difficulties associated with the control of this kind of processes become bigger when the corresponding RHP-zero approaches the origin. An important limitation due to the presence of RHP-zeros is the high-gain instability.



V.M. Alfaro and R. Vilanova, *Model-Reference Robust Tuning of PID Controllers*,
Advances in Industrial Control, DOI 10.1007/978-3-319-28213-8_7

Note that in this case the process model already has four parameters. In order to keep complexity under reasonable limits, no dead-time is assumed here as it will introduce an extra parameter therefore making the corresponding analysis really difficult and messy.

## 7.2 Proportional Integral Control

For the SOPRHPZ model, the PI control system is of third-order and the global control system output target $y^t(s)$ (4.7) is computed as

$$y^t(s) = \frac{-bTs + 1}{(\tau_c Ts + 1)(a\tau_c Ts + 1)} r(s) + \frac{(T_i/K_p)s(-bTs + 1)}{(\tau_c Ts + 1)^2(a\tau_c Ts + 1)} d(s), \tag{7.2}$$

where $\tau_c$ is the dimensionless design parameter.

The controller-normalized parameters $(\kappa_p, \tau_i, \beta)$ depend on the two model dimensionless parameters $a$ and $b$ and are obtained for right-half plane zero relative positions $b$ in the range from 0.25 to 4.0 and time constants ratios $a$ from 0.1 to 1.0.

It is to be noted that the position of the right-half plane zero affects the robustness level that may be achieved. Roughly $M_S = 2.0$ may be obtained up to $b \approx 4.0$, $M_S = 1.8$ up to $b \approx 3.5$, $M_S = 1.6$ up to $b \approx 2.25$ and $M_S = 1.4$ only up to $b \approx 1.25$. However to top it all, for higher values in above $b$ ranges the controller gains turn to be very low, with up to a 20 : 1 ratio between the highest and lowest values. Then, to obtain the controller-tuning relations, the zero relative position $b$ was selected from 0.25 to 1.0, 1.5, 2.0, and 2.5 for $M_S^t = 1.4$, 1.6, 1.8, and 2.0, respectively.

The $PI_2$ controller parameters obtained are used to fit the MoReRT equations for the normalized controller parameters and the proportional set-point weight for each one of the five model time constants ratios $a$ considered, given by

$$\kappa_p = a_0 + a_1 b^{a_2}, \tag{7.3}$$

$$\tau_i = \frac{b_0 + b_1 b}{b_2 + b_3 b + b_4 b^2 + b_5 b^3 + b_6 b^4}, \tag{7.4}$$

$$\beta = c_0 + c_1 b + c_2 b^2 + c_3 b^3. \tag{7.5}$$

Constants $a_i$, $b_i$ and $c_i$ for expressions (7.3)–(7.5) are listed in Tables 7.1, 7.2, 7.3, 7.4 and 7.5 [1] for four robustness target levels ($M_S^t \in \{1.4,\ 1.6,\ 1.8,\ 2.0\}$) and five model time constants ratios ($a \in \{0.1,\ 0.25,\ 0.5,\ 0, 75,\ 1.0\}$).

Tuning equations (7.3)–(7.5) are valid only for models with the right-half plane zero relative position $b$ in the ranges listed in Table 7.6.

**Table 7.1** MoReRT constants, SOPRHPZ models with $a = 0.1$

| $M_S^t$ | 1.4 | 1.6 | 1.8 | 2.0 |
|---|---|---|---|---|
| $a_0$ | −0.2695 | −0.1025 | −0.06981 | −0.06552 |
| $a_1$ | 0.5152 | 0.4652 | 0.5104 | 0.5677 |
| $a_2$ | −0.6464 | −0.8149 | −0.8569 | −0.8571 |
| $b_0$ | 0.7881 | −1.265 | −2.702 | −0.1015 |
| $b_1$ | −0.3003 | 12.93 | 92.63 | 8.507 |
| $b_2$ | 0.8813 | −0.7519 | 3.644 | −0.1015 |
| $b_3$ | −1.358 | 10.5 | 66.85 | 6.066 |
| $b_4$ | 2.784 | 0.8086 | 12.57 | 1 |
| $b_5$ | −2.728 | 1 | 1 | 0 |
| $b_6$ | 1 | 0 | 0 | 0 |
| $c_0$ | 0.3596 | 0.1708 | 0.464 | 0.4845 |
| $c_1$ | 2.403 | 1.082 | 0.7584 | 0.5218 |
| $c_2$ | −2.442 | −0.4081 | −0.1539 | −0.1766 |
| $c_3$ | 1.537 | 0.1708 | 0.04236 | 0 |

**Table 7.2** MoReRT constants, SOPRHPZ models with $a = 0.25$

| $M_S^t$ | 1.4 | 1.6 | 1.8 | 2.0 |
|---|---|---|---|---|
| $a_0$ | −1.196 | −0.2394 | −0.1668 | −0.1283 |
| $a_1$ | 1.404 | 0.6029 | 0.6198 | 0.648 |
| $a_2$ | −0.2819 | −0.6394 | −0.7079 | −0.7502 |
| $b_0$ | 1.872 | −1.061 | −0.1773 | 0.7155 |
| $b_1$ | −1.511 | 7.798 | 33.98 | 7.639 |
| $b_2$ | 1.705 | −0.8034 | 1.196 | 1.10 |
| $b_3$ | −2.222 | 6.258 | 23.34 | 5.665 |
| $b_4$ | 2.966 | −0.2579 | 3.007 | 1 |
| $b_5$ | −3.002 | 1 | 1 | 0 |
| $b_6$ | 1 | 0 | 0 | 0 |
| $c_0$ | 0.4325 | 0.4974 | 0.4896 | 0.5023 |
| $c_1$ | 2.144 | 0.9227 | 0.6414 | 0.4363 |
| $c_2$ | −2.434 | −0.3466 | −0.1187 | 0 |
| $c_3$ | 1.827 | 0.1798 | 0.04156 | 0 |

The robustness obtained with (7.3) and (7.4) for SOPRHPZ models are shown in Fig. 7.1 for the particular case of models with a time constant ratio $a = 0.25$ and right-half plane zero positions $b$ in the ranges where tuning relations are valid for. Robustness accomplishment for other $a$ values are shown in [1].

**Table 7.3** MoReRT constants, SOPRHPZ models with $a = 0.50$

| $M_S^t$ | 1.4 | 1.6 | 1.8 | 2.0 |
|---|---|---|---|---|
| $a_0$ | −1.46 | −0.4883 | −0.356 | −0.2266 |
| $a_1$ | 1.667 | 0.876 | 0.8524 | 0.799 |
| $a_2$ | −0.2298 | −0.4642 | −0.5458 | −0.6399 |
| $b_0$ | 9.525 | 7.481 | 1.364 | 1.293 |
| $b_1$ | −8.302 | −2.218 | 5.781 | 9.085 |
| $b_2$ | 7.048 | 5.899 | 1.247 | 1.349 |
| $b_3$ | −6.696 | −4.889 | 2.889 | 4.738 |
| $b_4$ | 4.844 | 6.401 | 1 | 1 |
| $b_5$ | −4.762 | −4.198 | 0 | 0 |
| $b_6$ | 1 | 1 | 0 | 0 |
| $c_0$ | 0.3617 | 0.5068 | 0.4807 | 0.4812 |
| $c_1$ | 2.419 | 0.6795 | 0.4604 | 0.3559 |
| $c_2$ | −3.262 | −0.1772 | 0 | 0 |
| $c_3$ | 2.401 | 0.1139 | 0 | 0 |

**Table 7.4** MoReRT constants, SOPRHPZ models with $a = 0.75$

| $M_S^t$ | 1.4 | 1.6 | 1.8 | 2.0 |
|---|---|---|---|---|
| $a_0$ | −2.225 | −0.4887 | −0.3895 | −0.3224 |
| $a_1$ | 2.456 | 0.9172 | 0.9365 | 0.9593 |
| $a_2$ | −0.1607 | −0.4531 | −0.5159 | −0.5687 |
| $b_0$ | 0.6535 | 6.43 | 2.705 | 1.684 |
| $b_1$ | 0.1825 | 8.407 | 7.026 | 12.12 |
| $b_2$ | 0.4613 | 4.614 | 1.98 | 1.518 |
| $b_3$ | −0.2916 | 1.685 | 3.024 | 5.597 |
| $b_4$ | 1.57 | 6.293 | 1 | 1 |
| $b_5$ | −1.934 | −3.449 | 0 | 0 |
| $b_6$ | 1 | 1 | 0 | 0 |
| $c_0$ | 0.345 | 0.5041 | 0.4598 | 0.4558 |
| $c_1$ | 2.137 | 0.4243 | 0.3662 | 0.2848 |
| $c_2$ | −2.837 | 0.05332 | 0 | 0 |
| $c_3$ | 1.96 | 0 | 0 | 0 |

### *Evaluation of MoReRT Controllers*

As an example of the MoReRT control of an inverse response process, consider the SOPRHPZ model given by the transfer function:

$$P(s) = \frac{-0.80s + 1}{(s + 1)(0.4s + 1)}. \tag{7.6}$$

**Table 7.5** MoReRT constants, SOPRHPZ models with $a = 1.0$

| $M_S^t$ | 1.4 | 1.6 | 1.8 | 2.0 |
|---|---|---|---|---|
| $a_0$ | −2.486 | −0.5199 | −0.447 | −0.406 |
| $a_1$ | 2.765 | 0.9989 | 1.056 | 1.117 |
| $a_2$ | −0.141 | −0.4339 | −0.4864 | −0.5237 |
| $b_0$ | 11.54 | −2.739 | 2.733 | 2.647 |
| $b_1$ | −2.237 | 18.01 | 10.61 | 16.11 |
| $b_2$ | 6.451 | −1.431 | 1.869 | 2.053 |
| $b_3$ | −0.446 | 9.672 | 4.295 | 6.579 |
| $b_4$ | 0.1767 | −0.3029 | 1 | 1 |
| $b_5$ | 1 | 1 | 0 | 0 |
| $b_6$ | 0 | 0 | 0 | 0 |
| $c_0$ | 0.4901 | 0.423 | 0.4358 | 0.4281 |
| $c_1$ | 0.8543 | 0.5706 | 0.2882 | 0.2282 |
| $c_2$ | −0.6064 | −0.2504 | 0 | 0 |
| $c_3$ | 0.5362 | 0.1038 | 0 | 0 |

**Table 7.6** Admissible SOPRHZ models $b$ ranges

| $M_S^t$ | 1.4 | 1.6 | 1.8 | 2.0 |
|---|---|---|---|---|
| $b_{min}$ | 0.25 | 0.25 | 0.25 | 0.25 |
| $b_{max}$ | 1.0 | 1.5 | 2.0 | 2.5 |

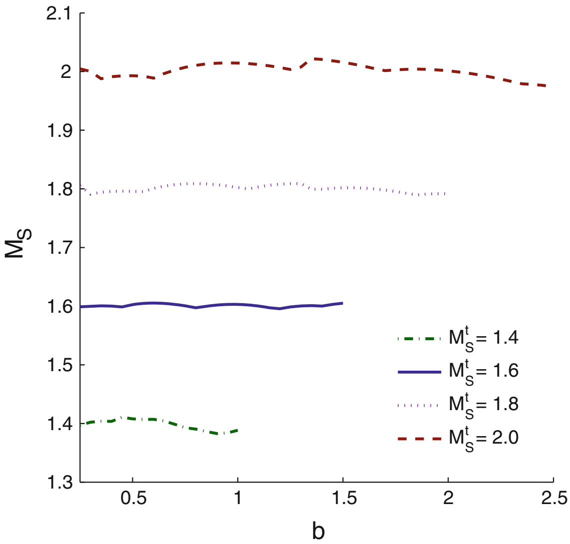



**Fig. 7.1** MoReRT PI controllers robustness accomplishment for second-order plus right-half plane zero models ($a = 0.25$)

**Table 7.7** Inverse response model example, $PI_2$ controller parameters

| $M_S^t$ | $K_p$ | $T_i$ | $\beta$ |
|---|---|---|---|
| 1.4 | 0.297 | 1.006 | 1.471 |
| 1.6 | 0.472 | 1.243 | 1.188 |
| 1.8 | 0.588 | 1.340 | 1.183 |
| 2.0 | 0.672 | 1.388 | 1.173 |

**Table 7.8** Inverse response model example, robustness, performance, and control effort indices

| $M_S^t$ | $M_S$ | $J_{er}/\Delta r$ | $J_{ed}/\Delta d$ | $TV_{ur}/\Delta r$ | $TV_{ud}/\Delta d$ |
|---|---|---|---|---|---|
| 1.4 | 1.40 | 2.923 | 3.756 | 1.000 | 1.236 |
| 1.6 | 1.61 | 2.401 | 3.013 | 1.117 | 1.377 |
| 1.8 | 1.82 | 2.044 | 2.676 | 1.541 | 1.486 |
| 2.0 | 1.99 | 1.988 | 2.471 | 1.957 | 1.682 |

The $PI_2$ controller parameters for (7.6) are listed in Table 7.7. Please not that in this case all set-point proportional weights $\beta > 1$.

Table 7.8 lists the robustness ($M_S$), performance ($J_e$), and control effort total variation ($TV_u$) obtained for the four target robustness levels ($M_S^t$) to a set-point step change ($\Delta r$) and a disturbance step change ($\Delta d$). These data show the trade-offs between the control system *robustness*, its *performance*, and the *control effort*. An increment in the control system target robustness (reducing $M_S^t$) reduces its performance (increases $J_{er}$ and $J_{ed}$) but made the control effort smoother (reduces $TV_{ur}$ and $TV_{ud}$).

The corresponding closed-loop responses to a 20 % set-point followed by a 10 % disturbance step change are shown in Fig. 7.2.

## 7.3 Proportional Integral Derivative Control

### *7.3.1 2DoF Ideal PID with Filter*

The application of MoReRT design to 2DoF Ideal PID with filter controllers for inverse response processes is presented in [3].

Now, the control system of a SOPRHPZ (5.3) with the $PID_{2F}$ (2.1) controller is of fourth order. Taking into consideration the results from Sect. 6.2.3, the regulatory control closed-loop transfer function target (4.15) is selected with two underdamped dominant poles given by

$$M_{yd}^t(s) = \frac{(T_i^*/K_p^*)(T_f s + 1)s(-bTs + 1)}{(\tau_c^2 T^2 s^2 + 2\zeta\tau_c Ts + 1)(\tau_c Ts + 1)(a\tau_c Ts + 1)}, \tag{7.7}$$

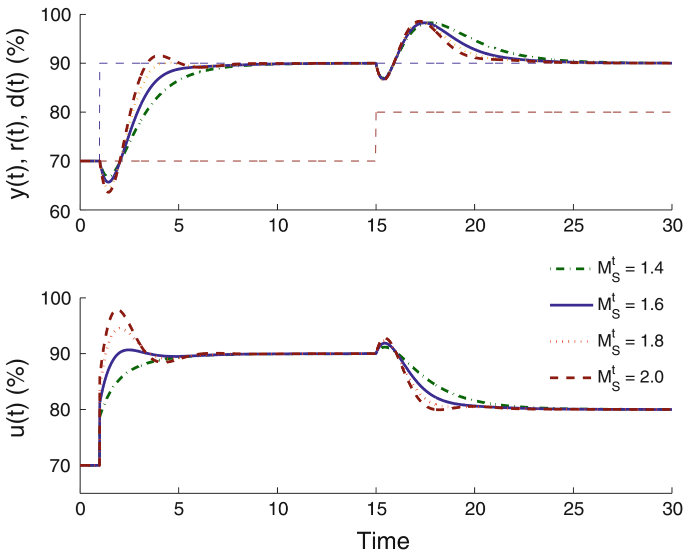



**Fig. 7.2** Inverse response model example, control system responses

where the closed-loop poles target damping ratio $\zeta$ and relative speed $\tau_c$ are the design parameters.

Selecting $T_f = \tau_c T$ in order to cancel the slowest closed-loop pole reduces (7.7) to the following third-order regulatory control closed-loop transfer function target

$$M_{yd}^t(s) = \frac{(T_i^*/K_p^*)s(-bTs+1)}{(\tau_c^2T^2s^2+2\zeta\tau_cTs+1)(a\tau_cTs+1)} = \frac{(-bTs+1)}{(\tau_c^2T^2s^2+2\zeta\tau_cTs+1)}\frac{(T_i^*/K_p^*)s}{(a\tau_cTs+1)}. \tag{7.8}$$

Using (7.8) the obtained servo-control closed-loop transfer function target (4.18) is

$$M_{yr}^t(s) = \frac{(\beta^*T_i^*s+1)(-bTs+1)}{(\tau_c^2T^2s^2+2\zeta\tau_cTs+1)(a\tau_cTs+1)}. \tag{7.9}$$

To locate the controller zero to the left of the underdamped closed-loop poles the proportional set-point weight is selected as $\beta^* \to \tau_c T/T_i^*$. Then, the servo-control closed-loop transfer function target is given by

$$M_{yr}^t(s) = \frac{(\tau_cTs+1)(-bTs+1)}{(\tau_c^2T^2s^2+2\zeta\tau_cTs+1)(a\tau_cTs+1)} = \frac{(-bTs+1)}{(\tau_c^2T^2s^2+2\zeta\tau_cTs+1)}\frac{(\tau_cTs+1)}{(a\tau_cTs+1)}. \tag{7.10}$$

Using (7.8) and (7.10) the closed-loop control system total response target is stated as

$$y^t(s) = \frac{(\tau_c Ts + 1)(-bTs + 1)}{(\tau_c^2 T^2 s^2 + 2\zeta \tau_c Ts + 1)(a\tau_c Ts + 1)} r(s) + \frac{(T_i^*/K_p^*)s(-bTs + 1)}{(\tau_c^2 T^2 s^2 + 2\zeta \tau_c Ts + 1)(a\tau_c Ts + 1)} d(s). \quad (7.11)$$

The performance analysis of the control system responses obtained using target damping ratios $\zeta$ from 1.0 to 0.6 shows that the highest performance improvement, measured with the integrated absolute error $J_{eT}$, is obtained, as previously with PI controllers, using $\zeta \approx 0.8$ but with an increase of the control effort total variation $TV_{uT}$.

For $\zeta = 0.8$ and each robustness target level the controller dimensionless parameters (5.12) are found for right-half plane zero positions $b$ in the range from 0.1 to 3.0 and model time constants ratio $a$ in the range from 0.1 to 1.0.

From the optimization data, it is to be noted that the position of the right-half plane zero constrains the robustness level that may be achieved. Additionally, high $b$ values made the controller derivative time to drop fast towards zero. Taking this into account, for the analysis, the position of the model right-half plane zero is taken in the range from 0.1 to 2.6 for robustness target level $M_S^t = 2.0$ and from 0.1 to 1.15 for $M_S^t = 1.6$.

The SOPRHPZ is dependent on two parameters, the right-half plane zero position $b$ and the model time constants ratio $a$. This implies that the tuning rule becomes now a surface instead of a curve. Therefore, for each robustness level we will have to fit a surface. Using a surface fitting techniques the obtained controller parameter are used to fit the $PID_{2F}$ MoReRT equations for SOPRHPZ models. For the target robustness level $M_S^t = 2.0$ the surface is given by

$$\{\kappa_p^*, \tau_i^*, \tau_d^*, \tau_f, \beta^*\} = c_0 + c_1\, a + c_2\, b + c_3\, a\, b + c_4\, b^2 + c_5\, a\, b^2 + c_6\, b^3, \quad (7.12)$$

**Table 7.9** MoReRT tuning constants, SOPRHPZ models, $M_S^t = 2.0$

| | $\kappa_p^*$ | $\tau_i^*$ | $\tau_d^*$ | $\tau_f$ | $\beta^*$ |
|---|---|---|---|---|---|
| $c_0$ | 3.808 | 0.8608 | 0.2728 | 0.316 | 0.4172 |
| $c_1$ | 0.4111 | 0.988 | 0.4044 | 0.1914 | −0.08662 |
| $c_2$ | −6.217 | 1.599 | 0.4831 | 1.082 | 0.2333 |
| $c_3$ | −0.3489 | 0.03134 | 0.09845 | −0.01138 | −0.05651 |
| $c_4$ | 4.25 | −0.9303 | −0.3138 | −0.4785 | −0.07367 |
| $c_5$ | 0.09239 | −0.06749 | −0.0327 | −0.03542 | 0.01223 |
| $c_6$ | −0.9784 | 0.2252 | 0.06335 | 0.1128 | 0.01114 |

**Table 7.10** MoReRT tuning constants, SOPRHPZ models, $M_S^t = 1.6$

| | $\kappa_p^*$ | $\tau_i^*$ | $\tau_d^*$ | $\tau_f$ | $\beta^*$ |
|---|---|---|---|---|---|
| $c_0$ | 3.042 | 1.079 | 0.3174 | 0.4545 | 0.4418 |
| $c_1$ | −0.07878 | 1.121 | 0.4253 | 0.2903 | −0.08582 |
| $c_2$ | −4.409 | 1.355 | 0.4505 | 1.042 | 0.2538 |
| $c_3$ | 0.1679 | −0.26 | −0.03513 | −0.189 | −0.07043 |
| $c_4$ | 2.153 | −0.4896 | −0.3367 | −0.1696 | −0.0276 |

and for $M_S^t = 1.6$ given by

$$\{\kappa_p^*, \tau_i^*, \tau_d^*, \tau_f, \beta^*\} = c_0 + c_1\, a + c_2\, b + c_3\, a\, b + c_4\, b^2. \tag{7.13}$$

where the $c_i$ constants in (7.12) and (7.13) are presented in Tables 7.9 and 7.10 [3], respectively. Tuning relations (7.12) ($M_S^t = 2.0$) are valid for zero relative position $b$ in the range from 0.25 to 2.0 and tuning relations (7.13) ($M_S^t = 1.6$) for $b$ from 0.25 to 1.0.

***Controller High-Frequency Gain and Measurement Noise Filtering***

From (2.21) the controller high-frequency gain is

$$K_{PID2F\infty} \doteq \left|C_y(\mathrm{j}\omega)\right|_{\omega\to\infty} = \frac{K_p^* T_d^*}{T_f^*}, \tag{7.14}$$

and in normalized form

$$\kappa_{PID2F\infty} = \frac{\kappa_p^* \tau_d^*}{\tau_f^*}. \tag{7.15}$$

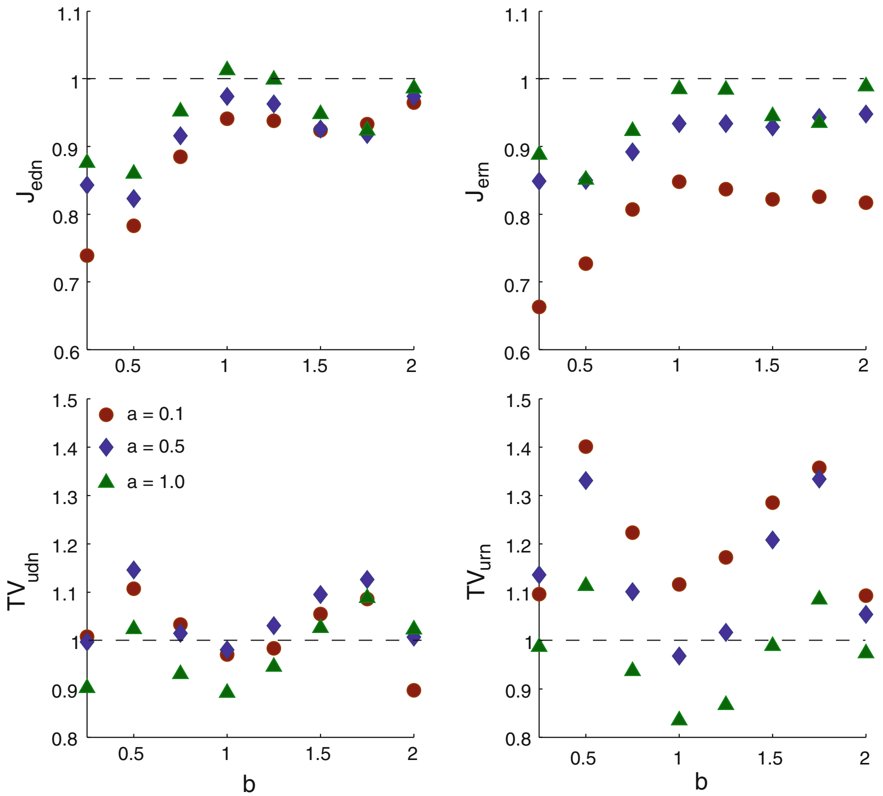



**Fig. 7.3** $PID_{2F}/PI_2$ performance and control effort total variation relation for $M_S^t = 2.0$. $J_{en} = J_{ePID}/J_{ePI}$, $TV_{un} = TV_{uPID}/TV_{uPI}$

The MoReRT $PID_{2F}$ controllers normalized high frequency is very low [3], $\kappa_{PID2F\infty} \leq 2$ in all cases except for $b \leq 0.15 + 0.30a$ ($M_S^t = 2.0$).

This characteristic will improve the measurement high-frequency noise filtering.

#### *Controllers Performance Comparison*

The characteristics of the two-degree-of-freedom PID controller with filter ($PID_{2F}$) tuning presented above for second-order inverse response controlled processes is compared with the corresponding to the two-degree-of-freedom PI controller ($PI_2$) tuning described in Sect. 7.2 for the same processes.

Figure 7.3 shows the $PID_{2F}$ to $PI_2$ performance, measured with the integrated absolute error, and control effort total variation relations for three SOPRHPZ model time constants ratios $a \in \{0.1, 0.5, 1.0\}$ and robustness target level $M_S^t = 2.0$. Comparison for $M_S^t = 2.0$ with other model time constants ratios $a$ and for robustness target level $M_S^t = 1.6$ are presented in [3].

Although indices depend on the model parameters $a$ and $b$, the control system operation (servo- or regulatory control), and the robustness level it can be said that, in general, the $PID_{2F}$ controllers produce control systems with higher performance, under the integrated absolute error metric, but at the same time with more variability in their control signal.

For the $M_S^t = 2.0$ level the $PID_{2F}$ controllers provide an average of 7.8 % (regulatory control) and 10.5 % (servo-control) performance increase with 2.3 and 11.8 % higher total variation of control effort, respectively. For the $M_S^t = 1.6$ level the corresponding average figures are 8.8 and 13.7 % more performance with 1.04 and 19.4 % more control effort variability.

If a general comment needs to be given, and based on the performance indices only, the use of the $PID_{2F}$ controllers is recommended in all cases for the inverse response models considered.

### 7.3.2 2DoF Ideal PID with Two Input Filters

The application of the MoReRT tuning methodology to tune PI and PID controllers aggregated with to input filters introduced in Sect. 2.4 to control a SOPRHP processes is presented in [2].

In this case, the regulatory control closed-loop target transfer function is selected as

$$M_{yd}^t(s) = \frac{(1/K_i)s(-bTs + 1)}{(\tau_c^2T^2s^2 + 2\zeta\tau_cTs + 1)(a\tau_cTs + 1)}. \tag{7.16}$$

From a case analysis (models (7.1) with $0.1 \le a \le 1.0$ and $0.1 \le b \le 2.0$) it is found that the minimum target robustness $M_S^t = 2.0$ can be obtained in all cases even if the closed-loop poles damping ratio $\zeta$ is decreased to 0.7, but the intermediate robustness $M_S^t = 1.6$ for $\zeta < 1$ can be obtained only if $b$ is very low.

For closed-loop transfer functions (7.16) with $\zeta = 1.0$ the target robustness can be guaranteed for right-half plane zero relative positions from 0.1 to the $b_{max}$ listed in Table 7.11.

As can be seen the $PI_{2IF}$ controller allows one to obtain robust control system for a wider range of inverse response models.

**Table 7.11** Model non-minimum phase zero position

| | $PI_{2IF}$ | | $PID_{2IF}$ | |
|---|---|---|---|---|
| $M_S^t$ | 2.0 | 1.6 | 2.0 | 1.6 |
| $b_{max}$ | 6.5 | 2.8 | 4.8 | 2.1 |

**Table 7.12** $PI(D)_{2IF}$ parameters and performance

| | $PI_{2IF}$ | | $PID_{2IF}$ | |
|---|---|---|---|---|
| $M_S^t$ | 2.0 | 1.6 | 2.0 | 1.6 |
| $K_p$ | 0.1844 | 0.1229 | 0.2166 | 0.1710 |
| $K_i$ | 0.05953 | 0.04685 | 0.07158 | 0.05900 |
| $K_d$ | 0.0 | 0.0 | 0.1710 | 0.1241 |
| $T_f$ | 0.2302 | 0.2852 | 0.2186 | 0.2518 |
| $T_r$ | 3.0976 | 2.6233 | 3.0260 | 2.8983 |
| $\sigma$ | 1.7739 | 2.1592 | 1.6366 | 1.8506 |
| $\gamma$ | 0 | 0 | 0 | 0 |
| $J_{ed}$ | 20.235 | 24.558 | 17.689 | 20.412 |
| $TV_{ud}$ | 1.653 | 1.345 | 2.200 | 1.775 |
| $E_{max}$ | 2.931 | 2.783 | 2.397 | 2.463 |
| $J_{er}$ | 6.377 | 6.989 | 6.512 | 6.890 |
| $TV_{ur}$ | 0.438 | 0.382 | 0.395 | 0.391 |
| $U_{max}$ | 0.378 | 0.358 | 0.364 | 0.362 |

***Evaluation of MoReRT Controllers***

Consider the second-order plus right-half plane zero model frequently used for controllers test given by the transfer function:

$$P(s) = \frac{3(-2s+1)}{(2s+1)(s+1)}, \;\; a = 2, \; b = 1. \tag{7.17}$$

The $PI_{2IF}$ and $PID_{2IF}$ controllers parameters are listed in Table 7.12. This table also lists the performance and control effort usage with these controllers.

Control system responses to a 10 % set-point change followed by a 5 % disturbance step change are shown in Fig. 7.4.

As the table and figure show all the responses are very similar, but for same robustness level better regulatory control performance ($J_{ed}$, $E_{max}$) is obtained with the $PID_{2IF}$ controller than with the $PI_{2IF}$.

Increasing the closed-loop system target robustness from $M_S^t = 2.0$ to $M_S^t = 1.6$ reduces the regulatory control performance (increase $J_{ed}$) 21 % (PI) and 15 % (PID).

## Chapter Remarks

Tuning relations with two robustness levels, $M_S^t \in \{2.0,\ 1.4\}$ for proportional integral (PI) controllers to control inverse response processes represented by SOPRHPZ models are obtained. The non-minimum phase zero position imposes constraints to the achievable robustness. These limitations become higher as the zero moves towards the origin (the model zero relative position $b$ increases).

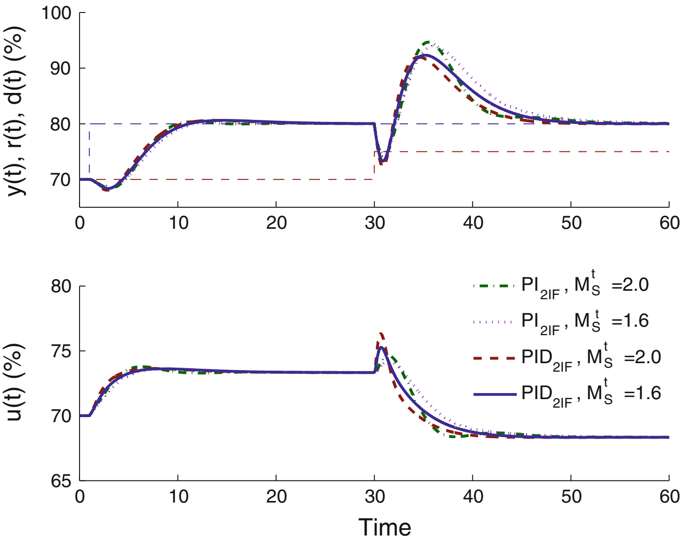



**Fig. 7.4** Inverse response model example, control system responses

It is shown that using a two-degree-of-freedom ideal PID controller with filter ($PID_{2F}$), more performance, measured with the integrated absolute error, is obtained in comparison with the performance obtained with a two-degree-of-freedom PI controller.

Aggregating the PID controller with two input filters (set-point and feedback signal filters) provides two additional useful features: high-frequency roll-off and lack of control effort abrupt changes. The addition of these two filters also expands the range of the non-minimum phase models (a wide $b$ range) that can be robustly controlled.

As with the $PID_{2F}$ to $PI_{2F}$ performance relation the $PID_{2IF}$ provides more regulatory control performance than the $PI_{2IF}$.

## References


1. Alfaro, V.M., Vilanova, R.: Two-degree-of-freedom proportional integral control of inverse response second-order processes. In: 16th International Conference on Systems Theory, Control and Computing (ICSTCC 2012), 12–14 October, Sinaia, Romania (2012)
2. Alfaro, V.M., Vilanova, R.: Performance and robustness considerations for tuning of proportional integral/proportional integral derivative controllers with two input filters. Ind. Eng. Chem. Res. **52**, 18287–18302 (2013)
3. Alfaro, V.M., Vilanova, R.: Robust tuning of 2DoF five-parameters PID controllers for inverse response controlled processes. J. Process Control **23**, 453–462 (2013)

# Chapter 8
# MoReRT Control of Integrating Processes

Even though most of the controlled processes found in the process industry are self-regulating, that is, the process output seeks a stable operating point under a constant input, there are others that under a constant input their output is unbounded, rise or decrease without limit. These non self-regulated processes are called integrating or unstable if their model transfer functions have poles at the *s*-plane origin or at its right-half plane, respectively. Stable processes with very long time constants may also be approximated by integrating models.

Integrating and unstable processes may be operated only under closed-loop control and their controller tuning needs a special treatment.

The extension of the MoReRT methodology to integrating plus dead-time (IPDT) and integrating second-order plus dead-time (ISOPDT) models is presented in [1, 2] for PI controllers and in [3] for the $PID_{2IF}$.

## 8.1 Introduction

For the integrating processes two models are considered: the integrating second-order plus dead-time (ISOPDT) model, and the integrating plus dead-time (IPDT) model. In both cases the process model transfer function includes an integrator and a dead-time but, in the first case, the extra pole allows a better description of the initial transient dynamics of the open-loop response. In fact, after a short time, because of the integrating nature of the process, the response will evolve in an identical divergent way. Therefore what is interesting for control is to catch the dynamics during the transient. As an example, Fig. 8.1 shows an integrating process step response and those of an IPDT and ISOPDT model. It is seen that ISOPDT models are able to better capture the initial transient dynamics.



V.M. Alfaro and R. Vilanova, *Model-Reference Robust Tuning of PID Controllers*,
Advances in Industrial Control, DOI 10.1007/978-3-319-28213-8_8

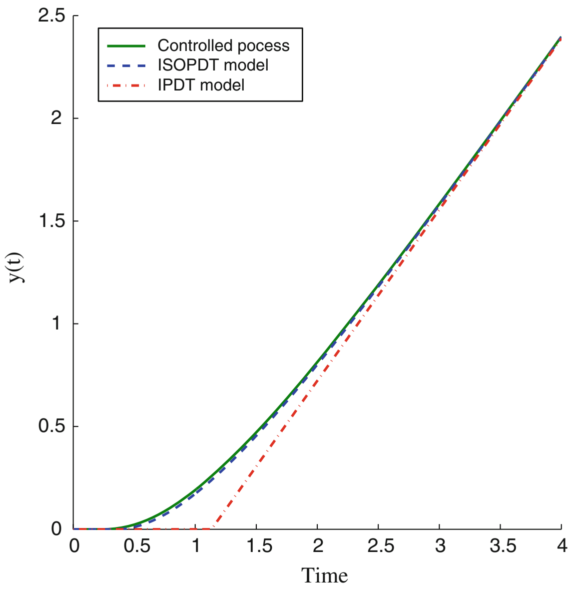



**Fig. 8.1** Integrating process and their IPDT and ISOPDT models

## 8.2 Integrating Second-Order Plus Dead-Time Models

The integrating second-order plus dead-time (ISOPDT) model (5.5) is rewritten here as

$$P(s) = \frac{K\mathrm{e}^{-Ls}}{s(Ts+1)}. \tag{8.1}$$

Its normalized model (5.6) has only one parameter, $\tau_L$. Controller normalized parameters can be obtained as function of $\tau_L$ and the target control system robustness level $M_S^t$.

### *8.2.1 2DoF Proportional Integral Control*

For the derivation of the 2DoF PI control tuning relations, two cases will distinguished in terms of the specified target responses: overdamped target responses and underdamped responses.

**Overdamped Target Responses**

The proportional integral ($PI_2$) control of (8.1) is of third order. Then, the non-oscillatory global control system output target $y^t(s)$ (4.7) is computed as

$$y^t(s) = \frac{e^{-Ls}}{(\tau_c Ts + 1)^2} r(s) + \frac{(T_i/K_p)se^{-Ls}}{(\tau_c Ts + 1)^3} d(s). \quad (8.2)$$

The $PI_2$ controller parameters obtained following the optimization procedure stated in Chap. 4 are used to fit the 2DoF PI controller normalized parameters equations as functions of the controlled process model (8.1) parameter and of the design robustness $M_S^t$ specified and given by

$$\kappa_p = \frac{a_0 + a_1\tau_L}{a_2 + a_3\tau_L + \tau_L{}^2}, \quad (8.3)$$

$$\tau_i = b_0 e^{b_1\tau_L} + b_2 e^{b_3\tau_L}, \quad (8.4)$$

$$\beta = c_0 + c_1\tau_L + c_2\tau_L{}^2. \quad (8.5)$$

The $a_i$, $b_i$ and $c_i$ constants for relations (8.3)–(8.5) for four robustness target levels ($M_S^t \in \{1.4,\ 1.6,\ 1.8,\ 2.0\}$) are listed in Table 8.1 [1].

The robustness obtained with (8.3) and (8.4) for ISOPDT models is shown in Fig. 8.2. As can be seen all the robustness profiles are nearly flat. This means that for an ISOPDT model of a controlled process the MoReRT tuning guarantees that the robustness target is attained for all normalized dead-times in the range considered. It is important to note that other available robust tuning rules for ISOPDT models do not produce control systems with a constant robustness level on their applicability range. Robustness evaluations are presented in [1].

**Table 8.1** MoReRT constants, ISOPDT models, overdamped target responses

| $M_S^t$ | 1.4 | 1.6 | 1.8 | 2.0 |
|---|---|---|---|---|
| $a_0$ | 0.4141 | 0.3781 | 0.3335 | 0.4156 |
| $a_1$ | 0.3126 | 0.4085 | 0.4826 | 0.5484 |
| $a_2$ | 0.9140 | 0.5324 | 0.3363 | 0.3317 |
| $a_3$ | 2.402 | 1.864 | 1.519 | 1.567 |
| $b_0$ | 18.38 | 12.77 | 10.27 | 9.123 |
| $b_1$ | 0.2110 | 0.2408 | 0.2637 | 0.2735 |
| $b_2$ | –11.08 | –7.192 | –5.589 | –4.978 |
| $b_3$ | –0.4811 | –0.6446 | –0.8042 | –0.8900 |
| $c_0$ | 0.3331 | 0.3336 | 0.3305 | 0.3257 |
| $c_1$ | –0.03254 | –0.03273 | –0.02941 | –0.002441 |
| $c_2$ | 0.007548 | 0.00713 | 0.005826 | 0.004094 |

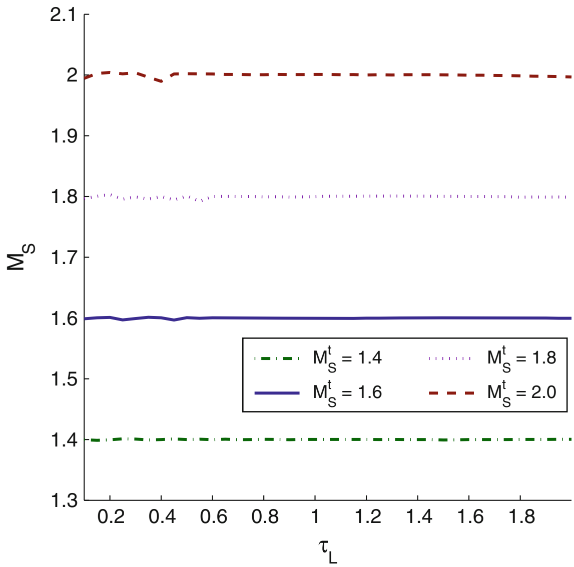



**Fig. 8.2** MoReRT PI controllers robustness accomplishment for integrating second-order plus dead-time models

### Under-Damped Target Responses

It is shown in [2] that if the target closed loop responses for the $PI_2$ control of an ISOPDT process are take as

$$y^t(s) = \frac{e^{-Ls}}{\tau_c^2 T^2 s^2 + 2\zeta\tau_c Ts + 1} r(s) + \frac{(T_i/K_p)se^{-Ls}}{(\tau_c^2 T^2 s^2 + 2\zeta\tau_c Ts + 1)(\tau_c Ts + 1)} d(s), \quad (8.6)$$

with a closed-loop poles damping ratio $\zeta = 0.80$, it is possible to improve the control system performance measured with the integrated absolute error without a significant increment in the control effort total variation.

In this case the normalized controller parameters can be obtained with the following equations:

$$\kappa_p = \frac{a_0 + a_1\tau_L}{a_2 + a_3\tau_L + \tau_L{}^2}, \quad (8.7)$$

$$\tau_i = b_0 e^{b_1\tau_L} + b_2 e^{b_3\tau_L}, \quad (8.8)$$

$$\beta = \frac{c_0 + c_1\tau_L}{c_2 + \tau_L}. \quad (8.9)$$

**Table 8.2** MoReRT constants, ISOPDT models, under-damped target responses

| $M_S^t$ | 1.4 | 1.6 | 1.8 | 2.0 |
|---|---|---|---|---|
| $a_0$ | 0.1040 | 0.1365 | 0.3886 | 1.0730 |
| $a_1$ | 0.2800 | 0.3740 | 0.4670 | 0.5955 |
| $a_2$ | 0.2539 | 0.2092 | 0.4455 | 0.9721 |
| $a_3$ | 1.1970 | 1.1410 | 1.7280 | 3.3320 |
| $b_0$ | 16.6700 | 10.9800 | 9.7360 | 8.3220 |
| $b_1$ | 0.2070 | 0.2533 | 0.2477 | 0.2708 |
| $b_2$ | –10.0600 | –5.9460 | –5.4550 | –4.5330 |
| $b_3$ | –0.4497 | –0.6769 | –0.6914 | –0.8444 |
| $c_0$ | 0.4673 | 0.5192 | 0.5853 | 0.6714 |
| $c_1$ | 0.3093 | 0.3001 | 0.2927 | 0.2860 |
| $c_2$ | 1.2150 | 1.3490 | 1.5360 | 1.7870 |

Table 8.2 [2] lists the $a_i$, $b_i$ and $c_i$ constants for (8.7)–(8.9) for four robustness levels.

### 8.2.2 2DoF PI Controller with Two Input Filters

From the performance analysis made in [3] using a $PI_{2IF}$ controller (see Sect. 2.4) it is found that a reduction of the closed-loop poles damping ratio worsens the control system performance. Then, a damping ratio $\zeta = 1$ is used for the controller design.

Normalized parameters (5.13) for robust tuning of $PI_{2IF}$ controllers for ISOPDT models with $0.1 \leq \tau_L \leq 2.0$ can be obtained with following relations for $M_S^t = 2.0$

$$\kappa_p = \frac{0.5163 + 0.5095\tau_L}{0.5788 + 2.099\tau_L + \tau_L^2}, \tag{8.10}$$

$$\kappa_i = \frac{0.09323}{0.5438 + 2.302\tau_L + \tau_L^2}, \tag{8.11}$$

$$\tau_f = 0.1407 + 0.1687\tau_L^{0.8976}, \tag{8.12}$$

$$\tau_r = 5.603 + 5.829\tau_L, \tag{8.13}$$

$$\sigma = \frac{1.393 + 2.379\tau_L + 1.215\tau_L^2}{0.9866 + 1.991\tau_L + \tau_L^2}, \tag{8.14}$$

and for $M_S^t = 1.6$ with

$$\kappa_p = \frac{0.2461 + 0.3372\tau_L}{0.5108 + 1.672\tau_L + \tau_L^2}, \tag{8.15}$$

$$\kappa_i = \frac{0.04682}{0.7068 + 2.192\tau_L + \tau_L^2}, \tag{8.16}$$

$$\tau_f = 0.2163 + 0.2416\tau_L^{0.9322}, \tag{8.17}$$

$$\tau_r = 7.528 + 7.663\tau_L, \tag{8.18}$$

$$\sigma = \frac{8.773 + 15.53\tau_L + 1.197\tau_L^2}{6.847 + 13.13\tau_L + \tau_L^2}. \tag{8.19}$$

Set-point filter parameters in (8.13) and (8.14) and in (8.18) and (8.19) are obtained selecting $\tau_r = \kappa_p/\kappa_i$ and $\sigma$ such that the integrated absolute error (3.7) is optimized.

## 8.3 Integrating Plus Dead-Time Models

Integrating and overdamped processes with very large time constants can also be approximated by an integrating plus dead-time (IPDT) model given by the following:

$$P(s) = \frac{K\mathrm{e}^{-Ls}}{s}, \tag{8.20}$$

were $K$ is the gain and $L$ the dead-time. The controlled process parameters are $\theta_p = \{K, L\}$.

### *8.3.1 Proportional Integral Control*

In this case, the PI control system is of second-order and the overdamped closed-loop control controlled variable target (4.7) is selected as

$$y^t(s) = \frac{\mathrm{e}^{-Ls}}{\tau_c Ls + 1} r(s) + \frac{(T_i/K_p)s\mathrm{e}^{-Ls}}{(\tau_c Ls + 1)^2} d(s), \tag{8.21}$$

where $\tau_c$ is the design parameter.

As the integrated plus dead-time controlled process (8.20) normalized model (5.8) does not have any variable parameter just one optimization run is required for each robustness target level.

In the same way as with the ISOPDT model, during the optimization process, the closed-loop relative speed parameter $\tau_c$ is selected in such a way that the robustness level of the resulting closed-loop system met a specific target ($M_S^t$) in the range from 1.4 to 2.0. The controller parameters are obtained directly as functions of only the closed-loop control system robustness.

The MoReRT $PI_2$ tuning equations for the IPDT models are as follows:

$$\kappa_p = a, \tag{8.22}$$

$$\tau_i = b, \tag{8.23}$$

$$\beta = c. \tag{8.24}$$

Constants $a, b$ and $c$ for (8.22)–(8.24) for robustness target $M_S^t \in \{1.4,\ 1.6,\ 1.8,\ 2.0\}$ are listed in Table 8.3 [1]. Relations (8.22)–(8.24) are valid for IPDT models with any nonzero dead-time.

For the case of under-damped target responses they are selected as

$$y^t(s) = \frac{(\tau_c Ls + 1)\mathrm{e}^{-Ls}}{\tau_c^2 L^2 s^2 + 2\zeta\tau_c Ls + 1} r(s) + \frac{(T_i/K_p)\mathrm{e}^{-Ls}}{\tau_c^2 L^2 s^2 + 2\zeta\tau_c Ls + 1} Ld(s). \tag{8.25}$$

The closed-loop poles damping ratio is selected as $\zeta = 0.8$.

For the under-damped target response constants $a$, $b$ and $c$ for (8.22)–(8.24) for robustness target $M_S^t \in \{1.4,\ 1.6,\ 1.8,\ 2.0\}$ are listed in Table 8.4 [2].

**Table 8.3** MoReRT constants, IPDT models, overdamped target responses

| $M_S^t$ | 1.4 | 1.6 | 1.8 | 2.0 |
|---|---|---|---|---|
| $a$ | 0.322 | 0.442 | 0.528 | 0.599 |
| $b$ | 10.636 | 7.885 | 6.579 | 5.823 |
| $c$ | 0.452 | 0.434 | 0.419 | 0.406 |

**Table 8.4** MoReRT constants, IPDT models, under-damped target responses

| $M_S^t$ | 1.4 | 1.6 | 1.8 | 2.0 |
|---|---|---|---|---|
| $a$ | 0.312 | 0.415 | 0.498 | 0.566 |
| $b$ | 8.086 | 6.217 | 5.320 | 4.802 |
| $c$ | 0.544 | 0.516 | 0.495 | 0.477 |

**Table 8.5** $PI(D)_{2IF}$ tuning constants for IPDT models

| $M_S^t$ | $PI_{2IF}$ | | $PID_{2IF}$ | |
|---|---|---|---|---|
| | 2.0 | 1.6 | 2.0 | 1.6 |
| $a$ | 0.4579 | 0.2991 | 0.6955 | 0.4738 |
| $b$ | 0.0751 | 0.0340 | 0.1825 | 0.0895 |
| $c$ | 0 | 0 | 0.3909 | 0.2902 |
| $d$ | 0.2662 | 0.4320 | 0.1876 | 0.2863 |
| $e$ | 6.0972 | 8.7972 | 3.8110 | 5.2939 |
| $f$ | 1.1847 | 1.1945 | 1.0786 | 1.1140 |

### *8.3.2 2DoF PID Controllers with Two Input Filters*

In this case, controller optimization is made in two steps. First, using the regulatory closed-loop transfer function target PI and PID controllers with a feedback filter are obtained for two robustness target levels.

The regulatory control target transfer function is

$$M_{yd}^t(s) = \frac{(1/K_i)se^{-Ls}}{\tau_c^2 L^2 s^2 + 2\zeta\tau_c Ls + 1}, \tag{8.26}$$

with $\zeta = 0.80$ for the PI controller and $\zeta = 0.70$ for the PID.

In the second step, and with above parameters at hand, the set-point filter parameters are obtained. The filter time constant is computed as $T_r = K_p/K_i$ and $\sigma$ is selected to improve the servo-control integrating absolute error $J_{er}$.

For robust tuning of $PI_{2IF}$ and $PID_{2IF}$ controllers for IPDT models the normalized parameters (5.17) are determined by the following simple relations:

$$\kappa_p = a, \tag{8.27}$$

$$\kappa_i = b, \tag{8.28}$$

$$\kappa_d = c, \tag{8.29}$$

$$\tau_f = d, \tag{8.30}$$

$$\tau_r = e, \tag{8.31}$$

$$\sigma = f, \tag{8.32}$$

$$\gamma = 0. \tag{8.33}$$

Table 8.5 [3] list the $a$, $b$, $c$, $d$, $e$, and $f$ constants for (8.27)–(8.32) for the two robustness levels.

## 8.4 Analysis of MoReRT Controllers

In order to exemplify the concrete application of the MoReRT methodology, some examples are presented in what follows. The examples will also serve the purpose of comparing the application of a PI or a PID controller, therefore allowing to evaluate the different performance levels that can be achieved under the same robustness.

### 8.4.1 $PID_{2IF}$ Comparison for Integrating First-Order Process

For evaluating purposes, the distillation column bottom level control manipulating the steam flow rate is used where the process model is given by the transfer function:

$$P_3(s) = \frac{0.2e^{-7.4s}}{s}. \tag{8.34}$$

Table 8.6 lists the robustness, performance, and control effort variation obtained using $PI_{2IF}$ and $PID_{2IF}$.

From this table it can be seen that for both robustness levels the $PID_{2IF}$ has the highest regulatory control performance, lowest $J_{ed}$, smoothest control effort, lowest $TV_{ud}$, and lowest maximum error, $E_{\max}$. By design, the $PI_{2IF}$ and $PID_{2IF}$ controllers do not have an output step to a set-point change and their control effort is very smooth.

**Table 8.6** $PI(D)_{2IF}$ tuning constants for IPDT models

| $M_S^t$ | $PID_{2IF}$ | | $PI_2$ | |
|---|---|---|---|---|
| | 2.0 | 1.6 | 2.0 | 1.6 |
| $M_S$ | 2.001 | 1.600 | 1.998 | 1.599 |
| $K^{\infty}$ | 0 | 0 | 0.382 | 0.280 |
| $J_{ed}$ | 66.557 | 134.588 | 93.019 | 164.959 |
| $TV_{ud}$ | 1.689 | 1.589 | 1.919 | 1.639 |
| $E_{\max}$ | 2.126 | 2.663 | 2.676 | 3.168 |
| $J_{er}$ | 30.178 | 41.433 | 19.639 | 25.568 |
| $TV_{ur}$ | 0.398 | 0.251 | 0.550 | 0.386 |
| $\Delta u_0$ | 0.0 | 0.0 | 0.182 | 0.145 |
| $U_{\max}$ | 0.188 | 0.118 | 0.262 | 0.190 |

### 8.4.2 Control of an Integrating Third-Order Process

As an example of the MoReRT tuning for integrating processes, consider the integrating third-order transfer function:

$$P(s) = \frac{0.833e^{-0.2s}}{s(0.833s+1)(0.1s+1)}. \tag{8.35}$$

For tuning purposes the controlled process model (8.35) is approximated by following ISOPDT and IPDT models:

$$P(s) \approx \frac{0.833e^{-0.353s}}{s(0.780s+1)} \approx \frac{0.833e^{-1.133s}}{s}. \tag{8.36}$$

Now, from the tunings obtained in this chapter, we will be able to compare the control of process (8.35) on the basis of the simple (IPDT) and more complex model (ISOPDT). One of the lessons to be learned is the benefit of using more complex models as more information about the process dynamics is available.

#### $PI_2$ Controllers

The MoReRT $PI_2$ controller parameters ($\zeta = 0.8$ tuning) for $M_S^t \in \{1.4,\ 1.6\}$ obtained with models (8.36) are listed in Table 8.7 and the control system responses to a 20 % set-point step change followed by a –5 % disturbance step change are shown in Fig. 8.3.

Table 8.7, includes the robustness ($M_S$) resulting with the third-order controlled process, that in practice can not be obtained. From the table it can be seen that the robustness of the control systems with the controllers tuned by using the ISOPDT

**Table 8.7** Integrating models example, controller parameters, robustness, performance, and control effort indices

| Model | IPDT | IPDT | ISOPDT | ISOPDT |
|---|---|---|---|---|
| $M_S^t$ | 2.0 | 1.6 | 2.0 | 1.6 |
| $K_p$ | 0.600 | 0.440 | 0.769 | 0.506 |
| $T_i$ | 5.441 | 7.044 | 4.925 | 6.191 |
| $\beta$ | 0.477 | 0.516 | 0.358 | 0.364 |
| $M_S$ | 1.724 | 1.487 | 1.945 | 1.582 |
| $J_{ed}$ | 9.068 | 16.067 | 6.404 | 12.235 |
| $TV_{ud}$ | 1.847 | 1.649 | 2.065 | 1.726 |
| $J_{er}$ | 3.195 | 4.007 | 3.165 | 4.011 |
| $TV_{ur}$ | 0.781 | 0.555 | 0.874 | 0.513 |
| $J_{eT}$ | 12.263 | 20.074 | 9.569 | 16.246 |
| $TV_{uT}$ | 2.629 | 2.205 | 2.939 | 2.239 |

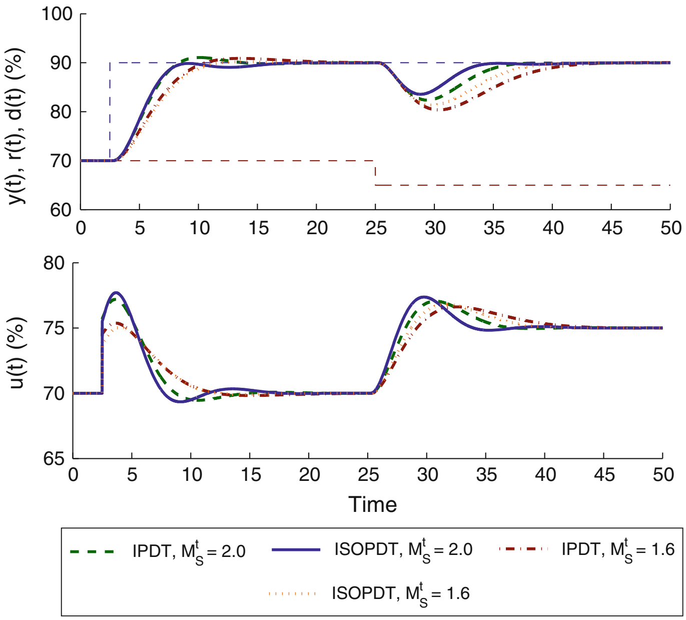



**Fig. 8.3** Integrating model example, control system responses

model are closer to the design robustness level than the ones obtained with the controllers tuned by using the IPDT model. This is interpreted as an indication that the ISOPDT model provides a better representation of the controlled process dynamics than the IPDT model.

The performance ($J_e$) and control effort ($TV_u$) indices shown on Table 8.7 were obtained with unitary set-point and disturbance step changes.

At both robustness levels, the regulatory performance of the controllers tuned with the ISOPDT model are higher that the ones obtained with the IPDT models and with similar servo-control performance. On the other hand, the control effort total variation is slightly higher for the controllers tuned with the ISOPDT model.

Comparison of MoReRT controllers for ISOPDT and IPDT models with other tuning rules is presented in [1] for relations (8.3)–(8.5) obtained with overdamped response targets and in [2] for relations obtained with under-damped response targets.

#### ***$PID_{2IF}$ Controller***

For the ISOPDT model in (8.36) Table 8.8 list the $PID_{2IF}$ controller parameters for $M_S^t \in \{2.0,\ 1.6\}$ and the regulatory and servo-control performance. If control system robustness needs to be increased some performance is lost ($J_{ed}$, $E_{\max}$, and $J_{er}$ increase), but the control effort is smoother ($TV_{ud}$, $TV_{ur}$, and $U_{\max}$ decrease).

Figure 8.4 shows the control system responses to a 20 % set-point change followed by a −10 % disturbance step-change.

**Table 8.8** $PID_{2IF}$ controller parameters and performance

| $M_S^t$ | 2.0 | 1.60 |
|---|---|---|
| $K_p$ | 2.0201 | 1.3172 |
| $K_i$ | 0.6099 | 0.3042 |
| $K_d$ | 1.3896 | 0.9138 |
| $T_f$ | 0.0977 | 0.1308 |
| $T_r$ | 3.3122 | 4.3300 |
| $\sigma$ | 1.1973 | 1.2097 |
| $\gamma$ | 0 | 0 |
| $J_{ed}$ | 1.643 | 3.287 |
| $TV_{ud}$ | 1.610 | 1.488 |
| $E_{\max}$ | 0.472 | 0.701 |
| $J_{er}$ | 3.050 | 3.913 |
| $TV_{ur}$ | 1.072 | 0.705 |
| $U_{\max}$ | 0.525 | 0.346 |

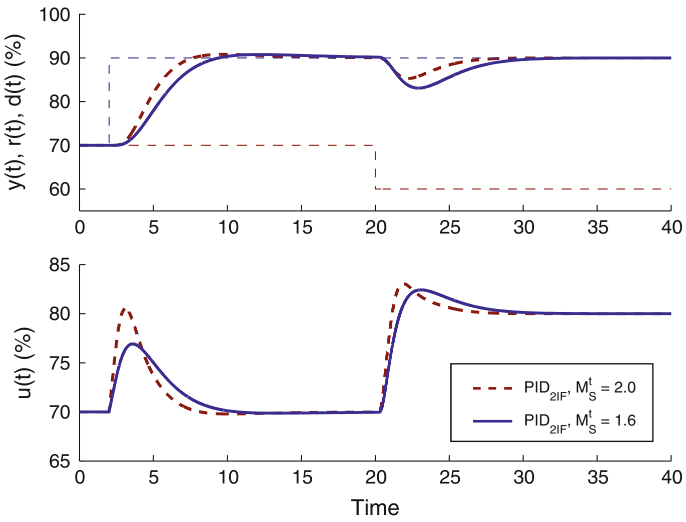



**Fig. 8.4** Example $PID_{2IF}$ control systems responses

Comparing $PI_2$ (Table 8.7) and $PID_{2IF}$ (Table 8.8) regulatory control performance for same robustness level, it is seen that the $PID_{2IF}$ can provide up to 390 % more performance than the $PI_2$.

### Chapter Remarks

For processes with integrating characteristics represented by integrated plus dead-time (IPDT) or integrated second-order plus dead-time (ISOPDT) models tuning relations are obtained using overdamped target responses and four robustness levels.

Using closed-loop under-damped response targets an analysis of the closed-loop poles damping ratio influence over the performance/control effort trade-off is made. Improvements in performance obtained reducing the damping ratio depend on the control system robustness and they turn smaller as the robustness level target is increased.

Tuning relations for IPDT and ISOPDT are obtained with the recommended damping ratio, $\zeta = 0.8$. These are valid for ISOPDT models with normalized dead-times $\tau_L$ from 0.1 to 2.0. Relations for IPDT models are valid for any nonzero dead-time $L$.

Performance comparison using an integrating third-order process shows that more performance is obtained with $PI_2$ controllers tuned using the ISOPDT model approximation than using the IPDT model. The use of the PID with two input filters ($PID_{2IF}$) provides even more regulatory performance, been this a better choice to control integrating processes.

## References

1. Alfaro, V.M., Vilanova, R.: Model reference robust tuning of 2DoF PI controllers for integrating controlled processes. In: IEEE 20th Mediterranean Conference on Control and Automation (MED 2012), Barcelona, Spain, 3–6 July 2012
2. Alfaro, V.M., Vilanova, R.: Robust tuning and performance analysis of 2DoF PI controllers for integrating controlled processes. Ind. Eng. Chem. Res. **51**(40), 13182–13194 (2012). doi:10.1021/ie300605w
3. Alfaro, V.M., Vilanova, R.: Performance and robustness considerations for tuning of proportional integral/proportional integral derivative controllers with two input filters. Ind. Eng. Chem. Res. **52**, 18287–18302 (2013)

# Chapter 9
# MoReRT Control of Unstable Processes

Previous chapter has considered the case of integrating processes. Integrating processes are a special case of unstable processes where the *unstable* pole is located at the origin. In process control, they are usually treated specifically as are representative of particular dynamics. In this chapter, we will proceed with the generic case of unstable processes. Open-loop unstable processes are presented in chemical industrial systems and are known to be difficult to control particularly if they include dead-time. The MoReRT methodology is applied to unstable controlled processes in [1–3].

## 9.1 Introduction

The unstable processes are represented by a first-order model with a pole at the right-half plane plus dead-time (UFOPDT) given by the following transfer function:

$$P(s) = \frac{K\mathrm{e}^{-Ls}}{Ts - 1}, \tag{9.1}$$

where $K$ is the gain, $T$ the time constant, and $L$ the process apparent dead-time. The controlled process parameters are $\theta_p = \{K, T, L\}$.

## 9.2 2DoF Proportional Integral Control

Proportional integral control system of first-order unstable processes is of second-order.

In this case due to the constraints imposed by the unstable process characteristics, it is not possible to obtain a first-order dynamics for the servo-control response



V.M. Alfaro and R. Vilanova, *Model-Reference Robust Tuning of PID Controllers*,
Advances in Industrial Control, DOI 10.1007/978-3-319-28213-8_9

by selecting $\beta \rightarrow \tau_c T/T_i$ as was made for first-order stable models. Then in the UFOPDT model case, the global control system output target is computed as

$$y^t(s) = \frac{(\beta T_i s + 1)\mathrm{e}^{-Ls}}{(\tau_c T s + 1)^2} r(s) + \frac{(T_i/K_p)s\mathrm{e}^{-Ls}}{(\tau_c T s + 1)^2} d(s). \tag{9.2}$$

The unstable controlled process normalized model (5.10) has only one dimensionless parameter, $\tau_L$. For a given $\tau_L$ during the optimization procedure, the closed-loop relative speed parameter $\tau_c$ is selected in such a way that the robustness level of the resulting closed-loop system meets a specific target ($M_S^t$).

On the basis of the optimization results it is found that the dead-time of the unstable processes imposes a severe constraint on the control system achievable robustness level. A closed-loop control systems with high robustness of $M_S^t = 1.4$ can only be obtainable for UFOPDT models with $\tau_L \leq 0.10$, the robustness $M_S^t = 1.6$ for $\tau_L \leq 0.15$, the $M_S^t = 1.8$ for $\tau_L \leq 0.20$, and the robustness $M_S^t = 2.0$ for models with $\tau_L \leq 0.26$. Robust control systems may be obtained for a very limited range of unstable models.

Although the usual control system minimum robustness level for stable processes corresponds to $M_S = 2.0$ in the unstable processes case the main control system purpose is to stabilize the process. Then for the UFOPDT models, the robustness level target is relaxed and the controller parameters are obtained for $M_S^t$ in the range from 2.0 to 6.0. The optimization also shows that for all cases the set-point weight $\beta = 0$.

Then the $PI_2$ controller parameters are obtained as functions of the model (9.1) parameter and of the closed-loop control system robustness target $M_S^t$ with following relations:

$$\kappa_p = a_0 + a_1 {\tau_L}^{a_2}, \tag{9.3}$$

$$\tau_i = \frac{b_0 + b_1 \tau_L}{b_2 + b_3 \tau_L + {\tau_L}^2}, \tag{9.4}$$

$$\beta = 0. \tag{9.5}$$

The $a_i$ and $b_i$ constants for expressions (9.3) and (9.4) for five robustness levels, $M_S^t \in \{2.0,\ 3.0,\ 4.0,\ 5.0,\ 6.0\}$, are listed in Table 9.1 [1].

Relations (9.3)–(9.5) may only be used for models with normalized dead-times in the ranges listed in Table 9.2.

The robustness obtained with the MoReRT tuning Eqs. (9.3) and (9.4) are shown in Fig. 9.1.

**Table 9.1** MoReRT constants

| $M_S^t$ | 2.0 | 3.0 | 4.0 | 5.0 | 6.0 |
|---|---|---|---|---|---|
| $a_0$ | –1.149 | –0.5287 | –0.5091 | –0.4010 | –0.3995 |
| $a_1$ | 0.9560 | 0.8898 | 0.9986 | 1.010 | 1.070 |
| $a_2$ | –0.8468 | –0.9564 | –0.9525 | –0.9684 | –0.9559 |
| $b_0$ | 0.03242 | 0.004109 | –0.03222 | –0.01103 | –0.0226 |
| $b_1$ | 0.0 | 2.90 | 4.722 | 3.008 | 3.237 |
| $b_2$ | 0.08534 | 0.8081 | 1.40 | 1.023 | 1.101 |
| $b_3$ | –0.5698 | –2.166 | –3.10 | –2.285 | –2.347 |

**Table 9.2** UFOPDT models $\tau_L$ ranges for PI control

| $M_S^t$ | 2.0 | 3.0 | 4.0 | 5.0 | 6.0 |
|---|---|---|---|---|---|
| $\tau_{L\mathtt{min}}$ | 0.10 | 0.10 | 0.10 | 0.10 | 0.10 |
| $\tau_{L\mathtt{max}}$ | 0.25 | 0.35 | 0.45 | 0.50 | 0.55 |

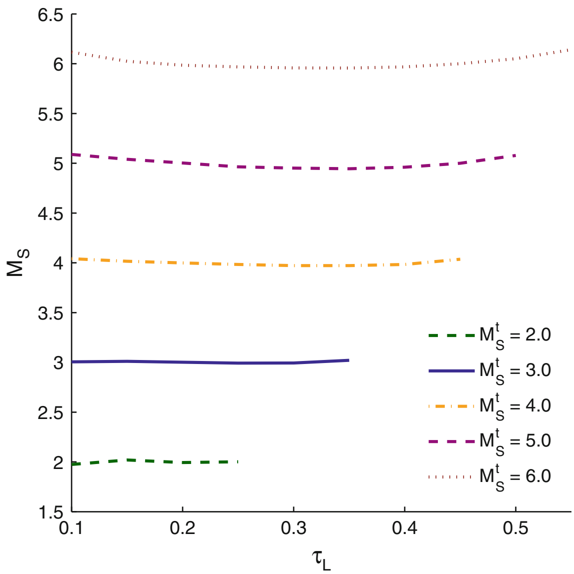


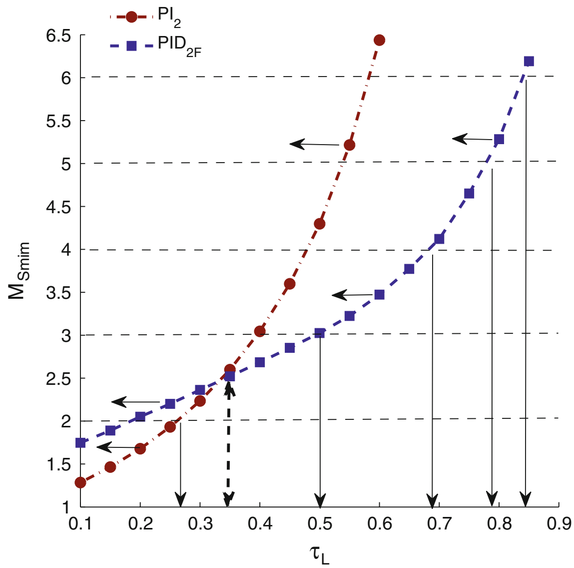

**Fig. 9.1** MoReRT PI controllers robustness for unstable first-order plus dead-time models

## 9.3 2DoF PID Controller with Filter

For unstable processes, the extension of the MoReRT tuning for 2DoF Ideal PID with filter ($PID_{2F}$) (2.21) controllers is presented in [3].

The closed-loop transfer function of the $PID_{2F}$ control of unstable controlled process (9.1) is of third-order. Then the target response is selected as

**Fig. 9.2** $PI(D)_2$ controllers maximum attainable robustness for unstable first-order plus dead-time models




$$y^t(s) = \frac{(\beta^* T_i^* s + 1)(T_f s + 1)\mathrm{e}^{-Ls}}{(\tau_c T s + 1)^3} r(s) + \frac{(T_i^*/K_p^*)(T_f s + 1)s\mathrm{e}^{-Ls}}{(\tau_c T + 1)^3} d(s). \quad (9.6)$$

As indicated above for the $PI_2$ case, the dead-time of the unstable process models impose severe constraints on the control system achievable robustness level and that a robust control system, $M_S \leq 2.0$, may only be obtained for a very limited range of unstable processes [1].

A comparison of the maximum robustness level attainable for unstable controlled processes using $PI_2$ controllers with target control system output (9.2) and $PID_{2F}$ controllers with target control system output (9.6) is shown in Fig. 9.2. As can be seen for models with low normalized dead-time, $\tau_L < 0.33$, the PI controller is capable to produce more robust control systems than the PID. On the other hand, for normalized dead-times $\tau_L \geq 0.33$ the PID controller expands the range of models that can be tuned with the same robustness level.

For the selection of the tuning parameters, the performance of the $PID_{2F}$ controller tuned to obtain the maximum possible robustness ($M_S$ minimum) and three constant robustness target levels ($M_S^t \in \{3.0,\ 4.0,\ 6.0\}$) is analyzed.

The results show that, if the control system over-damped output profile in (9.6) is maintained, the unstable model normalized dead-time does not adversely affects the performance and, more relevant, that in this case there is no performance/robustness tradeoff. Increasing the closed-loop control system robustness the performance, measured with the integrated absolute error, increases ($J_e$ decreases), but the control effort total variation increases. Then the highest performance is obtained with the more robust design.

The optimization also shows that for all the analyzed cases the proportional set-point weight $\beta^* \rightarrow 0$. Then, for the unstable processes the $PID_{2F}$ output (2.21) reduces to

$$u(s) = K_p^* \left\{ \frac{1}{T_i^* s} \left[ r(s) - \left( \frac{1}{T_f s + 1} \right) y(s) \right] - \left( \frac{T_d^* s + 1}{T_f s + 1} \right) y(s) \right\}. \tag{9.7}$$

Therefore, the resulting controller has only four adjustable parameters. For the development of the proposed tuning rule for UFOPDT models, the $PID_{2F}$ controller parameters that provide the maximum obtainable robustness are found.

The controller parameters obtained from the optimization procedure are used to fit the controller parameter equations of the proposed MoReRT approach for $PID_{2F}$ controllers applied to unstable first-order plus dead-time controlled processes.

The normalized $PID_{2F}$ controller parameters can be obtained with the following equations:

$$\kappa_p^* = 3.611 - 2.603\tau_L^{0.5343}, \tag{9.8}$$

$$\tau_i^* = 2.886 + 50.09\tau_L^{4.663}, \tag{9.9}$$

$$\tau_d^* = 0.345\tau_L^{0.9933}, \tag{9.10}$$

$$\tau_f = \frac{0.4337 - 0.2068\tau_L}{6.061 - 27.39\tau_L + 100\tau_L^2}, \tag{9.11}$$

$$\beta^* = 0, \tag{9.12}$$

$$\gamma^* = 0. \tag{9.13}$$

Relations (9.8)–(9.13) are valid for UFOPDT models with $0.1 \leq \tau_L \leq 0.85$.

The obtainable robustness with tuning parameters (9.8)–(9.11) can be estimated using the following relation:

$$M_S = \frac{101.2 + 107.5\tau_L - 219.3\tau_L^2}{72.07 - 79.71\tau_L + \tau_L^2}. \tag{9.14}$$

Using the $PID_{2F}$ parameters and the conversion relations in Chap. 2, the equivalent 2DoF Standard PID ($PID_2$) controller normalized parameters can be obtained with the following equations:

$$\kappa_p = 3.185 - 2.188\tau_L^{0.6658}, \tag{9.15}$$

$$\tau_i = 2.857 + 50.01\tau_L^{4.645}, \tag{9.16}$$

$$\tau_d = -0.3816 + 0.7151\tau_L^{0.4151}, \tag{9.17}$$

$$\alpha = \frac{3.013 - 2.794\tau_L}{0.894 - 26.03\tau_L + 100\tau_L^2}, \tag{9.18}$$

$$\beta = 0, \tag{9.19}$$

$$\gamma = 0. \tag{9.20}$$

It is found that for models with normalized dead-time $\tau_L < 0.22$, the equivalent controller derivative filter constant $\alpha$ turns negative and that for $0.22 \leq \tau_L < 0.25$ it is very high. This reduces the range of unstable models that can be robustly controlled with the equivalent $PID_2$ controller. Then, relations (9.15)–(9.20) are valid for UFOPDT models with $0.25 \leq \tau_L \leq 0.85$.

## 9.4 2DoF PID Controllers with Two Input Filters

The first step to follow now is a verification of the maximum control system robustness attainable with the $PI_{2IF}$ and $PID_{2IF}$ controllers [2]. This is shown in Fig. 9.3.

The attainable robustness levels are low in comparison with the best known picture for stable processes. We have to constrain ourselves to higher values of $M_S$. As an

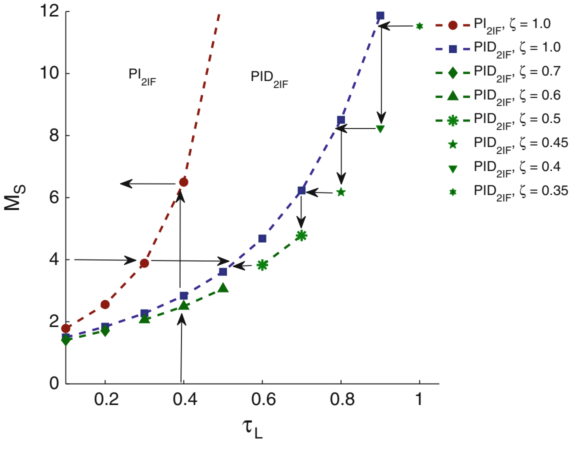



**Fig. 9.3** $PI_2$ and $PID_2$ controllers maximum attainable robustness for unstable first-order plus dead-time models

example for $\tau_L = 0.3$ maximum robustness are $M_{S\min} = 3.88$ ($PI_{2IF}$, $\zeta = 1.0$) and $M_{S\min} = 2.06$ ($PID_{2IF}$, $\zeta = 0.6$); for $\tau_L = 0.7$ it is only $M_{S\min} = 4.78$ ($PID_{2IF}$, $\zeta = 0.5$).

For a given process the maximum robustness with a $PI_{2IF}$ controller is obtainable for the critical damped closed-loop poles ($\zeta = 1.0$) but in all cases it is low than the robustness obtained with the $PID_{2IF}$.

For the $PID_{2IF}$ controller, it is possible to increase the control system robustness for a given process and to extend the range of processes that can be controlled with the same robustness reducing the closed-loop poles damping.

It is also found that if the closed-loop damping is reduced to a certain limit, the control system robustness increases as well almost all other indices. Then in this case, the maximum robustness (lower damping ratio) design procedure is recommended [2].

Tuning of the normalized parameters for maximum robustness for unstable first-order plus dead-time models with normalized dead-time $0.1 \le \tau_L \le 1.0$ obeys the fallowing relations:

$$\kappa_p = \frac{1}{0.3866 + 0.5494\tau_L - 0.0634\tau_L^2 + 0.0002184\tau_L^3}, \tag{9.21}$$

$$\kappa_i = \frac{1}{1.5422 - 4.9568\tau_L + 22.1484\tau_L^2}, \tag{9.22}$$

$$\kappa_d = 0.07318 + 0.6722\tau_L^{0.646}, \tag{9.23}$$

$$\tau_f = 0.08649 + 0.1452\tau_L^{1.132}, \tag{9.24}$$

$$\tau_r = 3.957 - 15.65\tau_L + 55.41\tau_L^2 - 26.58\tau_L^3, \tag{9.25}$$

$$\sigma = 1.156 - 1.684\tau_L + 3.562\tau_L^2 - 2.411\tau_L^3, \tag{9.26}$$

$$\gamma = 0. \tag{9.27}$$

The obtainable robustness with tuning parameters (9.21)–(9.27) can be estimated using the following relation:

$$M_S = \frac{1 + 1.4561\tau_L}{0.7656 - 0.2024\tau_L - 0.3848\tau_L^2 + 0.0003126\tau_L^3}. \tag{9.28}$$

## 9.5 Analysis of MoReRT Controllers

As an example of MoReRT control of unstable processes, consider the UFOPDT model

$$P_5(s) = \frac{e^{-0.2s}}{s-1}. \tag{9.29}$$

Next we will apply this to $PI_2$ and $PID_{2IF}$ controller and see the attainable performance and robustness with each option. This comparison provides some light on the quite often stated question of when to use a PI or a PID controller.

### 9.5.1 $PI_2$ Controller

Table 9.3 lists the robustness ($M_S$), performance ($J_e$), and control effort total variation ($TV_u$) obtained for five target robustness levels ($M_S^t$) to a set-point step change ($\Delta r$) and a disturbance step change ($\Delta d$) of the MoReRT $PI_2$ control of the unstable model (9.29). Figure 9.4 shows the corresponding responses to a 20 % set-point step change followed by a 10 % load-disturbance step change for four robustness levels.

Comparison of MoReRT (9.3)–(9.5) for UFOPDT models with other tuning rules is presented in [1].

### 9.5.2 $PID_{2IF}$ Controller

The $PID_{2IF}$ parameters in this example are listed in Table 9.4, and the resulting robustness and regulatory and servo-control performance in Table 9.5.

The maximum obtainable robustness with the $PID_{2IF}$ controller for this process ($\tau_L = 0.2$) estimated with (9.28) is $M_S = 1.819$ and $M_S = 1.83$ is obtained. If robustness is reduced to the usual minimum for stable process, $M_S = 2.0$, all the regulatory performance indices improve. They can be improved even more, with the exception of $TV_{ud}$, if the control system robustness is decreased more.

**Table 9.3** Unstable model example, robustness, performance, and control effort indices

| $M_S^t$ | $M_S$ | $J_{ed}/\Delta d$ | $TV_{ud}/\Delta d$ | $J_{er}/\Delta r$ | $TV_{ur}/\Delta r$ |
|---|---|---|---|---|---|
| 2.0 | 1.99 | 1.101 | 2.633 | 1.749 | 1.670 |
| 3.0 | 3.00 | 0.389 | 3.187 | 1.020 | 2.653 |
| 4.0 | 4.00 | 0.300 | 4.365 | 0.845 | 4.109 |
| 5.0 | 5.00 | 0.311 | 5.620 | 0.782 | 5.893 |
| 6.0 | 5.99 | 0.337 | 6.864 | 0.783 | 7.704 |

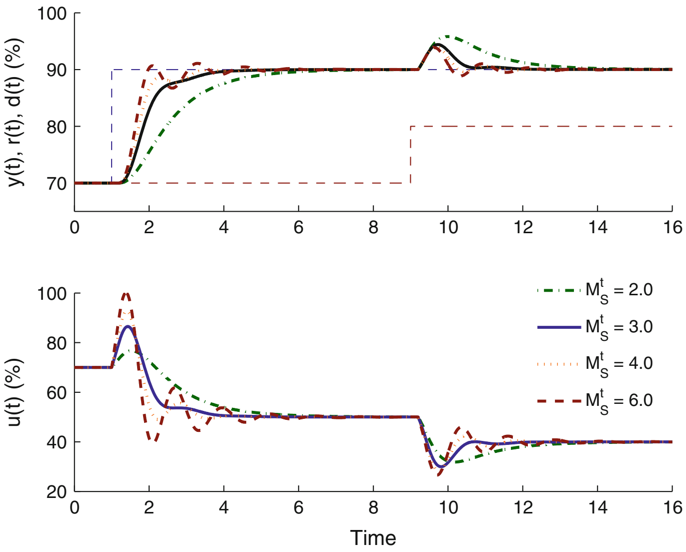



**Fig. 9.4** Unstable model example, $PI_2$ control system responses

**Table 9.4** $PID_{2IF}$ parameters

| $M_S^t$ | Min | 2.0 | 3.0 |
|---|---|---|---|
| $K_p$ | 2.025 | 2.974 | 4.625 |
| $K_i$ | 0.696 | 1.807 | 5.609 |
| $K_d$ | 0.311 | 0.380 | 0.520 |
| $T_f$ | 0.110 | 0.060 | 0.030 |
| $T_r$ | 2.907 | 1.621 | 0.819 |
| $\sigma$ | 0.963 | 1 | 1 |
| $\gamma$ | 0 | 0 | 0 |

For the same unstable process (9.29), the $PID_{2IF}$ controller outperforms the $PI_2$ controller in all indices except the control effort total variation to a set-point change.

Nyquist diagrams of $PID_{2IF}$ controllers for $M_{S\min}$ and $M_S^t = 2.0$ are shown in Fig. 9.5.

The control system responses to a 20 % set-point change followed by a 10 % disturbance step change are shown in Fig. 9.6.

Comparative analysis with other tuning rules for the control of unstable models are presented in [3] ($PID_{2F}$ controllers) and [2] ($PID_{2IF}$ controllers).

**Table 9.5** $PID_{2IF}$ performance

| $M_S^t$ | Min | 2.0 | 3.0 |
|---|---|---|---|
| $M_S$ | 1.83 | 2.00 | 3.00 |
| $K^{\infty}$ | 0 | 0 | 0 |
| $J_{ed}$ | 1.790 | 0.612 | 0.223 |
| $TV_{ud}$ | 3.055 | 2.317 | 2.805 |
| $E_{max}$ | 0.725 | 0.434 | 0.311 |
| $J_{er}$ | 1.849 | 1.172 | 0.715 |
| $TV_{ur}$ | 1.823 | 2.317 | 4.113 |
| $U_{max}$ | 0.272 | 0.573 | 1.417 |
| $\Delta u_0$ | 0.0 | 0.0 | 0.0 |

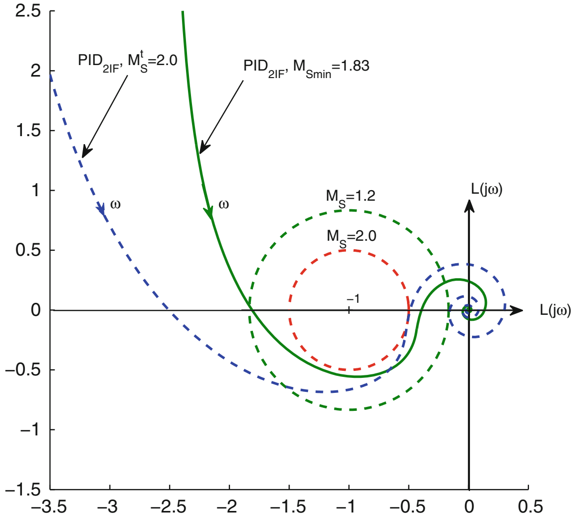



**Fig. 9.5** UFOPDT model example, $PID_{2IF}$ Nyquist diagram

## Chapter Remarks

For UFOPDT models with normalized dead-time $\tau_L \geq 0.33$, the $PID_{2F}$ controller allows to obtain control systems with higher robustness level than with the $PI_2$ counterpart. It also expands the normalized dead-time range of unstable models that can be controlled with certain robustness level.

For UFOPDT models, the $PID_{2F}$ design was based in obtain a control system with the highest possible robustness using over-damped target responses.

The $PID_{2IF}$ control of UFOPDT models suffer of the same robustness issue but provides two addition and important characteristics, no abrupt change at the controller output to a set-point step change and high frequency roll-off.

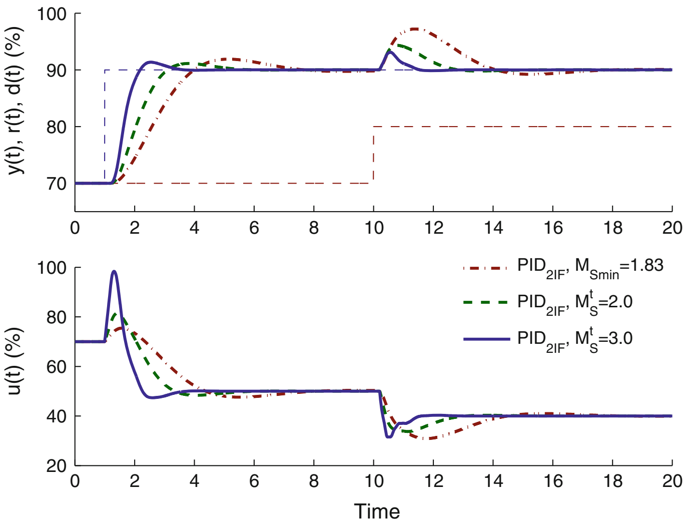



**Fig. 9.6** UFOPDT model, $PID_{2IF}$ control systems responses

# References


1. Alfaro, V.M., Vilanova, R.: Robustness-based tuning of two-degree-of-freedom proportionsal integral controllers for unstable processes. In: 16th International Conference on Systems Theory, Control and Computing (ICSTCC 2012), Sinaia, Romania. 12–14 Oct 2012
2. Alfaro, V.M., Vilanova, R.: Performance and robustness considerations for tuning of proportional integral/proportional integral derivative controllers with two input filters. Ind. Eng. Chem. Res. **52**, 18287–18302 (2013)
3. Alfaro, V.M., Vilanova, R.: Robust tuning of 2DoF PID controllers with filter for unstable first-order plus dead-time processes. In: 18th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA2013), Cagliari, Italy, 10–13 Sept 2013

# Chapter 10
# MoReRT Design Methodology Extensions

In previous chapters, the proposed *Model-Reference Robust Tuning* methodology was used to design standard feedback closed-loop control systems with 2DoF PI and PID controllers to control over-damped, inverse response, integrating, and unstable controlled processes. The control system block diagram considered in all these cases is depicted in Fig. 10.1.

The reference models used include a regulatory control closed-loop transfer function target $M^t_{yd}(s)$ and a servo-control closed-loop transfer function target $M^t_{yr}(s)$. Then the performance objective were the controlled variable response shapes to a disturbance and a set-point step changes with a robustness constraint measured with the maximum sensitivity $M_S$.

There may be however situations were either the main variable of interest is not the controlled variable or when the control system considered has a different topology with other transfer functions involved. In such case, the MoReRT approach can also be applied. The main idea is to keep the core aspects of MoReRT such as the specification of the desired target responses of interest and to solve the corresponding optimization problem that provides the controller that best matches the closed-loop responses with the target ones while respecting the desired robustness level for the feedback loop. To be able to extend this basic idea makes MoReRT a flexible approach able to face a great variety of control design problems.

In this chapter, three extensions of the MoReRT methodology are proposed, even there may be a wide range of other situations where it could also be applied. The proposed extensions are:

1. The use of controller output (control effort) target responses to a disturbance and set-point step changes.
2. Consider that the path from the disturbance to the controlled variable is different to the path from the controller output to the controlled variable.
3. The robust tuning of dead-time compensating controllers.

Here it is only shown how to proceed in order to solve these approaches. Not an extensive analysis and coverage is developed nor presented here.



V.M. Alfaro and R. Vilanova, *Model-Reference Robust Tuning of PID Controllers*,
Advances in Industrial Control, DOI 10.1007/978-3-319-28213-8_10

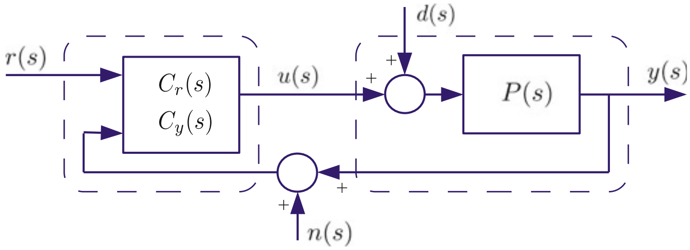



**Fig. 10.1** 2DoF closed-loop control block diagram

## 10.1 MoReRT Design with Control Effort Specifications

When facing the problem of designing a closed-loop control system, one of the first aspects to distinguish is that of the control and manipulated variables. Being of primary interest to keep the controlled variable at some desired value is what promotes to state the controller design problems in terms of the behavior of such signal. There is however the obvious point that the only way the controller has to affect and therefore determine the variations of the controlled variable is by acting on the process manipulated variable, the control effort. Therefore, the real degree of freedom is that of the possibility of changing the control variable.

Not all control signals are realistic being one of the primary aspects to bear in mind that of the control signal smoothness. This can be dealt with a posterior analysis using an appropriate evaluation metric. However, another way of looking at this problem is that of facing directly the design of the control signal profile instead that of the controlled variable. This section shows how this can be dealt with the use of the MoReRT approach.

From Fig. 10.1 the noise free control effort is given by

$$u(s) = M_{ur}(s)r(s) + M_{ud}(s)d(s), \tag{10.1}$$

where

$$M_{ud}(s) = \frac{-C_y(s)P(s)}{1 + C_y(s)P(s)}, \tag{10.2}$$

is the disturbance to control effort ($d \rightarrow u$) closed-loop transfer function, and

$$M_{ur}(s) = \frac{C_r(s)}{1 + C_y(s)P(s)}, \tag{10.3}$$

the corresponding set-point to control effort ($r \rightarrow u$) closed-loop transfer function.

Following the same procedure outlined in Chap. 4, the controlled process model and the controller transfer functions are expressed as a quotient of polynomials in $s$ as follows:

$$P(s) = \frac{N_p^-(s)N_p^+(s)}{D_p(s)}, \tag{10.4}$$

$$C_y(s) = \frac{N_{cy}(s)}{D_{cy}(s)}, \tag{10.5}$$

$$C_r(s) = \frac{N_{cr}(s)}{D_{cr}(s)}, \tag{10.6}$$

where $N_p^+(s)$ is the controlled process model non-minimum phase part (dead-time and/or right-half plane zeros).

Replacing $P(s)$, $C_y(s)$, and $C_r(s)$ in (10.2) and (10.3) by (10.4)–(10.6) the $d \to u$ closed-loop transfer function can be expressed by

$$M_{ud}(s) = \frac{-N_{cy}(s)N_p^-(s)N_p^+(s)}{D_{cy}(s)D_p(s) + N_{cy}(s)N_p^-(s)N_p^+(s)}, \tag{10.7}$$

and the $r \to u$ closed-loop transfer function by

$$M_{ur}(s) = \left(\frac{D_{cy}(s)}{D_{cr}(s)}\right) \frac{N_{cr}(s)D_p(s)}{D_{cy}(s)D_p(s) + N_{cy}(s)N_p^-(s)N_p^+(s)}. \tag{10.8}$$

For the particular case of a $PI_2$ controller $D_{cr}(s) = D_{cy}(s)$, then (10.8) reduces to

$$M_{ur}(s) = \frac{N_{cr}(s)D_p(s)}{D_{cy}(s)D_p(s) + N_{cy}(s)N_p^-(s)N_p^+(s)}. \tag{10.9}$$

### ***$PI_2$ Control of Over-Damped Processes***

The general second-order over-damped controlled process model transfer function is given by

$$P(s) = \frac{K\mathrm{e}^{-Ls}}{(Ts+1)(aTs+1)}, \quad \theta_p = \{K, T, a, L\}, \tag{10.10}$$

and the two-degree-of-freedom proportional integral controller output by

$$u(s) = K_p\left(\frac{\beta T_i s + 1}{T_i s}\right) r(s) - K_p\left(\frac{T_i s + 1}{T_i s}\right) y(s), \quad \theta_c = \left\{K_p, T_i, \beta\right\}. \tag{10.11}$$

In this case the closed-loop transfer functions are of third-order.

### **Target Models**

The closed-loop transfer functions targets for the controller effort (10.7) and (10.9) are selected as

$$M_{ud}^t(s) = \frac{-(T_i s + 1)\,\mathrm{e}^{-Ls}}{(\tau_c^2 T^2 s^2 + 2\zeta\tau_c T s + 1)(a\tau_c T s + 1)}, \tag{10.12}$$

$$M_{ur}^t(s) = \frac{(1/K)(\beta T_i s + 1)(Ts + 1)}{(\tau_c^2 T^2 s^2 + 2\zeta \tau_c Ts + 1)(a\tau_c Ts + 1)}. \tag{10.13}$$

with control system design parameters $\theta_d = \{\zeta, \tau_c\}$.

For the particular case of a first-order plus dead-time (FOPDT) model ($a = 0$) and closed-loop transfer functions targets with real poles only ($\zeta = 1$) the control effort target is given by

$$u^t(s) = \frac{(1/K)(\beta T_i s + 1)(Ts + 1)}{(\tau_c Ts + 1)^2} r(s) - \frac{(T_i s + 1)\, \mathrm{e}^{-Ls}}{(\tau_c Ts + 1)^2} d(s). \tag{10.14}$$

### Cost Functionals

To obtain the controller parameters, that best match the target response (10.14) in the *least-squares sense*, a minimization procedure is used based on the differences between the controller target responses and the actual ones.

The overall cost functional to be optimized is defined as follows:

$$\begin{aligned} J_{uT}(\theta_p, \theta_c, \theta_d) \doteq & \int_0^\infty \left[u_r^t(\theta_p, \theta_c, \theta_d, t) - u_r(\theta_p, \theta_c, t)\right]^2 \mathrm{d}t \\ & + \int_0^\infty \left[u_d^t(\theta_p, \theta_{cy}, \theta_d, t) - u_d(\theta_p, \theta_{cy}, t)\right]^2 \mathrm{d}t. \end{aligned} \tag{10.15}$$

Using (10.15) the controller parameters $\theta_c^o$ are obtained such that

$$J_{uT}^o \doteq J_{uT}(\theta_p, \theta_c^o, \theta_d) = \min_{\theta_c} J_{uT}(\theta_p, \theta_c, \theta_d), \tag{10.16}$$

for design parameters $\theta_d$ selected in such a way that the control system robustness matches a target value measured using the maximum sensitivity, $M_S$.

#### *Example—PI$_2$ Control of a FOPDT Process*

Consider the first-order plus dead-time process given by the transfer function

$$P(s) = \frac{\mathrm{e}^{-0.5s}}{s + 1} \tag{10.17}$$

and (10.14) control effort target responses.

The design parameter $\tau_c$ is selected in a suitable range to obtain control systems with robustness $2.0 \geq M_s \geq 1.4$.

All three controller parameters $\{K_p, T_i, \beta\}$ are obtained at once optimizing the cost function (10.15). Controller gain and integral time are shown in Fig. 10.2. From the optimization data, the set-point proportional weight $\beta = 0$ in all cases. This figure also shows the resulting control system robustness.

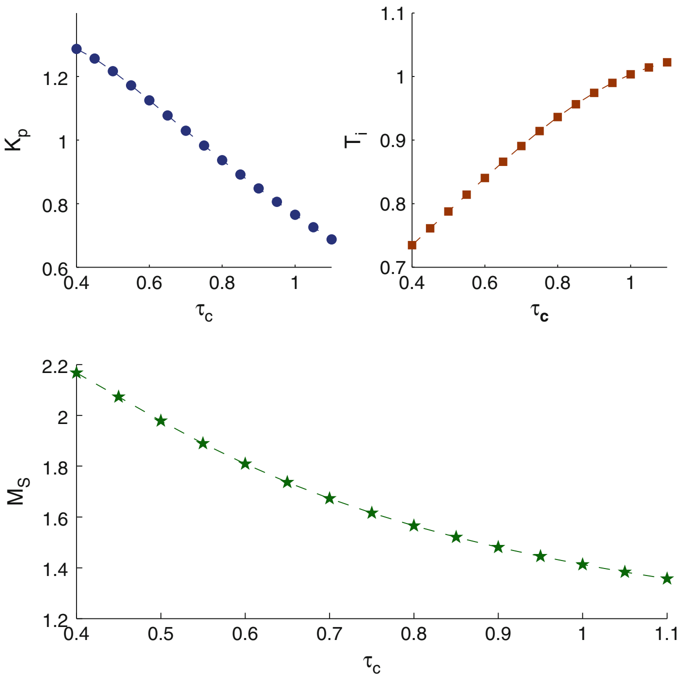



**Fig. 10.2** Example (FOPDT model)—proportional integral controller parameters and robustness

Variation of the design parameters $\tau_c$ allows to deal with the trade-off between performance and robustness. The control system can turned more robust reducing the controller gain and increasing its integral time. As a result of having $\beta = 0$, there is no controller output abrupt changes to a set-point step variation.

The controlled variable and control effort responses to unitary set-point and disturbance changes are shown in Fig. 10.3. It is worth to note here that we can look at the behavior of the controlled signal. However, what we are really determining with this design is the profile of the control signal. As expected as the desired robustness is increased, the control signal gets smoother.

***PID$_{2F}$ Control of Second-Order Over-Damped Controlled Processes***

Now the controlled process (10.10) is controlled with a 2DoF Ideal PID with filter whose output is

$$u(s) = K_p^* \left( \beta^* + \frac{1}{T_i^* s} \right) r(s) - K_p^* \left( 1 + \frac{1}{T_i^* s} + T_d^* s \right) \left( \frac{1}{T_f s + 1} \right) y(s), \quad (10.18)$$

with parameters $\theta_c^* = \left\{ K_p^*, T_i^*, T_d^*, T_f, \beta^*, \gamma^* = 0 \right\}$.

The control system is now of fourth-order and the target controlled output models (10.7) and (10.8) are selected as

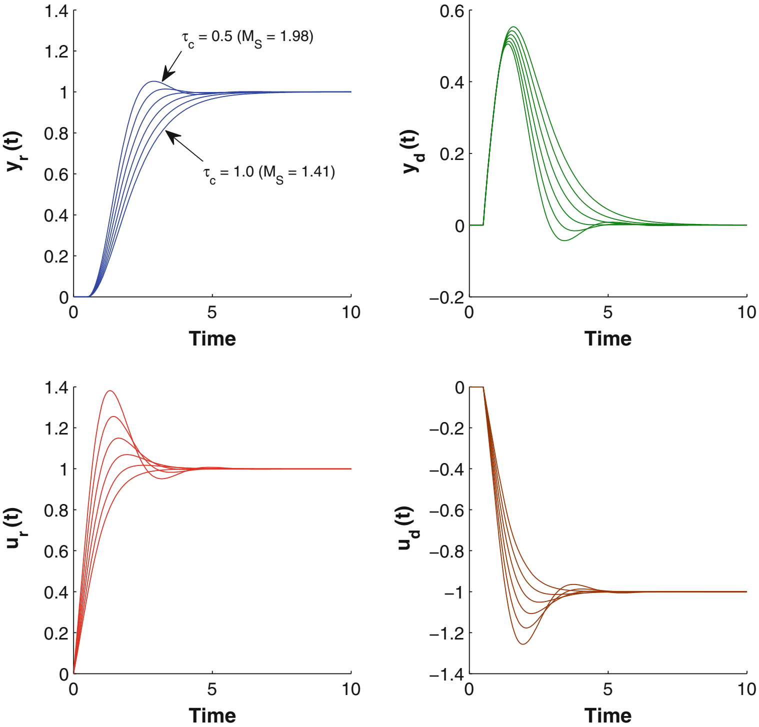



**Fig. 10.3** Example (FOPDT model)—control system and $PI_2$ controller output responses

$$M^t_{ud}(s) = \frac{-(T_i^* T_d^* s^2 + T_i^* s + 1)\,\mathrm{e}^{-Ls}}{(\tau_c T s + 1)^3 (a\tau_c T s + 1)}, \tag{10.19}$$

$$M^t_{ur}(s) = \frac{(1/K)(T_f^* s + 1)(\beta T_i^* s + 1)(T s + 1)(a T s + 1)}{(\tau_c T s + 1)^3 (a\tau_c T s + 1)}. \tag{10.20}$$

Then the controller output target used for tuning the $PID_{2F}$ controller is

$$u^t(s) = u^t_r(s) + u^t_d(s) = M^t_{ur}(s) r(s) + M^t_{ud}(s) d(s). \tag{10.21}$$

### *Example—$PID_{2F}$ Control of a SOPDT Process*

The controlled process has the following transfer function

$$P(s) = \frac{\mathrm{e}^{-0.5s}}{(s+1)(0.75s+1)}. \tag{10.22}$$

**Table 10.1** Example (SOPDT model)—$PID_{2F}$ controller parameters

| $M_S^t$ | $K_p^*$ | $T_i^*$ | $T_d^*$ | $T_f$ | $\beta^*$ |
|---|---|---|---|---|---|
| 2.0 | 2.351 | 1.481 | 0.451 | 0.0226 | 0 |
| 1.8 | 2.056 | 1.574 | 0.435 | 0.0218 | 0 |
| 1.6 | 1.686 | 1.651 | 0.415 | 0.0208 | 0 |
| 1.4 | 1.181 | 1.666 | 0.374 | 0.0187 | 0 |

If the controller parameters, $\theta_c^* = \left\{K_p^*, T_i^*, T_d^*, T_f, \beta^*, \gamma^* = 0\right\}$, are free during the cost functional optimization, it is found that $\beta \approx 0$ and that $T_f \to 0$ (the $PID_{2F}$ controller tends to a non-proper Ideal PID controller). Then the proportional set-point weight is set $\beta = 0$ and a lower limit $T_f \geq 0.05T_d$ is used for the controller filter time constant.

The $PID_{2F}$ controller parameters for four robustness target levels are listed in Table 10.1.

The control system responses to a 20 % set-point step change followed by a 10 % disturbance step change are shown in Fig. 10.4. All controlled variable responses are very smooth and without oscillation. The controller effort has no abrupt changes.

Table 10.2 lists control system evaluation for the four robustness target levels. This shows the performance/robustness trade-off.

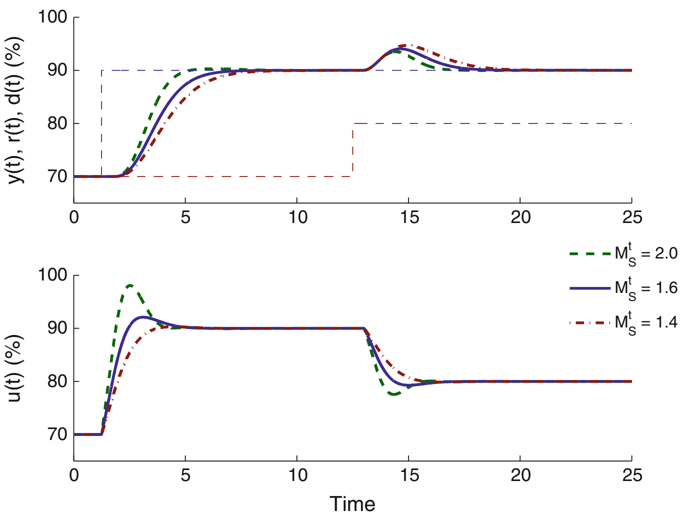



**Fig. 10.4** $PID_{2F}$ control system responses

**Table 10.2** Example (SOPDT model)—$PID_{2F}$ controller evaluation

| $M_S^t$ | $J_{ed}$ | $J_{er}$ | $TV_{ud}$ | $TV_{ur}$ | $E_{maxd\%}$ | $t_{a5\,\%d}$ |
|---|---|---|---|---|---|---|
| 2.0 | 0.065 | 0.431 | 0.153 | 0.365 | 3.50 | 4.147 |
| 1.8 | 0.077 | 0.467 | 0.132 | 0.298 | 3.71 | 4.709 |
| 1.6 | 0.098 | 0.522 | 0.115 | 0.242 | 4.04 | 5.437 |
| 1.4 | 0.141 | 0.612 | 0.103 | 0.207 | 4.68 | 6.591 |

## 10.2 Use of a Different Load Disturbance Path

In control system design, usually it is supposed that the controlled variable dynamics to a load disturbance and the one to the controller output are the same. This is a common situation in process control as pointed in [1] but there are also practical situations where the main load disturbance path is not the same as the manipulated variable path. For these cases, the closed-loop block diagram of Fig. 10.1 must be redraw as shown in Fig. 10.5 where $P_d(s) \neq P_u(s)$.

Now the controlled variable is given by

$$y(s) = \frac{C_r(s)P_u(s)}{1 + C_y(s)P_u(s)} r(s) + \frac{P_d(s)}{1 + C_y(s)P_u(s)} d(s). \tag{10.23}$$

Expressing the controlled process models and the controllers transfer functions as a quotient of polynomials in $s$ the regulatory control closed-loop transfer function is

$$M_{yd}(s) = \frac{D_{pu}(s)}{D_{pd}(s)} \left( \frac{D_{cy}(s)N_{pd}^{-}(s)N_{pd}^{+}(s)}{D_{cy}(s)D_{pu}(s) + N_{cy}(s)N_{pu}^{-}(s)N_{pu}^{+}(s)} \right). \tag{10.24}$$

If $P_d(s) = P_u(s) = P(s)$ (10.24) is the same (4.15).

The servo-control closed-loop transfer function is given by

$$M_{yr}(s) = C_r(s) \left( \frac{P_u(s)}{P_d(s)} \right) M_{yd}(s), \tag{10.25}$$

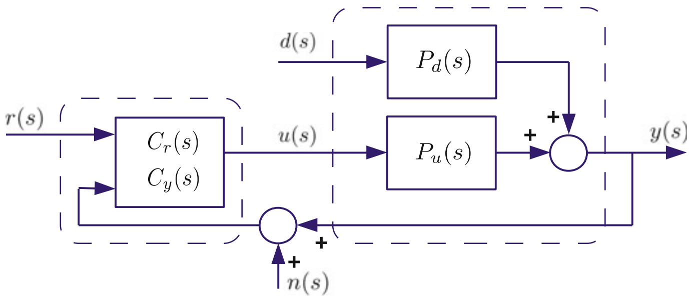



**Fig. 10.5** General 2DoF closed-loop control system block diagram

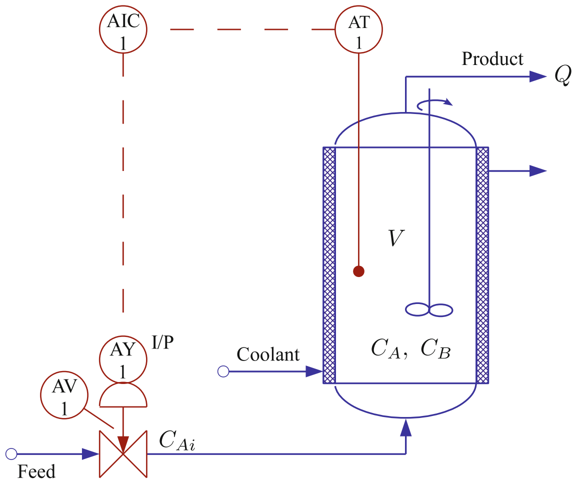



**Fig. 10.6** Continuous steering tank reactor control system

that can also be expressed as

$$M_{yr}(s) = C_r(s)\left(\frac{D_{cy}(s)N_{pu}^{-}(s)N_{pu}^{+}(s)}{D_{cy}(s)D_{pu}(s) + N_{cy}(s)N_{pu}^{-}(s)N_{pu}^{+}(s)}\right). \tag{10.26}$$

Due to $P_u(s)$ and $P_d(s)$ can represent any process dynamics, it is not possible to obtain general rules for controller tuning. In this case, the control design procedure is illustrated using a particular process. This way, the following example will also serve as a practical example of application of the MoReRT methodology.

***Control of a Nonlinear CSTR***

Figure 10.6 shows the control system of a nonlinear continuous stirred-tank reactor (CSTR) where takes place a series-parallel Van de Vusse reaction given by [2]:

$$A \xrightarrow{k_1} B \xrightarrow{k_2} C, \tag{10.27}$$

$$2A \xrightarrow{k_3} D. \tag{10.28}$$

From a mass balance and assuming a constant volume reactor, the concentrations of components A and B in the reactor are given by the following nonlinear differential equations:

$$\frac{\mathrm{d}C_A(t)}{\mathrm{d}t} = -k_1 C_A(t) - k_3 C_A^2(t) + \frac{1}{V}\left[C_{A_i} - C_A(t)\right] Q(t), \tag{10.29}$$

$$\frac{\mathrm{d}C_B(t)}{\mathrm{d}t} = k_1 C_A(t) - k_2 C_B(t) - \frac{1}{V} C_B(t) Q(t). \tag{10.30}$$

where $C_A$ and $C_B$ are the concentrations of components A and B, respectively, $k_1$, $k_2$, and $k_3$ the reaction rates of the three reactions, $V$ the reactor volume, $C_{A_i}$ the concentration of product A in the input flow, and $Q$ the volumetric flow rate through the reactor.

The controlled variable is the product B concentration $C_B(t)$ and the manipulated variable is the flow rate $Q(t)$. Disturbances in the component A concentration $C_{A_i}$ and inlet flow rate $Q(t)$ are also taken into account.

Using the particular reaction considered in [3] with constants $k_1 = 5/6\ \text{min}^{-1}$, $k_2 = 5/3\ \text{min}^{-1}$, $k_3 = 1/6\ \text{L/(gmol)/min}$, $C_{A_i} = 10\ \text{gmolL}^{-1}$ for a reactor tank of $V = 70$ L, the steady-state concentrations for a $Q_o = 40\ \text{L min}^{-1}$ flow rate are $C_{A_o} = 3\ \text{gmolL}^{-1}$ and $C_{B_o} = 1.117\ \text{gmolL}^{-1}$. Considering the time scale of the process, the transmitter and final control element are represented by simple gains $K_t = 50\,\%/(\text{gmol/L})$ and $K_v = 0.80\ (\text{L/min})/\%$, respectively.

### Models Identification

Figure 10.7 shows the controlled variable responses to ±5 % changes in the controller output ($u \rightarrow y$) from where the $P_u(s)$ can be obtained. The process shows to have an inverse response dynamics.

Using the identification method in [4] and considering both the increment and the decrement in controller output the $P_u(s)$ model is

$$P_u(s) = \frac{0.34(-0.38s + 1)}{(0.56s + 1)(0.31s + 1)}, \tag{10.31}$$

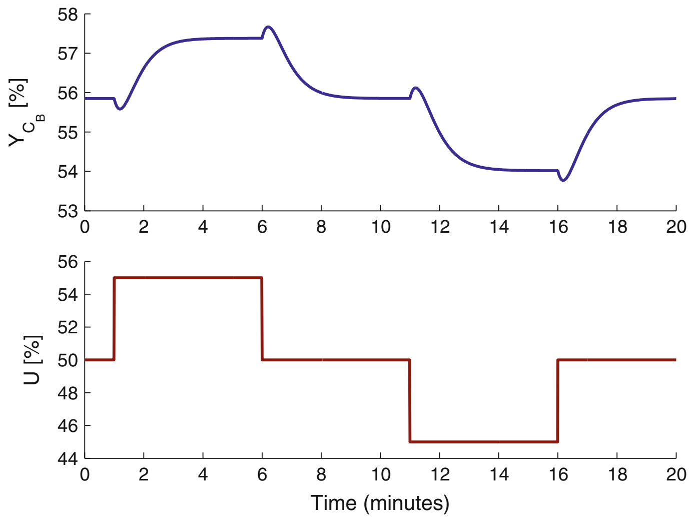



**Fig. 10.7** CSTR $U \rightarrow Y$ dynamics for $P_u(s)$ identification

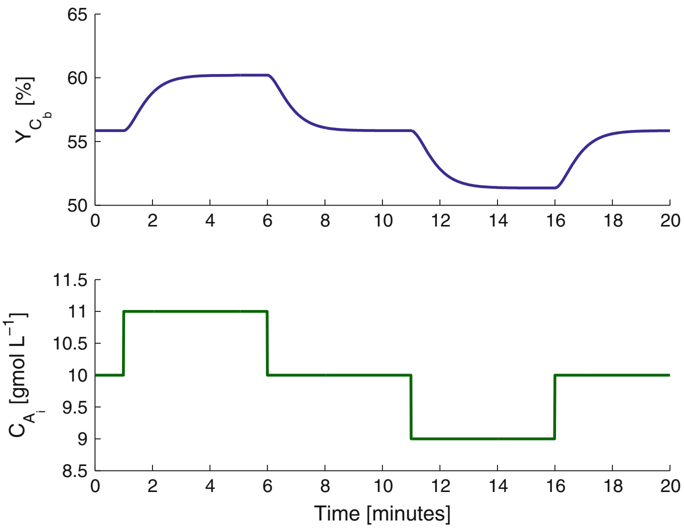



**Fig. 10.8** CSTR $C_{A_i} \to Y$ dynamics for $P_d(s)$ identification

with gain in %/% and time constants in minutes, $\theta_{P_u} = \{K_{P_u} = 0.34, T_{P_u} = 0.56, a_{P_u} = 0.554, b_{P_u} = 0.679\}$.

The controlled variable responses to $\pm 1$ gmolL$^{-1}$ changes in the input flow concentration ($d \to y$) are shown in Fig. 10.8 and is used for $P_d(s)$ identification. The process dynamics to a change in the main disturbance is over-damped.

Now using the identification methods in [5] and considering again the increment and the decrement in the disturbance, the obtained $P_d(s)$ model is a dual pole given by the transfer function

$$P_d(s) = \frac{4.43}{(0.43s + 1)^2}. \tag{10.32}$$

with gain in $\%/(\text{gmolL}^{-1})$ and time constants in minutes, $\theta_{P_d} = \{K_{P_d} = 0.443, T_{P_d} = 0.43, a_{P_d} = 1\}$.

For the CSTR process $P_d(s)$ is very different from $P_u(s)$.

The block diagram of the CSTR linear model is shown in Fig. 10.9.

A change in the inlet flow concentration $\delta C_{A_i}$ is considered the main disturbance but it is also possible to have changes in the inlet flow $\delta Q$ due to external influences.

The control system controlled variable is given then by the following relation

$$y(s) = \frac{C_r(s)P_u(s)}{1 + C_y(s)P_u(s)} r(s) + \frac{P_d(s)}{1 + C_y(s)P_u(s)} \delta C_{A_i}(s) + \frac{1/K_v P_u(s)}{1 + C_y(s)P_u(s)} \delta Q(s). \tag{10.33}$$

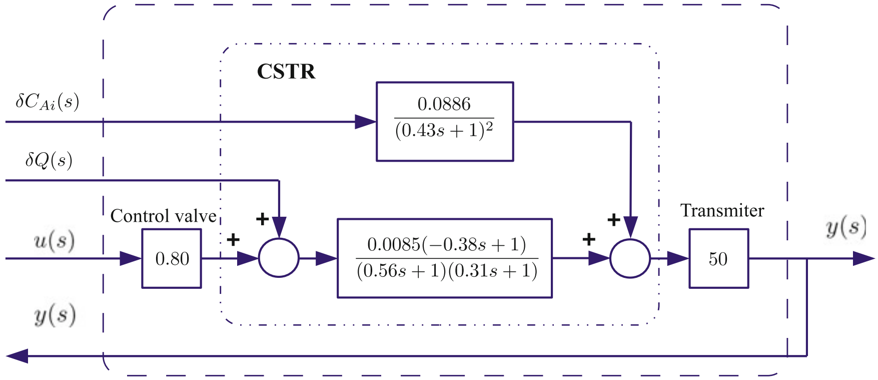



**Fig. 10.9** CSTR linear model

### Control Algorithm

The controller chosen is a two-degree-of-freedom standard proportional integral derivative control algorithm ($PID_2$) given by the output equation

$$u(s) = K_p \left\{ \beta r(s) - y(s) + \frac{1}{T_i s} \left[ r(s) - y(s) \right] - \left( \frac{T_d s}{\alpha T_d s + 1} \right) y(s) \right\}, \qquad (10.34)$$

with parameters $\theta_c = \left\{ K_p, T_i, T_d, \alpha = 0.1, \beta, \gamma = 0 \right\}$.

### Target Closed-Loop Transfer Functions

The CSTR $PID_2$ control system is of third-order.

For the design, and using (10.24) and (10.26), the closed-loop target transfer functions are selected with two under-damped poles as:

$$M^t_{yr}(s) = \frac{(\beta T_i s + 1)(-0.38s + 1)}{(0.3136\tau_c^2 s^2 + 1.23\zeta \tau_c s + 1)(0.31\tau_c s + 1)}, \qquad (10.35)$$

$$M^t_{y\delta C_{A_i}}(s) = \frac{13.03(T_i/K_p)s(0.56s + 1)(0.31s + 1)}{(0.43s + 1)^2(0.3136\tau_c^2 s^2 + 1.23\zeta \tau_c s + 1)(0.31\tau_c s + 1)}, \qquad (10.36)$$

$$M^t_{y\delta Q}(s) = \frac{1.25(T_i/K_p)s(-0.38s + 1)}{(0.3136\tau_c^2 s^2 + 1.23\zeta \tau_c s + 1)(0.31\tau_c s + 1)}, \qquad (10.37)$$

with $\zeta = 0.70$.

Considering that the main disturbance is the inlet flow concentration $C_{A_i}$, the overall control system response target used to obtain the controller parameters is

$$y^t(s) = M^t_{yr}(s)r(s) + M^t_{y\delta C_{A_i}}(s)\delta C_{A_i}(s). \qquad (10.38)$$

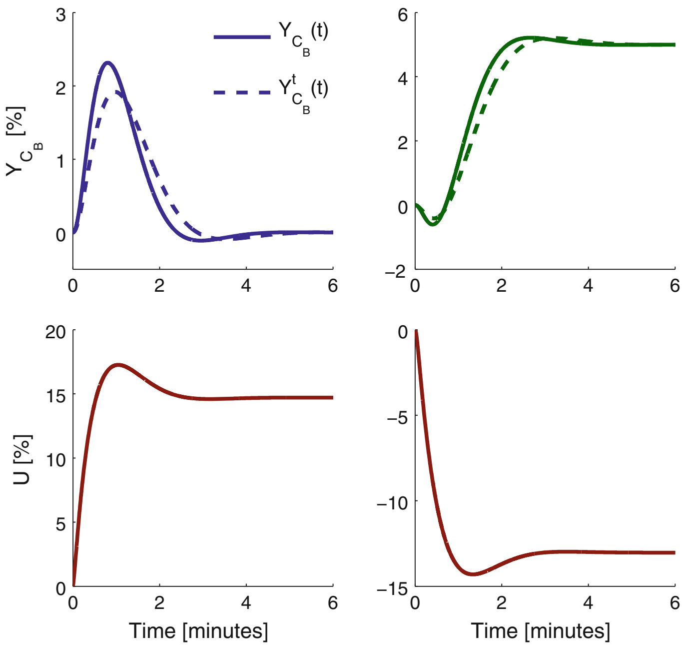



**Fig. 10.10** Target and resulting control system responses, $M_S^t = 2.0$

## Robust Controller Parameters

All the $PID_2$ controller parameters are obtained at once selecting the design parameter $\tau_c$ to obtain a closed-loop control system with a desired robustness $M_S^t$.

For $M_S^t = 2.0$ the controller parameters are: $K_p = 3.335$, $T_i = 0.685$ min, $T_d = 0.181$ min, $\beta = 0$ ($\alpha = 0.1$, $\gamma = 0$).

The target and resulting control system responses to a $\delta C_{A_i} = 1$ gmolL$^{-1}$ step change (left column) and a $\delta R = 5\,\%$ step change (right column) are shown in Fig. 10.10. As noticed there are only little differences between the actual control system responses and the target ones.

To increase the control system robustness up to $M_S^t = 1.6$, the target responses are made slower increasing the design parameter $\tau_c$. Now the controller parameters are: $K_p = 2.372$, $T_i = 0.663$ min, $T_d = 0.162$ min, $\beta = 0$ ($\alpha = 0.1$, $\gamma = 0$).

The new target and resulting control system responses to a $\delta C_{A_i} = 1$ gmolL$^{-1}$ step change (left column) and a $\delta R = 5\,\%$ step change (right column) are shown in Fig. 10.11. The obtained responses match exactly the ones from the target closed-loop transfer functions.

From the design process, it is found that due to the inverse response dynamic characteristics of the CSTR the control system robustness can not be increased beyond $M_S = 1.53$.

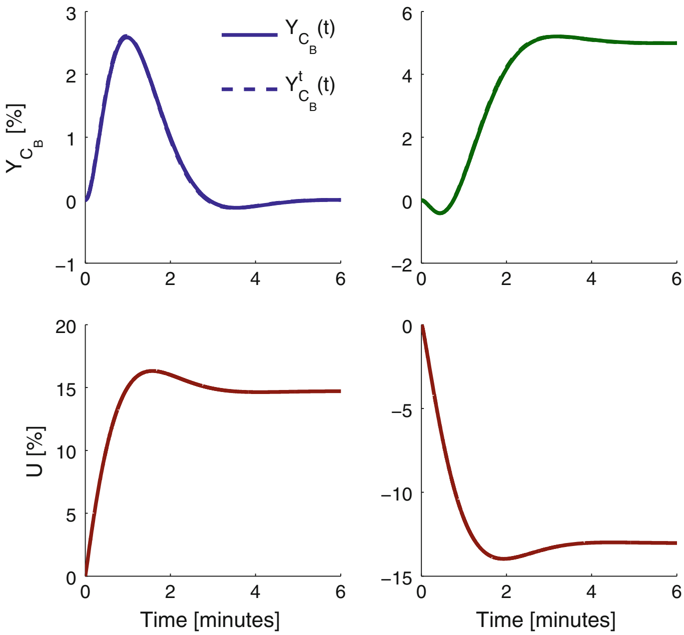



**Fig. 10.11** Target and resulting control system responses, $M_S^t = 1.6$

The $PID_2$ ($M_S = 1.6$) CSTR control system response to a 10 % set-point step change ($\delta R(C_B) = 10\,\%$) followed by a 10 % change in the inlet flow concentration ($\delta C_{A_i} = 1 \mathrm{gmolL}^{-1}$) and by a –10 % change in the inlet flow ($\delta Q = -4$ gmolL$^{-1}$) is shown in Fig. 10.12.

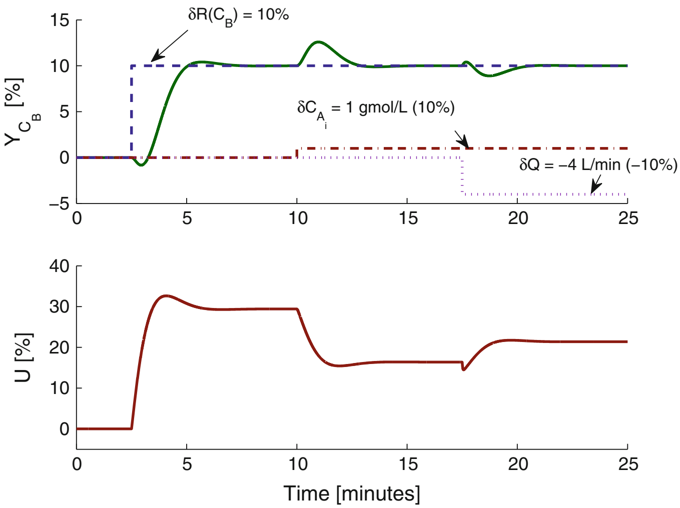



**Fig. 10.12** Control system responses to set-point and disturbances ($M_S = 1.6$)

## 10.3 Robust Tuning of Two-Degree-of-Freedom Dead-Time Compensating Controllers

It is well known that controlled processes with long dead-time are difficult to control and that the performance obtained controlling this type of processes with a standard PI or PID controllers usually is not satisfactory. A dead-time compensating controller is required.

For processes with long dead-times prediction using the derivative of the feedback signal does not provide enough information of its future variations then a PI controller is used and the prediction is based on the controller output and a controlled process model.

### *PI Controller with a Smith Predictor*

The most well known dead-time compensating control scheme is the seminal model-based *Smith Predictor* [6]. It is showed in Fig. 10.13 together with a two-degree-of-freedom proportional integral controller and denoted as $PI_{SP2}$. It has been followed by a variety of modified dead-time compensating control arrangements [7].

The measured controlled variable, the response of the controlled process model, and a prediction of the future controlled process response ($L_m$ times earlier) are feed back to the PI controller.

In the ideal case, with no modeling errors, the PI controller operates over a process without dead-time.

Tuning a $PI_{SP2}$ controller requires to select the proportional integral control algorithm parameters $\theta_c = \{K_p, T_i, \beta\}$ and the parameters $\theta_{cm} = \{K_m, T_m, L_m\}$ of the first-order-plus dead time model used in the predictor. Then, six parameters need to be adjusted, that usually is not a simple task.

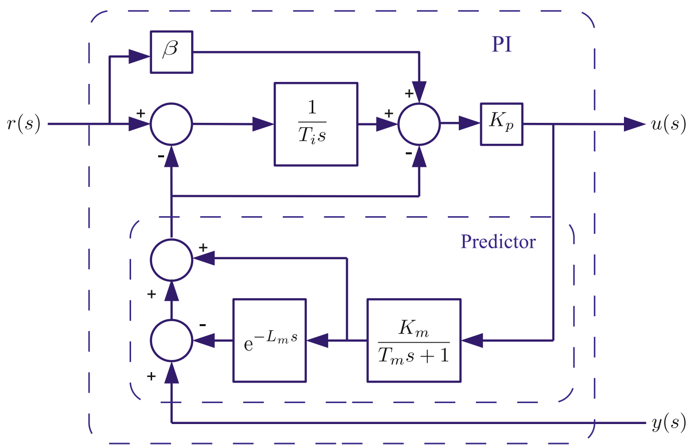



**Fig. 10.13** Proportional integral controller with a Smith predictor

The $PI_{SP2}$ controller output is

$$u(s) = K_p \left\{ \beta r(s) - y'(s) + \frac{1}{T_i s}[r(s) - y'(s)] \right\}, \tag{10.39}$$

where the feedback input to the PI controller is

$$y'(s) = y(s) + \left( \frac{K_m}{T_m s + 1} - \frac{K_m e^{-L_m s}}{T_m s + 1} \right) u(s). \tag{10.40}$$

Combining (10.39) and (10.40) the $PI_{SP2}$ controller output can be expressed for the analysis as

$$u(s) = C_{sp}(s) \left\{ C_r(s) r(s) - C_y(s) y(s) \right\}, \tag{10.41}$$

with

$$C_r(s) = K_p \left( \frac{\beta T_i s + 1}{T_i s} \right), \tag{10.42}$$

$$C_y(s) = K_p \left( \frac{T_i s + 1}{T_i s} \right), \tag{10.43}$$

$$C_{sp}(s) = \frac{T_m s + 1}{T_m s + 1 + K_m (T_i s + 1)(1 - e^{-L_c s})}, \tag{10.44}$$

where $C_{sp}(s)$ is the Smith predictor transfer function.

The closed-loop control system with a $PI_{SP2}$ dead-time compensating controller is shown in Fig. 10.14.

From this block diagram and using (10.41), the controlled variable as a function of the set-point and of the disturbance is given by

$$y(s) = \frac{C_{sp}(s) C_r(s) P(s)}{1 + C_{sp}(s) C_y(s) P(s)} r(s) + \frac{P(s)}{1 + C_{sp}(s) C_y(s) P(s)} d(s). \tag{10.45}$$

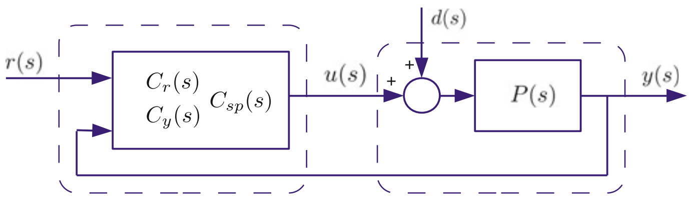



**Fig. 10.14** $PI_{SP2}$ feedback control system

Expressing the controlled process model, the controllers, and the predictor transfer functions as quotients in $s$

$$P(s) = \frac{N_p^-(s)N_p^+(s)}{D_p(s)},\ C_r(s) = \frac{N_{cr}(s)}{D_{cr}(s)},\ C_y(s) = \frac{N_{cy}(s)}{D_{cy}(s)},\ C_{sp}(s) = \frac{N_{sp}(s)}{D_{sp}(s)}, \tag{10.46}$$

the regulatory closed-loop transfer function can be expressed by

$$M_{yd}(s) = \frac{D_{sp}(s)D_{cy}(s)N_p^-(s)N_p^+(s)}{D_{sp}(s)D_{cy}(s)D_p(s) + N_{sp}(s)N_{cy}(s)N_p^-(s)N_p^+(s)}, \tag{10.47}$$

and the servo-control closed-loop transfer function by

$$M_{yr}(s) = C_{sp}(s)C_r(s)M_{yd}(s). \tag{10.48}$$

The main use of the dead-time compensating controllers is to improve the servo-control response for processes with a long dead-time. Then, a target must be stated for (10.48) to apply the MoReRT methodology to tune a $PI_{SP2}$ controller.

#### $PI_{SP2}$ Servo-Control of a SOPDT Process

Consider the controlled process model given by the SOPDT transfer function

$$P(s) = \frac{K\mathrm{e}^{-Ls}}{(Ts+1)(aTs+1)}, \theta_p = \{K, T, a, L\}. \tag{10.49}$$

Using (10.48) the set-point closed-loop transfer function target for the $PI_{SP2}$ control of the SOPDT model is selected as

$$M_{yr}^t(s) = \frac{(\beta T_i s + 1)\mathrm{e}^{-Ls}}{(\tau_c Ts + 1)^2(a\tau_c Ts + 1)}, \tag{10.50}$$

where $\tau_c$ is the design parameter and $\theta_{PI_{SP2}} = \{K_p, T_i, \beta, K_m, T_m, L_m\}$ the dead-time compensating controller parameters to tune.

To avoid abrupt changes in the controller output when a step change in the set-point is made the proportional set-point weight $\beta \to 0$. Then, the servo-control target (10.50) used for controller tuning is reduced to

$$M_{yr}^t(s) = \frac{\mathrm{e}^{-Ls}}{(\tau_c Ts + 1)^2(a\tau_c Ts + 1)}. \tag{10.51}$$

The design parameter $\tau_c$ (closed-loop poles relative speed respect to the controlled process main time constant) is adjusted to obtain a control system with a target robustness $M_S^t$.

**Table 10.3** Example—$PI_{SP2}$ controller parameters (SOPDT model, $\tau_L = 5$)

| $\tau_c$ | $K_p$ | $T_i$ | $\beta$ | $K_m$ | $T_m$ | $L_m$ | $M_S$ |
|---|---|---|---|---|---|---|---|
| 0.7 | 7.531 | 2.397 | 0 | 0.495 | 2.635 | 10.592 | 2.037 |
| 0.8 | 5.361 | 2.619 | 0 | 0.492 | 2.591 | 11.025 | 1.992 |
| 0.9 | 3.894 | 2.742 | 0 | 0.492 | 2.553 | 11.064 | 1.950 |
| 1.0 | 2.836 | 2.749 | 0 | 0.489 | 2.521 | 11.097 | 1.915 |
| 1.1 | 2.037 | 2.623 | 0 | 0.488 | 2.497 | 11.124 | 1.872 |
| 1.2 | 1.411 | 2.348 | 0 | 0.491 | 2.477 | 11.142 | 1.822 |
| 1.3 | 0.909 | 1.913 | 0 | 0.495 | 2.461 | 11.147 | 1.782 |
| 1.4 | 0.508 | 1.325 | 0 | 0.502 | 2.450 | 11.128 | 1.750 |
| 1.5 | 0.195 | 0.618 | 0 | 0.510 | 2.451 | 11.075 | 1.735 |

### Example—Robust Tuning of a $PI_{SP2}$ Controller

The controlled process model is

$$P(s) = \frac{K e^{-Ls}}{(Ts + 1)(aTs + 1)}, \tag{10.52}$$

with parameters $K = 0.5$, $T = 2$, $a = 0.75$, $L = \{10,\ 20\}$ ($\tau_L = \{5,\ 10\}$).

Optimization results confirm that to obtain a servo-control target response without a zero as (10.51) the set-point weight $\beta \approx 0$.

The $PI_{SP2}$ controller parameters are listed in Table 10.3.

The controller and predictor parameters for both controlled processes ($\tau_L = 5$ and $\tau_L = 10$) are shown in Fig. 10.15.

From this figure, it is noticed that the parameters obtained for the FOPDT model used in the predictor are nearly constant. Their average $\{K_m, T_m, L_m\}$ values are $\{0.495, 2.515, 11.087\}$ for the controlled process with $L = 10$, and $\{0.500, 2.570, 20.966\}$ for the $L = 20$ case.

Using a two-point identification method [5], the parameters of a FOPDT approximation for the controlled process model (10.52) are $\{0.50, 2.764, 10.90\ (20.90)\}$, very similar to the ones obtained for the predictor model from the servo-control responses matching procedure.

The responses of the control $PI_{SP2}$ system to a 10 % set-point step change followed by a 5 % change in the disturbance are shown in Figs. 10.16 and 10.17 for three different robustness levels.

From the optimization results, it is also noticed that although the Smith predictor compensates for the dead-time still are constraints in the highest robustness that can be achieved: $M_S = 1.73$ ($\tau_L = 5$) and $M_S = 1.86$ ($\tau_L = 10$).

The Nyquist plots of the $PI_{SP2}$ control system of the SOPDT controlled processes are shown in Fig. 10.18 that also show the robustness circles corresponding to $M_S = 2$ (red) and $M_S = 1.2$ (green). These confirm the control system robustness measured with the maximum sensitivity.

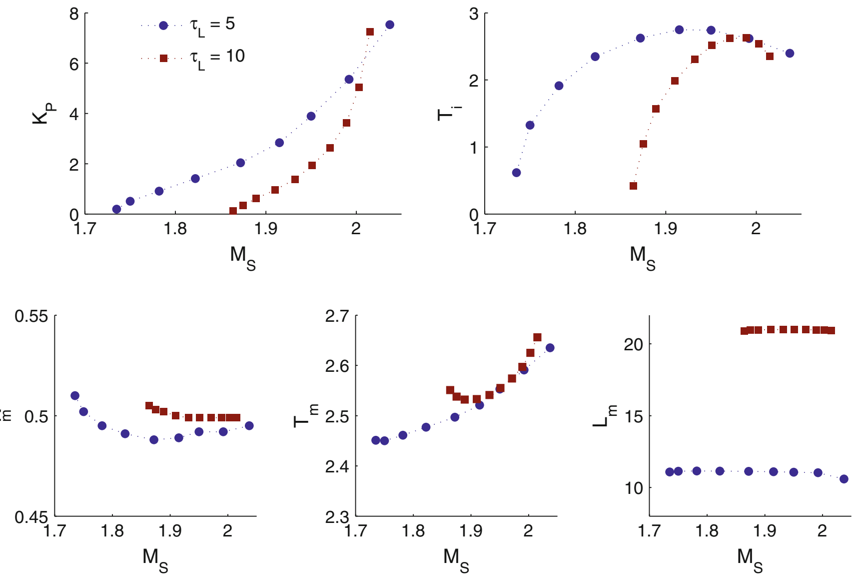



**Fig. 10.15** $PI_{SP2}$ parameters (SOPDT model)

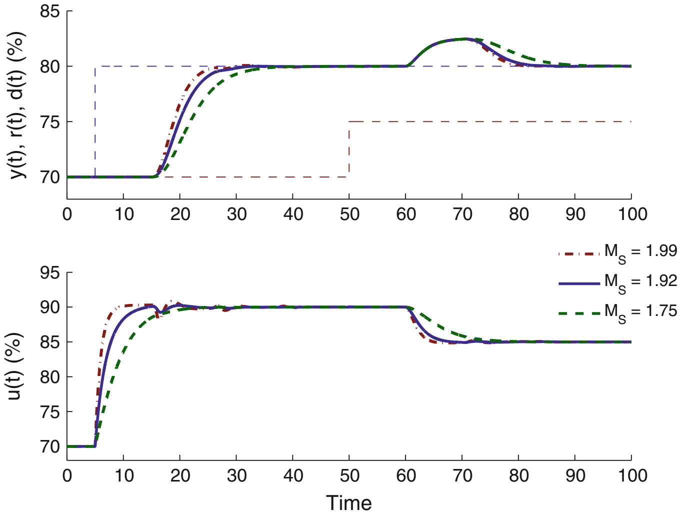



**Fig. 10.16** $PI_{SP2}$ control responses (SOPDT model, $\tau_L = 5$)

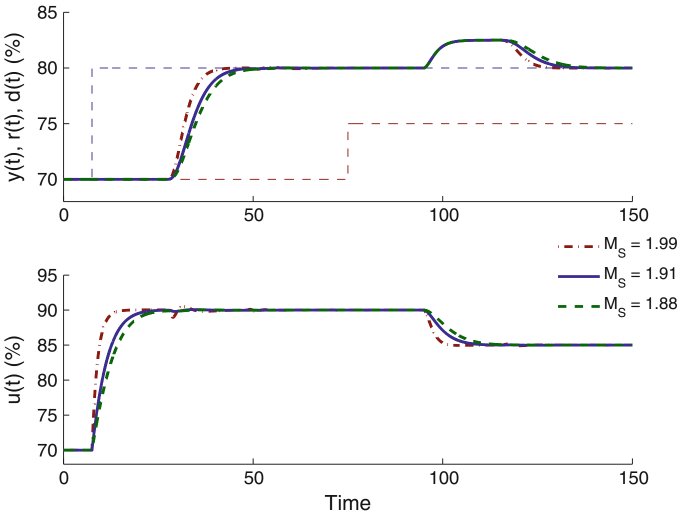



**Fig. 10.17** $PI_{SP2}$ control responses (SOPDT model, $\tau_L = 10$)

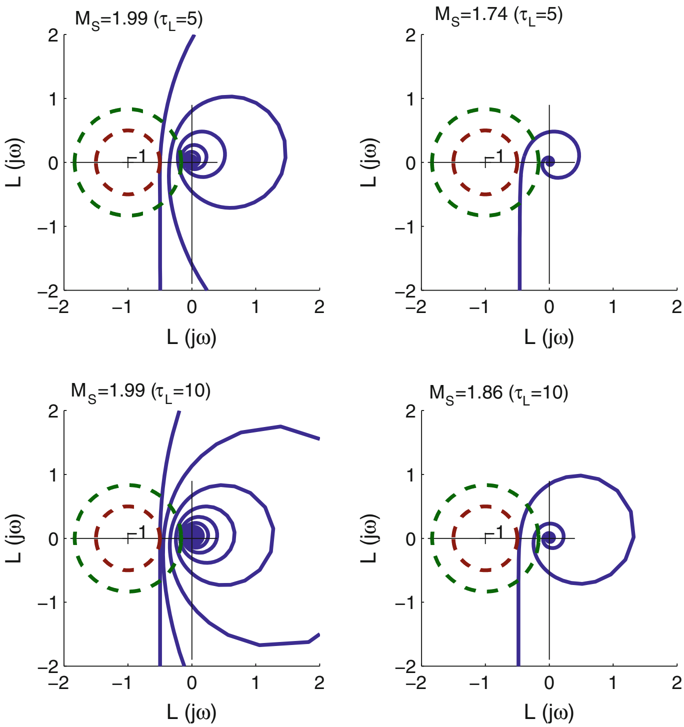



**Fig. 10.18** SOPDT process $PI_{SP2}$ control system Nyquist plots

For control systems with dead-time compensating controllers like the Smith predictor, the maximum sensitivity does not tell us all the history about the control system robustness to changes in the controlled process characteristics. They can turn unstables if the controlled process dead-time increases but also if it decreases.

For the controlled process of the example the worst case corresponds to the $\tau_L = 10$ process and $M_S = 1.99$ $PI_{SP2}$ tuning. The control system turns unstable if the controlled process dead-time change +12.75 % and also if it change –13.75 % respect to the dead-time used in the predictor model as shown in Bode plots of Fig. 10.19.

The servo-control responses with a $\pm 10$ % change in the controlled process dead-time (still stables but oscillatory) and with a $\pm 20$ % change (unstables) are show in Fig. 10.20.

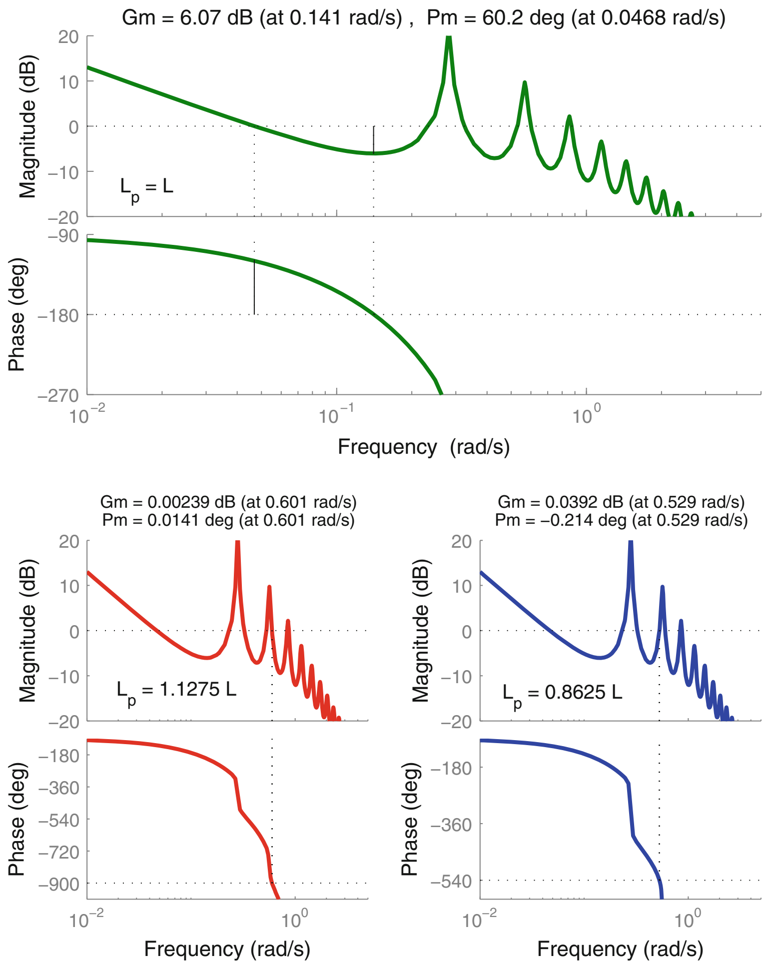



**Fig. 10.19** $PI_{SP2}$ control system Bode plots (SOPDT model, $\tau_L = 10$, $M_S = 1.99$)

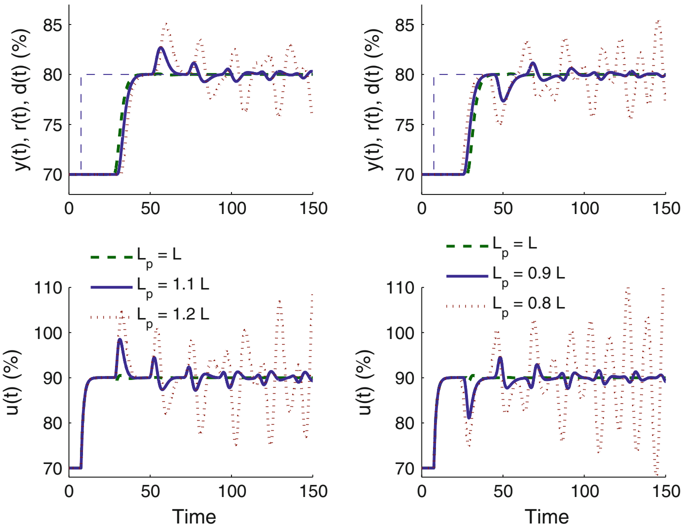



**Fig. 10.20** $PI_{SP2}$ control system responses with ±10 % and ±20 % process dead-time, SOPDT controlled process ($\tau_L = 10$), control system robustness $M_S = 1.99$

Control systems with a Smith predictor are sensible to both model dead-time underestimation and overestimation.

**Chapter Remarks**

Three possible extensions of the MoReRT design methodology for PID controllers are proposed.

The use of control effort response targets and consider that the controlled process disturbance dynamics is not the same of the manipulated variable for tuning PID controllers. Also the use of the MoReRT procedure for robust tuning of dead-time compensating controllers (Smith predictor) is presented. It is shown that the same design procedure can be used with diverse controllers and controlled processes to design robust control systems. In some cases, the restricted structure of the controller used or the dynamic characteristics of the controlled process impose constrains to the maximum robustness level that can be achieved.

## References


1. Shinskey, F.G.: Process control: as taught vs as practiced. Ind. Eng. Chem. Res. **41**, 3745–3750 (2002)
2. van de Vusse, J.G.: Plug-flow type reactor versus tank reactor. Chem. Eng. Sci. **19**(12), 994–996 (1964)
3. Bequette, B.W.: Process Control—Modeling, Design, and Simulation. Pearson Education, Inc., Upper Saddle River (2003)

4. Balaguer, P., Alfaro, V.M., Arrieta, O.: Second order inverse response process identification from transient step response. ISA Trans. **50**, 231–238 (2011)
5. Alfaro, V.M.: Low-order models identification from the process reaction curve. Ciencia y Tecnología (Costa Rica) **24**(2), 197–216 (2006). (in Spanish) http://revista-ciencia-tecnologia.ucr.ac.cr/index.php/ciencia-tecnologia/article/view/35
6. Smith, O.J.M.: Closed control of loops with dead time. Chem. Eng. Prog. **53**, 217–219 (1957)
7. Normey-Rico, J.E., Camacho, E.F.: Control of Dead-time Processes. Springer verleg London Limited, London (2007)

# Chapter 11
# MoReRT Practical Application

We have used the MoReRT design procedure for tuning different two-degree-of-freedom (2DoF) proportional integral derivative (PID) control algorithms in order to control diverse controlled process dynamics. The design methodology considerations were explained and explicit controller tuning relations were obtained that can be used directly knowing the controlled process model and the controller control algorithm.

The appendix presents a MATLAB® software package that helps the application of the proposed methodology. This is provided in order to help the practical application of the MoReRT design procedure. However, in order to go on that direction there is the need for other practical considerations. This is the purpose of this chapter that addresses different aspects that will help for the consideration of the practical implementation of the MoReRT design procedure. The following practical considerations are addressed in this chapter:

- On one side, to be able to go from the theoretical results to a practical application of the proposed design methodology, it is necessary to know the availability of 2DoF PID control algorithms in actual commercial controllers, their equation, tuning parameters, and limitations. This chapter presents a review of the available commercial PID controller functions that allow for a 2DoF implementation. Therefore, providing an overview of the possibilities of direct application of the suggested tunings.
- Considering that the application of the MoReRT design procedure to different controllers and controlled processes has been described along several previous chapters it is also convenient to gather the developed tuning rules into a single section for easy access and use. The overall set of process dynamics and applicability ranges are summarized.
- Even the set of process dynamics and controller structures has been as wide as possible, it is also possibly that tuning rules have not been obtained for a particular controlled process/control algorithm combination. For instance for some normalized controlled process models with two or more parameters. In these cases, a detailed flowchart of the method bearing in mind the applicability of the digital



V.M. Alfaro and R. Vilanova, *Model-Reference Robust Tuning of PID Controllers*,
Advances in Industrial Control, DOI 10.1007/978-3-319-28213-8_11

simulation/optimization procedure is presented. This is presented in order to help end users to develop their own versions of the MoReRT design procedure even with alternative different cost functions.

The practical application is also exemplified by considering a practical case study, a continuous stirred-tank heater (CSTH) where different disturbance profiles are considered, selection of desired target dynamics, etc.

## 11.1 Commercial Two-Degree-of-Freedom PID Controllers

In the following, the two-degree-of-freedom control algorithms implemented in some commercial stand-alone controllers, programmable logic controllers (PLC), distributed control systems (DCS), data acquisition, and/or control software are presented. The main purpose is to provide a listing of available 2DoF PID controllers as well as a practical reference of the parameters each manufacturer defines in their implementation that have their corresponding equivalent on the developments presented here.

### *11.1.1 ABB Control Technologies*

**The PID01 Function**

The PID01 is a functional unit for closed-loop process control available in the ABB Extended Automation System 800xA [1]. This last one integrates a distributed control system (DCS) with an electrical control system, and a safety system for process automation (continuous and batch control).

In the PID01 function [2, 3] the (scaled) control system deviation is obtained with the following equation

$$Dev = (MV - WSP) \cdot \left( \frac{OUT_{max} - OUT_{min}}{MV_{max} - MV_{min}} \right), \tag{11.1}$$

and the PID controller "transfer function" as described in [2] is:

$$G(s) = Gain \left( \beta \cdot WSP - MV + \frac{1}{s \cdot TI} + \frac{s \cdot TD}{1 + s \cdot TF} \right). \tag{11.2}$$

Unfortunately this expression, taken from the functional unit manual, is not really a transfer function. It mixes the transfer functions of the I and D control modes with the signal of the P mode.

The PID01 controller output signal should be given by the following expression

$$OutP(s) = Gain\left(\beta \cdot WSP - MV + \frac{1}{s \cdot TI}Dev + \frac{s \cdot TD}{1 + s \cdot TF}Dev\right), \qquad (11.3)$$

if the derivative mode is performed on the error signal ($Deriv = 1$), and by

$$OutP(s) = Gain\left(\beta \cdot WSP - MV + \frac{1}{s \cdot TI}Dev - \frac{s \cdot TD}{1 + s \cdot TF}MV\right), \qquad (11.4)$$

if the derivative mode is performed on the measured signal ($Deriv = 0$, the default value).

Equivalences of the manufacturer notation with the one use in this work, and the PID01 default parameters values, are:

- $\beta$—Beta Factor (set-point factor), default value 1.0,
- $Deriv$ ($\gamma$)—Selection of derivation, 0 or 1, default value 0 (derivative mode over $MV$ only),
- $Dev$—Deviation (error),
- $Gain$ ($K_p$)—Gain, default value 0.5,
- $MV$ ($y$)—Measured Value,
- $OutP$ ($u$)—Output,
- $TD$ ($T_d$)—Derivative Time, default value 0.0 s,
- $TI$ ($T_i$)—Integration Time, default value 15 s,
- $TF$ ($= \alpha T_d$)—Filter Time, default value 0.0 s,
- $Ts$—Sampling Time,
- $WSP$ ($r$)—Working Setpoint.

In addition, the PID01 controller algorithm performs the following tests:

- If $TI < Ts$, then $TI = Ts$,
- If $TD$ and $TF \leq Ts$, then $TD = TF = 2Ts$.

According to the manufacturer information there is no constraint on the values that the proportional set-point weight factor $\beta$ can take, it can be set lower or higher than one.

Other PID01 functionalities include: bumpless transfer between operation modes, controllable change rate of set-point and output signal, external feedback, and gain scheduling. The PID01 controller *ExtCtrl* (external control) input parameter can be connected to the *GainSched* [3] function block *ExtCtrl* output parameter to set gain scheduling of the PID tuning parameters.

The *GainSched* function block can include up to five zones (five sets of controller parameter $\{Gain, TI, TD, TF\}$) with four interzone limits (controller parameters set switching points). Zone switching is controlled by the *SchedIn* variable. The controller set-point weight $\beta$ remains the same for all zones.

*GainSched* function block default values are:

- *SchedIn*: Scheduling input, default value 0.0,
- *ZLim12*: Zone 1 to 2 limit, default value 20.0,
- *ZLim23*: Zone 2 to 3 limit, default value 40.0,
- *ZLim34*: Zone 3 to 4 limit, default value 60.0,
- *ZLim45*: Zone 4 to 5 limit, default value 80.0.

If the controlled process gain is highly nonlinear or if its dynamic characteristics change significantly over the entire control system operation range, the possibility of use gain scheduling on this controller allows the user to obtain more control system performance reducing the robustness requirements for each operation zone in lieu of use a high robust design and a single controller setting.

The ABB PID01 function controller (with $Deriv = 0$) corresponds to the (2.11) two-degree-of-freedom Standard PID.

**The PID01A Function**

A PID control algorithm with an auto-tuning function (the PID01A [4]) is also available.

Process model identification is relay based and PI or PID controller tuning parameters are given for a "slow," "normal," or "fast" response ("Extra Damped", "Damped," "Normal," "Fast," or "Extra Fast" response according to [5]). The auto-tuning function calculates and suggests new values for the controller $Gain$, $TI$, *TD*, and *TF* parameters. There is no indication of the controlled process model and the tuning relations or design criteria used in the auto-tuning function.

Manufacturer manual [4] shows a one-degree-of-freedom Standard PID control algorithm for the PID01A function with all three control modes applied to the error signal but it also indicates that $\gamma$ $(Deriv)$ can be 0 or 1 and that the controller has an adjustable proportional set-point weight ($\beta$). The PID control algorithm implemented in the PID01A functional unit should be the same two-degree-of-freedom Standard PID control algorithm used in the PID01 function.

### *11.1.2 Emerson Process Management*

**The DeltaV PID Function Block**

The PID function block of Emerson DeltaV control system [6–8] supports standard and series form one-degree-of-freedom PID control algorithms as well as a two-degree-of-freedom PID control algorithm. The PID STRUCTURE parameter is used to select the controller control algorithm.

If the STRUCTURE parameter is set to "Two Degrees of Freedom Controller" the controller output is given by the following relation:

$$OUT(s) = \pm GAIN_a \cdot \left\{ BETA \cdot SP - PV + \frac{1}{T_r\, s}(SP - PV) + \frac{T_d\, s}{\alpha\, T_d\, s + 1}(GAMMA \cdot SP - PV) \right\}. \tag{11.5}$$

Controller can also include a process variable first-order filter ($PV\_FTIME$, s) and a set-point first-order filter ($SP\_FTIME$, s).

Equivalences of the manufacturer notation with the one use in this work, and the default parameters values, are:

- $\pm$—Reverse/Direct acting,
- *ALPHA* ($\alpha$)—Derivative filter constant, $0.05 \leq \alpha \leq 1.0$, default value 0.125,
- *BETA* ($\beta$)—Proportional mode set-point weight, $0 \leq \beta \leq 1$,
- *ERROR*—Error ($SP - PV$),
- *GAIN* ($K_p$)—Gain, default value 0.5,
- $GAIN_a$—Scaled gain,
- *GAMMA* ($\gamma$)—Derivative mode set-point weight, $0 \leq \gamma \leq 1$,
- *OUT* ($u$)—Controller output,
- $PV,\ IN$ ($y$)—Process variable,
- *RATE* ($T_d$)—Derivative action time constant, default value 0.0 s,
- $RESET,\ T_r$ ($T_i$)—Integral action time constant, default value 10 s,
- *SP* (r)—Set-point value.

Other capabilities of the PID block include: set-point rate limits, nonlinear gain, gain scheduling, output tracking, and dynamic reset limiting.

The *PID* function block can be combined with the $PID\_GAINSCHED$ (gain scheduling) module [6, 8] to define three operation ranges (three sets of controller parameters $\{Gain,\ Reset,\ Rate\}$ with two region boundary values). Gain scheduling can be based on PV, OUT, or on an auxiliary variable (AuxVar) signal.

The Emerson DeltaV PID function block (with STRUCTURE = Two Degrees of Freedom Controller) corresponds to the (2.11) 2DoF Standard PID with or without set-point and process controlled variable filters (both of first order).

It restricts the set-point weight $\beta$ values that can be used to the range $0 \leq \beta \leq 1$.

### *11.1.3 Mitsubishi Electric*

#### The S.2PID Instruction

The process control S.2PID instruction [9] of the MELSEC-Q Series PLC compute a velocity type two-degree-of-freedom PID control algorithm using incomplete differentiation.

Manipulated variable increment (at time step $n$) is given by following equation:

$$\Delta MV_n = K_p \cdot \left\{ (1-\alpha_m)\cdot(DV_n - DV_{n-1}) + \frac{CT}{T_I}\cdot DV_n \right. \\ \left. +(1-\beta_m)\cdot B_n + \alpha_m \cdot C_n + \beta_m \cdot D_n \right\}, \qquad (11.6)$$

where

$$B_n = B_{n-1} + \frac{M_D \cdot T_D}{M_D \cdot CT + T_D}\cdot\left\{DV_n - 2DV_{n-1} + DV_{n-2} - \frac{CT\cdot B_{n-1}}{T_D}\right\}, \qquad (11.7)$$

$$C_n = \begin{cases} PV_n - PV_{n-1} & \text{if PN = 1,} \\ -(PV_n - PV_{n-1}) & \text{if PN = 0,} \end{cases} \qquad (11.8)$$

$$D_n = \begin{cases} D_{n-1} + \frac{M_D\cdot T_D}{M_D\cdot CT+T_D}\cdot\left\{PV_n - 2PV_{n-1} + PV_{n-2} - \frac{CT\cdot D_{n-1}}{T_D}\right\} & \text{if PN = 1,} \\ D_{n-1} + \frac{M_D\cdot T_D}{M_D\cdot CT+T_D}\cdot\left\{-(PV_n - 2PV_{n-1} + PV_{n-2}) - \frac{CT\cdot D_{n-1}}{T_D}\right\} & \text{if PN = 0.} \end{cases} \qquad (11.9)$$

Also, the controller performs the following tests:

- If $T_D = 0$, then $B_n = 0$ and $D_n = 0$,
- If $T_I = 0$, then $\frac{CT}{T_I}\cdot DV_n = 0$.

Equivalences of the manufacturer notation with the one use in this work, and the default parameters values, are:

- $\alpha_m$ $(= 1-\beta)$[1]—Proportional two-degree-of-freedom parameter, $0.0 \leq \alpha_m \leq 1.0$, default value 0.0,
- $\beta_m$ $(= 1-\gamma)$[1]—Derivative two-degree-of-freedom parameter, $0.0 \leq \beta_m \leq 1.0$, default value 1.0,
- $\Delta MV$ $(\Delta u)$—Controller output change,
- $CT$—Control cycle, it is an integral multiple of the execution cycle ($\Delta T$), default value 1.0 s,
- $DV$—Deviation value (error), $DV = PV - SV$ for forward operation and $DV = SV - PV$ for reverse operation,
- $K_p$—Gain, default value 1.0,
- $M_D$ $(= 1/\alpha)$—Derivative gain, default value 8.0,
- $PN$—Operation mode, 0: reverse operation, 1: forward operation, default value 0,
- $PV$ $(u)$—Process value,
- $SV$ $(r)$—Setting value (set-point),
- $T_D$ $(T_d)$—Derivative constant, default value 0.0 s,
- $T_I$ $(T_i)$—Integral constant, default value 10.0 s,

[1]A $m$ (Mitsubishi) subindex has been added to the manufacturer $\alpha$ and $\beta$ parameters to avoid a confusion with same Greek letters used in this work but with a different meaning.

In this controller, to decrease the proportional set-point weight factor $\beta$ from its default value ($\alpha_m = 0 \Rightarrow \beta = 1$), the $\alpha_m$ parameter must be increased. The opposite situation occurs with the derivative set-point weight. To change the derivative set-point weight $\gamma$ from its default value ($\beta_m = 1 \Rightarrow \gamma = 0$) the $\beta_m$ parameter must be decreased.

### 11.1.4 National Instruments

**LabVIEW PID Advanced VI**

The linear-gain advanced proportional integral derivative control virtual instrument (VI) of the LabVIEW 2014 Full Development System [10], implements the following equation:

$$CO = K_c \left\{ beta \cdot SP - PV + \frac{1}{T_i \cdot s}(SP - PV) + \left( \frac{T_d \cdot s}{1 + alpha \cdot T_d \cdot s} \right)(gamma \cdot SP - PV) \right\}. \tag{11.10}$$

Equivalences of the manufacturer notation with the one use in this work, and the default parameters values, are:

- $alpha$ ($\alpha$)—Derivative action filter parameter, $0.0 \le alpha \le 1.0$,
- $beta$ ($\beta$)—Setpoint weighting (P), $0.0 \le beta \le 1.0$, default value 1.0,
- $gamma$ ($\gamma$)—Setpoint weighting (D), $0.0 \le gamma \le 1.0$, default value 0.0,
- $CO$ ($u$)—Controller Output,
- $K_c$ ($K_p$)—Proportional Gain, default value 1.0,
- $PV$ ($y$)—Process Variable,
- $T_d$—Derivative Time, default value 0.0 min,
- $T_i$—Integral Time, default value 0.01 min,
- $SP$ ($r$)—Set-point,

Other characteristics of the PID Advanced VI include: nonlinear integral action and error-squared control.

The PID Advanced VI can be combined with the PID Control Input Filter (first-order), PID Gain Schedule, PID Output Rate Limiter, and PID Setpoint Profile VIs. Control algorithm in the LabVIEW PID Advanced VI controller (with $gamma = 0$) corresponds to the (2.11) two-degree-of-freedom Standard PID.

It restricts the set-point weight $\beta$ values that can be used ($0.0 \le \beta \le 1.0$).

**LabVIEW PID Advanced Auto-tuning VI**

The PID Advanced Auto-tuning VI control algorithm is the same two-degree-of-freedom Standard PID of the PID Advanced VI with an auto-tuning function for the controller parameters.

The auto-tuning process includes following parameters:

- Technique:
  - 0 (Step Open Loop): use an open-loop step test to base tuning on a process first-order plus dead-time model,
  - 1 (Step Closed Loop): use ultimate gain (steady-state oscillation response) for tuning,
  - 2 (Relay): use relay feedback test (with hysteresis) to obtain process ultimate information,
  - 3 (PID Relay): keeps the PID controller in the loop with the relay.
- Type of controller:
  - 0 (P)—for retuning only $K_c$;
  - 1 (PI)—for retuning $K_c$ and $T_i$,
  - 2 (PID)—for retuning $K_c$, $T_i$, and $T_d$.
- Control specification—0 (“normal”), 1 (“fast”), or 2 (“slow”, default) response.

### *11.1.5 OMRON*

**Basic PID Block Model <011>**

The two-degree-of-freedom PID control algorithm used in the Basic PID block 011 of OMROM PLCs is not explicitly shown in manufacturer manual [11]. From general description and parameters tables it can be supposed that it corresponds to a Standard 2DoF PID with a nonadjustable derivative filter.

Equivalences of the manufacturer notation with the one use in this work, and the default parameters values, are:

- $alpha_o$ ($\beta$)[2]—PID 2 degrees of freedom parameter (proportional control mode), $0.0 \leq alpha_o \leq 1.0$, default value 0.65,
- $beta_o$ ($\gamma$)—PID 2 degrees of freedom parameter (derivative control mode), $0.0 \leq beta_o \leq 1.0$, default value 1.0,
- $D$ ($T_d$)—Differential time (0–9999 s), default value 0.0 s,
- $I$ ($T_i$)—Integral time (0, 1–9999 s), default value 0 (No integral action),
- $MV$ ($u$)—Manipulated variable,
- $P$ ($= 100/K_p$)—Proportional band (0.1–999.0 %), default value 100.0 %,
- $PV$ ($y$)—Process variable,

[2]An $o$ (OMRON) sub index has been added to the manufacturer $alpha$ and $beta$ parameters to avoid a confusion with equivalent Greek letters used in this work but with a different meaning.

### 11.1.6 REX Controls

#### The PIDU Function Block

The PIDU is one of the two-degree-of-freedom control function blocks available in the REX Control System [12].

The control algorithm implemented on the PIDU function block is [13]:

$$U(s) = \pm K \left\{ bW(s) - Y(s) + \frac{1}{T_i s}[W(s) - Y(s)] + \frac{T_d s}{\frac{T_d}{N}s + 1}[cW(s) - Y(s)] \right\} + Z(s). \tag{11.11}$$

- $b$ ($\beta$)—Setpoint weight, proportional part, default value 1.0,
- $c$ ($\gamma$)—Setpoint weight, derivative part, default value 0.0,
- $K$ ($K_p$)—Controller gain, default value 1.0,
- $N$ ($= 1/\alpha$)—Derivative filter parameter, default values 10.0,
- $T_d$—Derivative time, default value 1.0 s,
- $T_i$—Integral time, default value 4.0 s,
- $U$ ($u$)—Manipulated variable `mv`,
- $Y$ ($y$)—Process variable `pv`,
- $W$ ($r$)—Setpoint variable `sp`,
- $Z$—Feedforward control variable `dv`.

Other 2DoF PID control blocks are: PIDAT (PID controller with relay autotuner), PIDGS (PID controller with gain scheduling, up to six parameters sets), PIDMA (PID controller with moment autotuner), and PIDUI (PID controller with variable parameters).

Control algorithm in the REX PIDU control function block (with $c = 0$) corresponds to the (2.11) two-degree-of-freedom Standard PID.

### 11.1.7 Siemens AG

#### The PID_Compact Function

The PID_Compact function of the SIMATIC S7 products and particularly of the S7-1200 programmable logic controller [14, 15] use following equation to calculate the controller output value:

$$y = K_p \left[ (b \cdot w - x) + \frac{1}{T_I \cdot s}(w - x) + \frac{T_D \cdot s}{a \cdot T_D \cdot s + 1}(c \cdot w - x) \right]. \tag{11.12}$$

Equivalences of the manufacturer notation with the one use in this work, and the default parameters values, are:

- $a$ ($\alpha$)—Derivative delay coefficient, default value 0.0,
- $b$ ($\beta$)—Proportional action weighting, default value 0.0,
- $c$ ($\gamma$)—Derivative action weighting, default value 0.0,
- $K_p$—Proportional gain, default value 1.0,
- $T_D$ ($T_d$)—Derivative action time, default value 0.0 s,
- $T_I$ ($T_i$)—Integral action time, default value 20.0 s,
- $w$ ($r$)—Set-point value,
- $x$ ($y$)—Process value,
- $y$ ($u$)—Output value.

There is no restriction on the $a$, $b$, and $c$ parameters ($0 \leq$ Real 32-bit values). Other characteristics of the PID_Compact include pre-tuning on start-up, gain scheduling, and PID-Tuner.

The GainSched block of the PCS 7 Advanced Process Library (APC) [16, 17] allows to store PID parameters ($K_p$, $T_I$, $T_D$) for up to three operating points. Current operating point is represented by a continuous variable input (controlled variable or other). Controller parameters between two control system operation points are obtained by a linear interpolation of the corresponding parameters stored for the two nearest operation points allowing a continuous adaptation of controller parameters when system goes from one operation point to another. If the current operating point is below the lowest operating point or above the highest operation point in the table the controller parameters for the corresponding boundary point are used. The change of the controller parameters is controlled by the dead band and the control zone parameters.

This gain scheduling scheme is different of the schemes (by operating ranges) usually found in other controllers.

The Siemens PID_ Compact function (with $c = 0$) corresponds to the (2.11) two-degree-of-freedom Standard PID. Other 2DoF PID control algorithm (incremental form) is available in the PID_3Step instruction designed for use with motor actuated devices (MOVs).

## 11.1.8 Toshiba Corporation

### The PID_P Function

The PID_P (current output) function control algorithm of the Unified Controller nv series and the Integrated Controller V series is [18]:

$$\begin{aligned} MV = K_p \left(1 + \frac{1}{T_i \cdot s}\right) \Bigg\{ \alpha_t \cdot SV - PV + \left(\frac{Td \cdot s}{1 + \eta \cdot Td \cdot s}\right) (\alpha_t \cdot \gamma_t \cdot SV - PV) \\ + \left(\frac{1}{1 + \beta_t \cdot T_i \cdot s}\right) \left[ (1 - \alpha_t) \cdot SV - \left(\frac{Td \cdot s}{1 + \eta \cdot T_d \cdot s}\right) (\alpha_t \cdot \gamma_t \cdot SV - PV) \right] \Bigg\}. \end{aligned} \tag{11.13}$$

It can be rewritten as

$$MV = \beta_t \cdot K_p \left(\frac{1 + T_i \cdot s}{1 + \beta_t \cdot T_i \cdot s}\right) \left\{\alpha_t \cdot SV - PV + \frac{1}{\beta_t \cdot T_i \cdot s}(SV - PV) + \left(\frac{T_d \cdot s}{1 + \eta \cdot T_d \cdot s}\right)(\alpha_t \cdot \gamma_t \cdot SV - PV)\right\}. \quad (11.14)$$

Equivalences of the manufacturer notation with the one use in this work are:

- $\alpha_t$ $(\beta)$[3]—P term two-degrees-of freedom coefficient,
- $\beta_t^2$—I term two degrees of freedom coefficient,
- $\eta$ $(\alpha)$—Rate gain,
- $\gamma_t$ $(= \gamma/\beta)$—D term two-degrees-of-freedom coefficient,
- $K_p$—Proportional gain,
- $PV$ $(y)$—Process value,
- $MV$ $(u)$—Output,
- $SV$ $(r)$—Setting value,
- $T_d$—Rate time, minutes,
- $T_i$—Reset time, minutes.

Coefficients $\alpha_t$, $\beta_t$, $\eta$, and $\gamma_t$ are all positive Real numbers (32-bits).

It is noted in (11.14) that the two-degrees-of-freedom coefficient of the integral control term ($\beta_t$) does not multiply the set-point in the integrand. It can be used to reduce the integral control term relative gain. Integration is performed over the error signal (SV-PV) in order to assure zero steady-state error.

If $\alpha_t = \beta_t = \gamma_t = 1$ (11.14) reduces to the one-degree-of-freedom Standard PID algorithm with all models applied over the error signal

$$MV = K_p \left\{1 + \frac{1}{T_i \cdot s} + \frac{T_d \cdot s}{1 + \eta \cdot T_d \cdot s}\right\}(SV - PV). \quad (11.15)$$

It is also available an incremental version of the (11.14) two-degree-of-freedom PID algorithm in the PIDP_P (pulse output) function.

**The LC531/532 Single-Loop Controllers**

The PID function implemented in the LC531 (current output) and LC532 (pulse output) single-loop controllers [19] is the same two-degree-of-freedom Unified controller nv series PID function described in previous paragraph.

LC531/532 parameters values ranges and default values are [19]:

- 2DoF proportional factor, $0 \le \alpha_t \le 1$, default value 0.4,
- 2DoF integral factor, $1 \le \beta_t < 2$, default value 1.35,
- 2DoF derivative factor, $0 \le \gamma_t < 2$, default value 1.25,
- Rate coefficient, $0 < \eta \le 1$, default value 0.1,

[3] A $t$ (Toshiba) subindex has been added to the manufacturer $\alpha$ and $\beta$ parameters to avoid a confusion with same notation used in this work that have a different meaning.

- Proportional gain $K_p$, default value 0.8,
- Reset time $T_i$, default value 0.1 min,
- Rate time $T_d$, default value 0.0 min.

If $\beta_t = 1$ and $\gamma_t = 0$ (11.14) reduces to the (2.11) 2DoF Standard PID

$$MV = K_p \left\{ \alpha_t \cdot SV - PV + \frac{1}{T_i \cdot s}(SV - PV) - \left( \frac{T_d \cdot s}{1 + \eta \cdot T_d \cdot s} \right) PV \right\}. \tag{11.16}$$

The control algorithm implemented in Toshiba nv series controllers is a two-degree-of-freedom PID with three weighting factors (proportional, integral, and derivative) that made it more flexible than the other commercially available 2DoF PID control algorithms.

### *11.1.9 Main Characteristics and Limitations*

As seen, one common factor of the commercial 2DoF PID controllers is that they implement directly or with some variations, particularly Toshiba, the 2DoF Standard PID control algorithm. Normally, the control algorithm can be aggregated with set-point and controlled variable signal input filters.

It is also noted that although several manufactures do not impose any restriction on the set-point weighting factors values, there are others that limited their settings to be in the range from zero to one.

Restrict the derivative set-point weighting $\gamma$ to be one or lower normally does not implies any constraint for the control. As has been suggested, to avoid a controller output abrupt change on a set-point step change it is necessary to set $\gamma = 0$.

On the other hand, as has been show in [20] and on the examples in previous chapters, if the controlled process characteristics impose a high robust design (low $M_S^t$ values) the resulting proportional set-point weight $\beta$ values would be higher than one. In these cases, the use of $\beta = 1$ will reduce the achievable servo-control performance.

Controllers with $\beta \leq 1$ would constraint the performance to a set-point change of a control system tuned using a high robustness design criteria.

## 11.2 MoReRT Controllers Design Implementation

In the following, first the MoReRT *tuning rules* are summarized. Their scope and applicability are also indicated in terms of the parameters of the considered process dynamics transfer function models. It is worth to notice that this aspect is sometimes neglected in most of the existing tuning rules. After that, and for the application of the MoReRT *design methodology* in those cases were an explicit tuning rule has not been obtained, its steps are detailed in order to help user own implementation of the required optimization steps. This will allow end users to develop tuning rules along the MoReRT approach for some specific an concrete cases not covered in the previous chapters.

### *11.2.1 MoReRT Tuning Rules*

MoReRT tuning rules for direct tuning of 2DoF PI and PID controllers are already derived for overdamped, inverse response, integrating and unstable processes. Corresponding controller dimensionless parameters are presented in Sect. 5.3.

In the following for each controlled process model, control algorithms and robustness levels considered are indicated. Tuning relations and their parameters are referred to the corresponding equations and tables presented in previous chapters.

**Over-Damped Processes**

Controlled process model:

$$P(s) = \frac{K\mathrm{e}^{-Ls}}{(Ts+1)(aTs+1)},\ \theta_p = \{K,\ T,\ a,\ L\},$$
$$0.1 \leq \tau_L = L/T \leq 2.0.$$

1. $a \in \{0.0,\ 0.10,\ 0.25,\ 0.50,\ 0.75,\ 1.0\}$ (FOPDT, SOPDT, DPPDT).

  - Controller control algorithm and parameters:
    2DoF PI ($PI_2$) (2.12), $\theta_c = \left\{K_p,\ T_i,\ T_d = 0,\ \beta\right\}$.
  - Robustness Levels:
    $M_S^t \in \{1.4,\ 1.6,\ 1.8,\ 2.0\}$.
  - Tuning rule:
    Equations (6.8)–(6.10) and Tables 6.1–6.6.
  - Parameters interpolation:
    If controlled process model parameter $a \notin \{0.0,\ 0.10,\ 0.25,\ 0.50,\ 0.75,\ 1.0\}$ and $a_1 \leq a \leq a_2$, the required $\theta_c$ controller parameters must be obtained using a linear interpolation:

$$\theta_c = \left(\frac{a - a_1}{a_2 - a_1}\right)(\theta_{c2} - \theta_{c1}) + \theta_{c1},$$

    where $\theta_{c1}$ and $\theta_{c2}$ are the controller parameters obtained using $a_1$ and $a_2$, respectively.

2. Models with $a = 0$ (FOPDT).

   - Controller control algorithm and parameters:
     2DoF Ideal PID with Filter ($PID_{2F}$) (2.21),
     $\theta_c^* = \left\{K_p^*,\ T_i^*,\ T_d^*,\ T_f,\ \beta^*,\ \gamma^* = 0\right\}$.
   - Robustness levels:
     $M_S^t \in \{1.4,\ 1.6,\ 1.8,\ 2.0\}$.
   - Tuning rule:
     Equations (6.16)–(6.21) and Table 6.15.

3. Models with $a = 0$ (FOPDT).

   - Controller control algorithm and parameters:
     2DoF Ideal Parallel PID with Two Input Filters ($PID_{2IF}$) (6.24), (6.25), and (6.27), $\theta_c = \left\{K_p,\ K_i,\ K_d,\ T_f,\ T_r,\ \sigma,\ \gamma = 0\right\}$.
   - Robustness level:
     $M_S^t = 1.6$.
   - Tuning rule:
     Equations (6.34), and (6.39) or (6.40).

**Inverse Response Processes**

Controlled process model:

$$P(s) = \frac{K(-bTs + 1)}{(Ts + 1)(aTs + 1)},\ \theta_p = \{K,\ T,\ a,\ b\}\,.$$

1. Models with $a \in \{0.10,\ 0.25,\ 0.50,\ 0.75,\ 1.0\}$ and $b$ in ranges listed in Table 7.6 (SOPRHPZ).

   - Controller control algorithm and parameters:
     2DoF PI ($PI_2$) (2.12), $\theta_c = \left\{K_p,\ T_i,\ T_d = 0,\ \beta\right\}$.
   - Robustness Levels:
     $M_S^t \in \{1.4,\ 1.6,\ 1.8,\ 2.0\}$.
   - Tuning rule:
     Equations (7.3)–(7.5) and Tables 7.1–7.5.
   - Parameters interpolation:
     If controlled process model parameter $a \notin \{0.10,\ 0.25,\ 0.50,\ 0.75,\ 1.0\}$ the required controller parameters must be obtained using a linear interpolation as explained above.

2. Models with $0.1 \le a \le 1.0$, and $0.1 \le b \le 2.6$ for $M_S^t = 2.0$ or $0.1 \le b \le 1.15$ for $M_S^t = 1.6$.

   - Controller control algorithm and parameters:
     2DoF Ideal PID with Filter ($PID_{2F}$) (2.21),
     $\theta_c^* = \left\{K_p^*,\ T_i^*,\ T_d^*,\ T_f,\ \beta^*,\ \gamma^* = 0\right\}$.

- Robustness Levels:
  $M_S^t \in \{1.6,\ 2.0\}$.
- Tuning rule:
  Equation (7.12) and Table 7.9 for $M_S^t = 2.0$,
  Equation (7.13) and Table 7.10 for $M_S^t = 1.6$.

**Integrating Processes**

Controlled process model:

$$P(s) = \frac{K\mathrm{e}^{-Ls}}{s(Ts+1)}.$$

1. Models with $T \neq 0$ (ISOPDT)

   - Controller control algorithm and parameters:
     2DoF PI ($PI_2$) (2.12), $\theta_c = \left\{K_p,\ T_i,\ T_d = 0,\ \beta\right\}$.
   - Robustness Levels:
     $M_S^t \in \{1.4,\ 1.6,\ 1.8,\ 2.0\}$.
   - Tuning rule:
     Equations (8.3)–(8.5) and Table 8.1 for an over-damped target response.
     Equations (8.7)–(8.9) and Table 8.2 for an under-damped ($\zeta = 0.80$) target response.

2. Models with $T \neq 0$ (ISOPDT)

   - Controller control algorithm and parameters:
     2DoF Parallel PI with Two Input Filters ($PI_{2IF}$) (6.24)–(6.26),
     $\theta_c = \left\{K_p,\ K_i,\ K_d = 0,\ T_f,\ T_r,\ \sigma\right\}$.
   - Robustness Levels:
     $M_S^t \in \{1.6,\ 2.0\}$.
   - Tuning rule:
     Equations (8.10)–(8.14) for $M_S^t = 2.0$,
     Equations (8.15)–(8.19) for $M_S^t = 1.6$.

3. Models with $T = 0$ (IPDT)

   - Controller control algorithm and parameters:
     2DoF PI ($PI_2$) (2.12), $\theta_c = \left\{K_p,\ T_i,\ T_d = 0,\ \beta\right\}$.
   - Robustness Levels:
     $M_S^t \in \{1.4,\ 1.6,\ 1.8,\ 2.0\}$.
   - Tuning rule:
     Equations (8.22)–(8.24), and Table 8.3 for an overdamped target response or Table 8.4 for an underdamped ($\zeta = 0.8$) target response.

4. Models with $T = 0$ (IPDT)

   - Controller control algorithm and parameters:
     2DoF Ideal Parallel PI and PID with Two Input Filters ($PI_{2IF}$, $PID_{2IF}$) (6.24)–(6.27), $\theta_c = \left\{K_p,\ K_i,\ K_d,\ T_f,\ T_r,\ \sigma,\ \gamma = 0\right\}$.
   - Robustness Levels:
     $M_S^t \in \{1.6,\ 2.0\}$.
   - Tuning rule:
     Equations (8.27)–(8.32) and Table 8.5.

**Unstable Process**

Controlled process model:

$$P(s) = \frac{K\mathrm{e}^{-Ls}}{Ts - 1},\ \tau_L = \frac{L}{T}.$$

1. Models with $0.1 \leq \tau_L \leq 0.55$ (UFOPDT).

   - Controller control algorithm and parameters:
     2DoF PI ($PI_2$) (2.12), $\theta_c = \left\{K_p,\ T_i,\ T_d = 0,\ \beta\right\}$.
   - Robustness Levels:
     $M_S^t \in \{2.0,\ 3.0,\ 4.0,\ 5.0,\ 6.0\}$.
   - Tuning rule:
     Equations (9.3)–(9.5) and Tables 9.1 and 9.2.

2. Models with $0.1 \leq \tau_L \leq 0.85$ (UFOPDT).

   - Controller control algorithm and parameters:
     2DoF Ideal PID with Filter ($PID_{2F}$) (2.21),
     $\theta_c^* = \left\{K_p^*,\ T_i^*,\ T_d^*,\ T_f,\ \beta^*,\ \gamma^* = 0\right\}$.
   - Robustness Levels:
     Maximum robustness allowed estimated with (9.14).
   - Tuning rule:
     Equations (9.8)–(9.13).

3. Models with $0.25 \leq \tau_L \leq 0.85$ (UFOPDT).

   - Controller control algorithm and parameters:
     2DoF Standard PID ($PID_2$) (2.12), $\theta_c = \left\{K_p,\ T_i,\ T_d,\ \beta\right\}$.
   - Robustness Levels:
     Maximum robustness allowed.
   - Tuning rule:
     Equations (9.15)–(9.20).

4. Models with $0.1 \leq \tau_L \leq 1.0$ (UFOPDT).
   - Controller control algorithm and parameters:
     2DoF Ideal Parallel PID with Two Input Filters ($PID_{2IF}$) (6.24), (6.25), and (6.27), $\theta_c = \left\{K_p,\ K_i,\ K_d,\ T_f,\ T_r,\ \sigma,\ \gamma = 0\right\}$.
   - Robustness Levels:
     Maximum robustness allowed estimated with (9.28).
   - Tuning rule:
     Equations (9.21)–(9.27).

### 11.2.2 MoReRT Controllers Design Procedure Outline

Controller design has considered overdamped, inverse response, integral, and unstable controlled processes. All these dynamics are particular cases of the following general controlled process model:

$$P_g(s) = \frac{K(-bTs+1)\mathrm{e}^{-Ls}}{s^m(Ts \pm 1)(aTs+1)}, \quad \theta_{pg} = \{K,\ T,\ a,\ b,\ L,\ m\}. \tag{11.17}$$

Without a loss of generality, and taking into account that industrial controlled process dynamics usually are overdamped, a stable second-order overdamped controlled model given by the transfer function:

$$P(s) = \frac{K\mathrm{e}^{-Ls}}{(Ts+1)(aTs+1)}, \quad \theta_p = \{K,\ T,\ a,\ L\}. \tag{11.18}$$

will be used to outline the design procedure.

Regarding the controller structure, several implementations of the 2DoF proportional integral derivative control algorithms were considered: Standard, Ideal with filter, and Parallel with two input filters. As shown in Sect. 11.1, the more common 2DoF PID control algorithm implemented in commercial controllers is the Standard form, whose output equation is

$$u(s) = K_p\left\{\beta r(s) - y(s) + \frac{1}{T_i s}\left[r(s) - y(s)\right] + \left(\frac{T_d s}{\alpha T_d s + 1}\right)\left[\gamma r(s) - y(s)\right]\right\}. \tag{11.19}$$

$$\theta_c = \{K_p,\ T_i,\ T_d,\ \alpha,\ \beta,\ \gamma\},$$

Then, (11.18) together with (11.19) will became the particular controlled process model/controller control algorithm combination used to exemplify the design procedure.

On that basis, the detailed steps to follow in order to apply the MoReRT design methodology are exposed in what follows as well as an optimization flowchart:

1. Controlled process information

   a. Model transfer function
   Parameters $\theta_p = \{K, T, a, L\}$ of model (11.18) can be identified using an open or closed-loop test performed at the control system normal operating point.
   The model must be a trustworthy representation of the controlled process dynamics over the entire control system operating range.

   b. Robustness requirements
   Variability of the model parameters due to changes on the control system operating point (due to changes on the set-point and/or the disturbances) must be analyzed to set the control system minimum robustness allowed for the control system design.

   c. Normalized controlled process model
   Model (11.18) is normalized using $\hat{s} = Ts$ as follows:

$$\hat{P}(\hat{s}) = \frac{e^{-\tau_L \hat{s}}}{(\hat{s}+1)(a\hat{s}+1)}, \quad \hat{\theta}_p = \{a,\ \tau_L\}. \tag{11.20}$$

2. Control algorithm

   a. Normalized controller output equation
   Using the same transformation as above the control algorithm output (11.19) is normalized as follows:

$$u(\hat{s}) = \kappa_p \left\{ \beta r(\hat{s}) - y(\hat{s}) + \frac{1}{\tau_i \hat{s}} \left[ r(\hat{s}) - y(\hat{s}) \right] - \left( \frac{\tau_d \hat{s}}{\alpha \tau_d \hat{s} + 1} \right) y(\hat{s}) \right\}, \tag{11.21}$$

$$\hat{\theta}_c = \{\kappa_p,\ \tau_i,\ \tau_d,\ \alpha,\ \beta,\ \gamma = 0\},$$

   that can be rewriten as

$$u(\hat{s}) = \hat{C}_r(\hat{\theta}_{cr}, \hat{s}) r(\hat{s}) + \hat{C}_y(\hat{\theta}_{cy}, \hat{s}) y(\hat{s}). \tag{11.22}$$

   Controller normalized parameters $\hat{\theta}_c = \hat{\theta}_{cr} \bigcup \hat{\theta}_{cy}$, where $\hat{\theta}_{cr} = \{\kappa_p, \tau_i, \beta\}$ and $\hat{\theta}_{cy} = \{\kappa_p, \tau_i, \tau_d, \alpha\}$.

   b. Unknown controller parameters
   For the design of a $PI_2$ controller the unknown normalized parameters are $\hat{\theta}_{cPI} = \{\kappa_p, \tau_i, \beta\}$.
   In the $PID_2$ case, the derivative filter constant $\alpha$ value can be obtained from the controller Users' Manual. If this information is not available, $\alpha = 0.10$ can be used without any significant effect over the resulting controller setting. Derivative filter parameter $\alpha$ usual values are in the range from 0.05 to 0.20.

Then for design of a $PID_2$ controller the unknown normalized parameters are $\hat{\theta}_{cPID} = \{\kappa_p, \tau_i, \tau_d, \beta\}$.

3. Desired closed-loop control target responses

   a. Normalized control system output
   Controller design will consider, at the same time, the disturbance, and the set-point control systems responses in order to obtain all the controller parameters at once, given by the expression

$$y(\hat{\theta}_c, \hat{s}) = y_r(\hat{\theta}_c, \hat{s})r(\hat{s}) + y_d(\hat{\theta}_{cy}, \hat{s})d(\hat{s}), \tag{11.23}$$

$$y(\hat{\theta}_c, \hat{s}) = \frac{\hat{C}_r(\hat{\theta}_{cr}, \hat{s})\hat{P}(\hat{s})}{1 + \hat{C}_y(\hat{\theta}_{cy}, \hat{s})\hat{P}(\hat{s})} r(\hat{s}) + \frac{\hat{P}(s)}{1 + \hat{C}_y(\hat{\theta}_{cy}, \hat{s})\hat{P}(\hat{s})} d(\hat{s}). \tag{11.24}$$

   b. Closed-loop reference models
   The target responses are selected considering the controlled process model order and the controller characteristics.
   In this case, the closed-loop target transfer functions are stated as:

$$\hat{M}^t_{yr}(\theta_d, \hat{s}) = \frac{(\tau_c\hat{s} + 1)\mathrm{e}^{-\tau_L\hat{s}}}{(\tau_c^2\hat{s}^2 + 2\zeta\tau_c\hat{s} + 1)(a\tau_c\hat{s} + 1)}, \tag{11.25}$$

$$\hat{M}^t_{yd}(\theta_d, \hat{s}) = \frac{(\tau_i/\kappa_p)\hat{s}\mathrm{e}^{-\tau_L\hat{s}}}{(\tau_c^2\hat{s}^2 + 2\zeta\tau_c\hat{s} + 1)(a\tau_c\hat{s} + 1)}. \tag{11.26}$$

   c. Design parameters
   The design parameters included into the target responses are:

$$\theta_d = \{\zeta, \tau_c\}. \tag{11.27}$$

   If non-oscillatory responses are desired $\zeta = 1$ is used in the target models. As reported, it is possible to obtain more performance from the control system, under the integrated absolute error criteria, without a significant deterioration on the controller effort total variation if small oscillations are allowed in the control systems responses. In this case, it is recommended to use $\zeta = 0.80$ for the $PI_2$ design and $\zeta = 0.70$ for the corresponding for the $PID_2$.
   With the selection of $\zeta$, there is only one design parameter, the closed-loop relative speed $\tau_c$. The closed-loop control system main time constant $T_c = \tau_c T$.

   d. Control system output target
   The total target response, the set-point change response plus the disturbance response, is a function of $\tau_c$:

$$y^t(\tau_c, \hat{s}) = \hat{M}^t_{yr}(\tau_c, \hat{s})r(\hat{s}) + \hat{M}^t_{yd}(\tau_c, \hat{s})d(\hat{s}). \tag{11.28}$$

4. Optimization procedure
   a. Cost functionals
   To obtain the controller parameters that provide the better match, in the least-squares sense, of the control system response with the target one the total cost functional for the optimization is expressed as:

$$J_T(\cdot) \doteq J_r(\hat{\theta}_p, \hat{\theta}_c, \tau_c) + J_d(\hat{\theta}_p, \hat{\theta}_{cy}, \tau_c) \tag{11.29}$$

   where

$$J_r(\cdot) \doteq \int_0^\infty \left[ y_r^t(\hat{\theta}_p, \hat{\theta}_c, \tau_c, \tau) - y_r(\hat{\theta}_p, \hat{\theta}_c, \tau) \right]^2 \mathrm{d}\tau, \tag{11.30}$$

   is the servo-control response matching cost functional, and

$$J_d(\cdot) \doteq \int_0^\infty \left[ y_d^t(\hat{\theta}_p, \hat{\theta}_{cy}, \tau_c, \tau) - y_d(\hat{\theta}_p, \hat{\theta}_{cy}, \tau) \right]^2 \mathrm{d}\tau, \tag{11.31}$$

   the corresponding for the regulatory control response matching.
   b. Initial controller parameters estimation
   The initial controller parameters $\hat{\theta}_c^0$ (starting point) can be estimated using an appropriate PI/PID robust tuning rule or simply selected as $\kappa_p^0 = 1.0$, $\tau_i^0 = 1.0$, $\tau_d^0 = 0.25$, and $\beta^0 = 1.0$.
   c. Initial design parameter
   The design parameter $\tau_c$ is directly related with the resulting control system robustness. The start $\tau_c$ value ($\tau_c^0$) can be obtained using and analytically deducted tuning rule that includes a criteria for its selection based in the target robustness.
   Such type of rules can be found in [21] for FOPDT model/$PI_2$ controller and in [22] for SOPDT model/$PID_2$ controller combinations. For inverse response processes the rule in [23] provides an estimation of the design parameter.
   In absence of a initial $\tau_c$ estimation $\tau_c^0 = 1$ can be used for a first test run. Based on the robustness of the resulting control system, the $\tau_c$ can be increased or decreased to obtain a control system with the desired target robustness level.

5. Control system time response (simulation)
   The required total time span for the control system simulation (for the servo and regulatory control responses) must be checked.
6. Controller tuning
   If the normalized controller parameters obtained from the design procedure are $\theta_c^d = \{\kappa_c^d, \tau_i^d, \tau_d^d, \beta^d\}$ the required parameters for the controller are

$$K_p = \frac{\kappa_p^d}{K},\ T_i = \tau_i^d T,\ T_d = \tau_d^d T,\ \beta = \beta^d,\ \gamma = 0. \tag{11.32}$$

In case that the controller algorithm of the PID controller to tune be other than the Standard used for the design, conversion relations in Sect. 2.3 can be used to obtain the required parameters.

A more direct approach is using on the reference models and control system simulation the PID control algorithm corresponding to the controller to tune.

The work flow of a computer program for the above MoReRT design procedure is summarized in Fig. 11.1. In this it is supposed that the selected start design parameter $\tau_c^0$ will result in a control control system with a robustness lower that the target one ($M_S^0 > M_S^t$). This outlined procedure can be refined including some design parameter $\tau_c$ range and increment/decrement ($\pm\Delta\tau_c$) control.

## 11.3 Case Study

A practical case study is presented in this section in order to exemplify the application of the MoReRT design. A simple process frequently found in industrial applications is used to describe the steps required for tuning the required controllers.

### *11.3.1 Control of a Continuous Stirred-Tank Heater*

Consider a perfectly mixed continuous stirred-tank heater (CSTH) whose control system is depicted in Fig. 11.2. The fluid in the tank is an aqueous solution that needs to be heated. It is of interest to control the level and temperature of the fluid in the tank. To do this the tank outlet flow rate and the heating fluid flow rate are manipulated.

**System Modeling** [24–27]

- Main assumptions:
  - The fluid in the tank is perfectly mixed, the temperature of the outflow is equal to the tank fluid temperature.
  - Fluids densities and heat capacities are assumed to be constant (their temperature dependance is negligible).
  - The tank inlet fluid flow rate and temperature can change.
  - The tank is perfectly insulated (there are no heat losses to the environment).
  - For system modeling the instrumentation signals (pneumatic and electric) are considered changing in the range for 0–100 %.

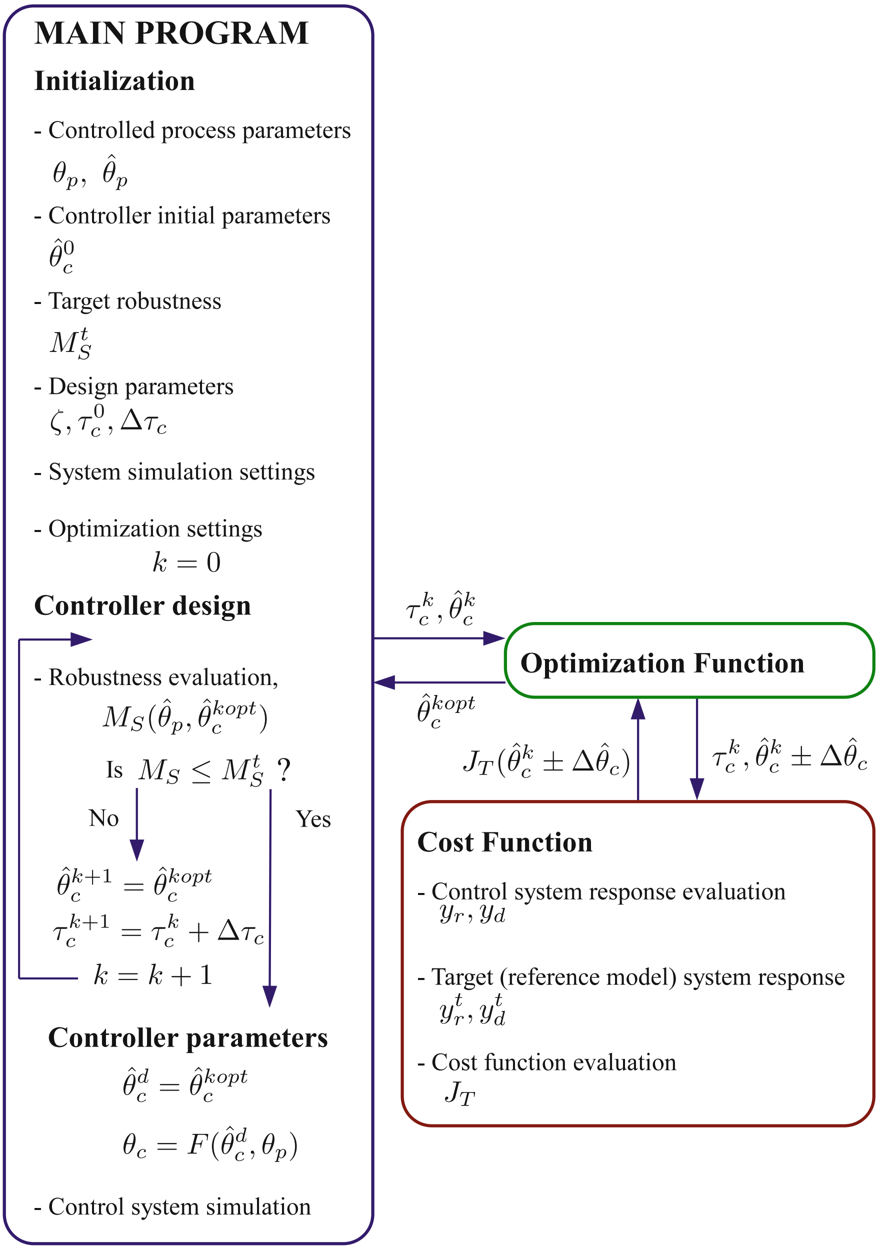



**Fig. 11.1** MoReRT design general work flow

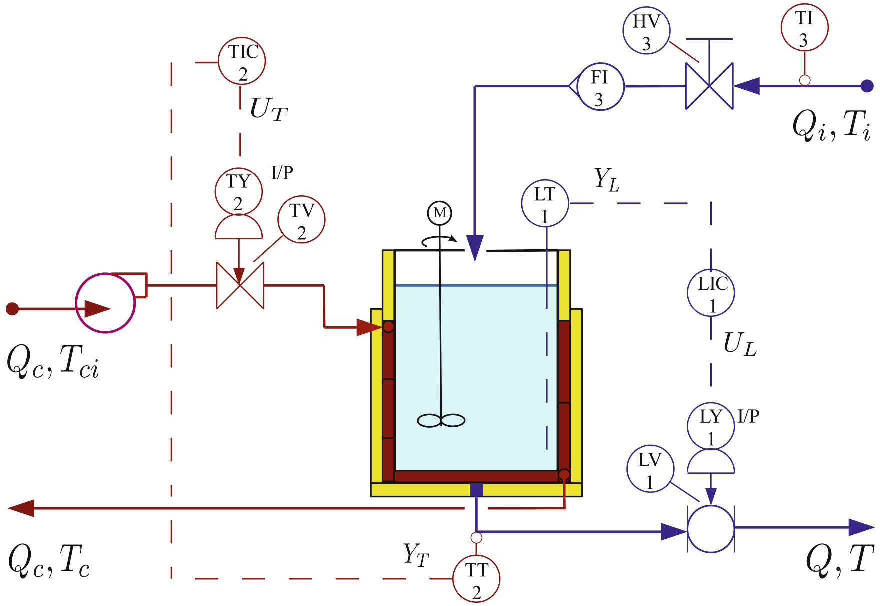



**Fig. 11.2** CSTH control system process and instrument diagram (P&ID)

- Tank mass balance:
  The rate of change of the fluid mass in the tank is given by

$$\rho A \frac{\mathrm{d}H(t)}{\mathrm{d}t} = \rho_i Q_i(t) - \rho Q(t). \tag{11.33}$$

  If the density of the fluid in the tank does not change with temperature ($\rho = \rho_i =$ constant), then

$$A \frac{\mathrm{d}H(t)}{\mathrm{d}t} = Q_i(t) - Q(t). \tag{11.34}$$

- Tank energy balance:
  The energy of the fluid in the tank changes at a rate given by the following equation:

$$\frac{\mathrm{d}[\rho C_p A H(t) T(t)]}{\mathrm{d}t} = \rho C_p T_i(t) Q_i(t) - \rho C_p T(t) Q(t) + W(t), \tag{11.35}$$

  that can be expanded as

$$\rho C_p A H(t) \frac{\mathrm{d}T(t)}{\mathrm{d}t} + \rho C_p A T(t) \frac{\mathrm{d}H(t)}{\mathrm{d}t} = \rho C_p T_i(t) Q_i(t) - \rho C_p T(t) Q(t) + W(t). \tag{11.36}$$

Replacing (11.33) into (11.36) the tank energy balance equation reduces to:

$$\rho C_p A H(t)\frac{\mathrm{d}T(t)}{\mathrm{d}t} = \rho C_p Q_i(t)\,[T_i(t) - T(t)] + W(t). \tag{11.37}$$

- Heating jacket mass balance:
  As the jacket is completely full of the (incompressible) heating fluid

$$\rho_c \frac{\mathrm{d}V_c}{\mathrm{d}t} = \rho_c Q_c(t) - \rho_c Q_{co}(t) = 0, \tag{11.38}$$

then

$$Q_{co}(t) = Q_c(t). \tag{11.39}$$

- Heating jacket energy balance:
  The change of the energy of the heating fluid in the tank heating jacket is given by following relation:

$$\rho_c C_{pc} V_c \frac{\mathrm{d}T_{ca}(t)}{\mathrm{d}t} = \rho_c C_{pc} Q_c(t)\,[T_{ci}(t) - T_{co}(t)] - W(t). \tag{11.40}$$

- Heat transfer from the jacket to the tank

$$W(t) = U A_c\,[T_{ca}(t) - T(t)]\,. \tag{11.41}$$

- Heating fluid average temperature

$$T_{ca}(t) = \frac{T_{ci}(t) + T_{co}(t)}{2}. \tag{11.42}$$

- Transmitters:
  ∗ Level transmitter—For tank fluid level measurement a capacitive type electronic level transmitter installed through the tank head is used and modeling with following relation

$$T_L \frac{\mathrm{d}Y_L(t)}{\mathrm{d}t} + Y_L(t) = K_L H(t). \tag{11.43}$$

  The level transmitter measurement range is from 0 to 0.80 m.
  ∗ Temperature transmitter—For tank fluid temperature measurement an electronic temperature transmitter with a $Pt_{100}$ RTD sensor and a thermowell installed at the tank outlet pipe is used. Its dynamic is of second order given by:

$$T_T^2 \frac{\mathrm{d}^2 Y_T(t)}{\mathrm{d}t^2} + 2T_T \frac{\mathrm{d}Y_T(t)}{\mathrm{d}t} + Y_T(t) = K_T T(t). \tag{11.44}$$

  The temperature transmitter measurement range is from 0 to 50°C.

- Control valves:
  ∗ Level control valve—A normally close ball valve with a pneumatic actuator and a current-to-pressure transducer is used to manipulate the tank outlet flow. Valve inherent flow characteristics is nearly quadratic and it is represented by following relations:

$$T_{vL}\frac{\mathrm{d}X_L(t)}{\mathrm{d}t} + X_L(t) = K_{xL}U_L(t), \tag{11.45}$$

$$Q(t) = K_{vL}X_L^2(t)\sqrt{\rho g H(t)}. \tag{11.46}$$

  ∗ Temperature control valve—A normally close globe valve with a pneumatic actuator and a current-to-pressure positioner is used to manipulate the heating fluid. Valve has an equal-percentage inherent flow characteristics and it is represented by following relations:

$$T_{vT}\frac{\mathrm{d}X_T(t)}{\mathrm{d}t} + X_T(t) = K_{xT}U_T(t), \tag{11.47}$$

$$Q_c(t) = K_{vT}R_{vT}^{[X_T(t)-1]}\sqrt{P_{cp} - [R_c Q_c^2(t) + P_{cr}]}. \tag{11.48}$$

- Variables of interest:
  – Controlled variables: tank fluid level $H(t)$ and temperature $T(t)$,
  – Manipulated variables: tank fluid outlet flow rate $Q(t)$ and jacket heating fluid flow rate $Q_c(t)$,
  – Disturbances: tank inlet fluid flow rate $Q_i(t)$ (system load) and temperature $T_i(t)$, and heating fluid inlet temperature $T_{ci}(t)$.

- Parameters and variables:
  – $\rho$: tank fluid density, $\rho = 1200$ kgm$^{-3}$,
  – $\rho_c$: heating fluid density, $\rho_c = 800$ kgm$^{-3}$,
  – $A$: tank inside section area, $A = \pi D^2/4$ m$^2$,
  – $A_c$: jacket heat transfer area, $A_c = \pi D H_c + \pi D^2/4$ m$^2$,
  – $C_p$: tank fluid heat capacity, $C_p = 4190$ Jkg$^{-1}$°C$^{-1}$,
  – $C_{pc}$: heating fluid heat capacity, $C_{pc} = 2400$ Jkg$^{-1}$°C$^{-1}$,
  – $D$: tank diameter, $D = 0.30$ m,
  – $g$: gravity constant, $g = 9.8$ ms$^{-2}$,
  – $H$: level of fluid in the tank, m,
  – $H_c$: heating jacket height, $H_c = 0.60$ m,
  – $H_t$: tank wall height, $H_t = 0.90$ m,
  – $H_{sp}$: level controller set-point, $H_{sp} = 0.70$ m,
  – $K_L$: level transmitter gain, $K_L = 125$ %m,
  – $K_T$: temperature transmitter gain, $K_T = 2.0$ %/°C,
  – $K_{vL}$: level control valve constant, $K_{vL} = 1.5 \times 10^{-5}$,
  – $K_{vT}$: temperature control valve constant, $K_{vT} = 3 \times 10^{-6}$,
  – $K_{xL}$: level control valve stem constant, $K_{xL} = 0.01/\%$,
  – $K_{xT}$: temperature control valve stem constant, $K_{xT} = 0.01/\%$,

- $P_{cp}$: heating fluid pump discharge pressure, $P_{cp} = 414$ kpa,
- $P_{cr}$: heating fluid system return pressure, $P_{cr} = 138$ kpa;
- $Q$: tank outlet fluid flow rate, $m^3s^{-1}$,
- $Q_c$: jacket heating fluid flow rate, $m^3s^{-1}$,
- $Q_i$: tank inlet fluid flow rate, $6 \times 10^{-4}$ $m^3s^{-1} \leq Q_i \leq 7.5 \times 10^{-4}$ $m^3s^{-1}$, normal tank inlet fluid flow rate $Q_i^n = 7 \times 10^{-4}$ $m^3s^{-1}$,
- $R_c$: heating system pipe nominal flow resistance, $R_c = 5.5 \times 10^7$ kPa/(m$^3$/s)$^2$
- $R_{vT}$: temperature control valve rangeability, $R_{vT} = 50$;
- $t$: time, s,
- $T$: temperature of fluid in the tank, °C,
- $T_{sp}$: temperature controller set-point, $T_{sp} = 38$ °C
- $T_i$: fluid inlet temperature, 22 °C $\leq T_i \leq$ 26 °C, temperature $T_i^n = 24$ °C,
- $T_c$: heating fluid temperature, °C,
- $T_{ca}$: heating fluid average temperature, °C,
- $T_{ci}$: heating fluid inlet temperature, $T_{ci} = 320$ °C,
- $T_{co}$: heating fluid outlet temperature, °C,
- $T_L$: level transmitter time constant, $T_L = 2$ s,
- $T_T$: temperature transmitter time constant, $T_T = 15$ s,
- $T_{vL}$: level control valve time constant, $T_{vL} = 3$ s,
- $T_{vT}$: temperature control valve time constant, $T_{vT} = 5$ s
- $U$: overall heat-transfer coefficient, $U = 440$ Js$^{-1}$m$^{-2}$°C$^{-1}$,
- $U_L$: level controller output signal, %,
- $U_T$: temperature controller output signal, %,
- $V_c$: heat jacket volume, $V_c = H_c\pi[(D + 2W_c)^2 - D^2]/4 + W_c\pi(D + 2Wc)^2/4$ m$^3$,
- $W$: rate of heat transfer from jacket to tank, Js$^{-1}$,
- $W_c$: heating jacket wide, $W_c = 0.02$ m,
- $X_L$: level control valve stem normalized travel, $0 \leq X_L \leq 1$,
- $X_T$: temperature control valve stem normalized travel, $0 \leq X_T \leq 1$,
- $Y_L$: level transmitter output signal, %,
- $Y_T$: temperature transmitter output signal, %,

The CSTH model includes seventeen variables, five of them are input variables, related by eight nonlinear differential equations and four nonlinear algebraic equations.

**Steady-State Operation Conditions**

The steady-state operation points of the tank level system for the minimum ($Q_i^m$), normal ($Q_i^n$), and maximum ($Q_i^M$) inlet flow rates are listed in Table 11.1.

To maintain the tank level at the set-point $H_{sp} = 0.70$ m over the entire inlet fluid flow rate range ($6.0 \times 10^4$ $m^3s^{-1} \leq Q_i \leq 7.5 \times 10^4$ $m^3s^{-1}$) the level control valve opening range is $66.40\,\% \leq X_L \leq 74.23\,\%$.

For the heating system the steady-state operation points for the inlet flow rate and temperature extreme minimum ($Q_i^m$, $T_i^M$), normal ($Q_i^n$, $T_i^n$), and extreme maximum ($Q_i^M$, $T_i^m$) combinations are listed in Table 11.2.

**Table 11.1** Tank fluid level operation points

| | $Q_i^m$ | $Q_i^n$ | $Q_i^M$ |
|---|---|---|---|
| $Q^o$ ($m^3s^{-1}$) | $6.0 \times 10^{-4}$ | $7.0 \times 10^{-4}$ | $7.5 \times 10^{-4}$ |
| $X_L^o$ | 0.6640 | 0.7172 | 0.7423 |
| $U_L^o$ (%) | 66.40 | 71.72 | 74.23 |

$H_{sp} = 0.70$ m
$Y_L^o = 87.50$ %

**Table 11.2** Heating system operation points

| | $(Q_i^m, T_i^M)$ | $(Q_i^n, T_i^n)$ | $(Q_i^M, T_i^m)$ |
|---|---|---|---|
| $Q_c^o$ ($m^3s^{-1}$) | $6.97 \times 10^{-5}$ | $1.56 \times 10^{-4}$ | $4.19 \times 10^{-4}$ |
| $T_{ca}^o$ (°C) | 167.33 | 214.03 | 253.55 |
| $T_{co}^o$ (°C) | 14.66 | 108.07 | 187.10 |
| $W^o$ $Js^{-1}$ | 36202 | 49274 | 60336 |
| $X_T^o$ | 0.1719 | 0.3440 | 0.5151 |
| $U_T^o$ (%) | 17.19 | 34.40 | 51.51 |

$T_{sp} = 38$ °C
$Y_T^o = 76$ %

As the fluid flow rate range is $6.0 \times 10^4$ $m^3s^{-1}$ $\leq Q_i \leq 7.5 \times 10^4$ $m^3s^{-1}$ and the inlet temperature range 22.0 °C $\leq T_i \leq$ 26.0 °C then, to maintain the fluid temperature at its set-point $T_{sp} = 38$ °C the required temperature control valve travel is $17.19\,\% \leq X_T \leq 51.51\,\%$.

### Dynamic Characteristics

Although the CSTH has two controlled variables and two associated manipulated variables, $(H,\ Q)$, $(T,\ Q_c)$, the system is not interacting. The level control system of the fluid in the tank perturbs its temperature control system but this last one does not affect the fluid level. This is schematically shown in Fig. 11.3.

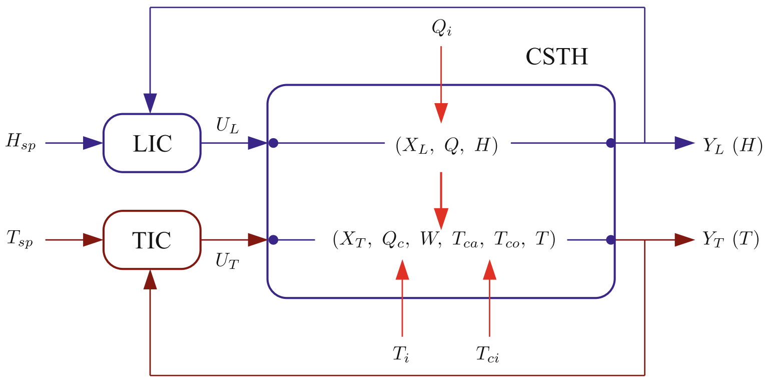



**Fig. 11.3** CSTH control system schematic

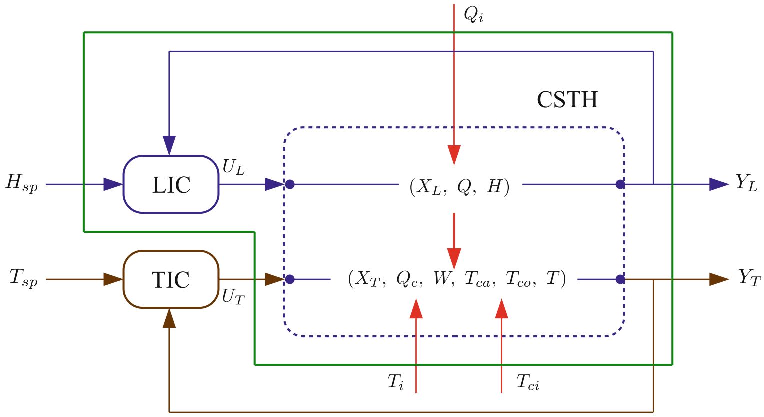



**Fig. 11.4** Controlled process seen by the temperature controller

Changes in the inlet flow temperature $T_i$, the heating fluid temperature $T_{ci}$, and the heating fluid flow rate $Q_c$ do not affect the fluid level in the tank. On the other hand, the inlet fluid flow rate $Q_i$ affects the fluid level $H$, the heat transfer from the jacket to the tank fluid, and then the fluid temperature $T$.

From Fig. 11.3 it seems that the CSTH model must include transfer functions representing the $U_L \to Y_L$, $U_L \to Y_T$, $Q_i \to Y_L$, $Q_i \to Y_T$, $U_T \to Y_T$, $T_i \to Y_T$, and $T_{ci} \to Y_T$ dynamics but changes in the inlet flow rate $Q_i$ and level control set-point $H_{sp}$ are filtered by the tank level control system.

The controlled process "seen" by the temperature controller (TIC) includes the whole level control system as depicted in Fig. 11.4. Then, we must consider first the tank fluid level control system.

### CSTH Level Control System

To obtain the controlled process models for the level control system we must consider the three operating points $\{Q_i^m,\ Q_i^n,\ Q_i^M\}$ and changes in the level control output signal $U_L$ ($\delta U_L = \pm 2\,\%$) and in the inlet fluid flow rate $Q_i$ ($\delta Q_i = \pm 5\,\%$). All these responses are first-order as shown in Fig. 11.5 for the $(Q_i^n,\ \delta U_L)$ case.

From the process reaction curves and using a two-point identification method [28] first-order models are obtained whose parameters are listed in Table 11.3. All identified models are first-order without dead-time. Process gain and time constant depend on the operation point and on the type and sign of the input signal changes.

At the normal operating point the $\delta U_L \to \delta Y_L$ and $\delta Q_i \to \delta Y_L$ average models are:

$$\tilde{P}_{Y_L U_L}(s) = \frac{-4.90}{146.24s + 1}, \tag{11.49}$$

$$\tilde{P}_{Y_L Q_i}(s) = \frac{2.50\ 10^5}{142.35s + 1}. \tag{11.50}$$

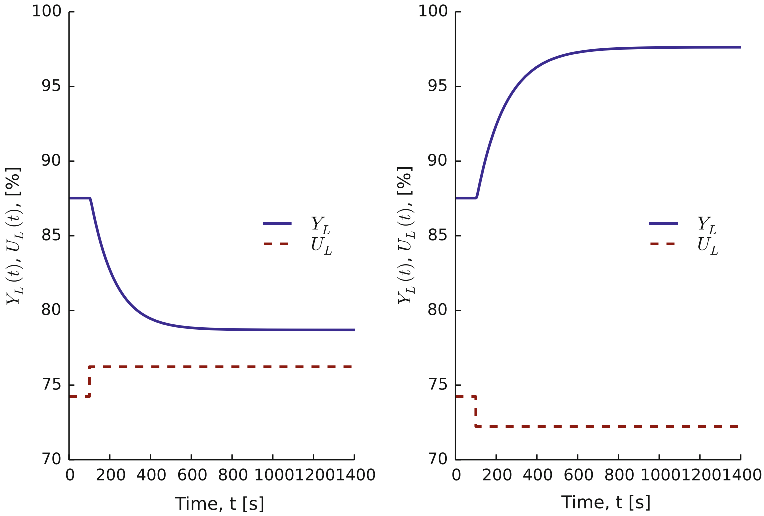



**Fig. 11.5** Controlled process reaction curve ($\delta Y_L/\delta U_L$)

**Table 11.3** CSTH models for level control

| | $+\Delta U_L$ | $-\Delta U_L$ | $+\Delta Q_i$ | $-\Delta Q_i$ |
|---|---|---|---|---|
| $Q_i = Q_i^m$ | | | | |
| $K$ | –4.90 | –5.69 | $2.99 \times 10^5$ | $2.84 \times 10^5$ |
| $T$ | 153.07 | 186.88 | 171.63 | 159.99 |
| $Q_i = Q_i^n$ | | | | |
| $K$ | –4.56 | –5.24 | $2.57 \times 10^5$ | $2.43 \times 10^5$ |
| $T$ | 132.77 | 159.70 | 147.75 | 136.94 |
| $Q_i = Q_i^M$ | | | | |
| $K$ | –4.42 | –5.05 | $2.39 \times 10^5$ | $2.28 \times 10^5$ |
| $T$ | 124.63 | 148.94 | 138.70 | 128.29 |

From these average models the changes in the process characteristics over the entire system operation range are [$K$ (–9.9 %, +16.1 %), $T$ (–14.8 %, +27.8 %)] for $\delta U_L$ and [$K$ (–9.0 %, +19.5 %), $T$ (–9.9 %, +20.8 %)] for $\delta Q_i$ that are not very severe. Robustness requirements are set to the minimum level.

Then, for the design of the tank fluid level control system depicted in Fig. 11.6 following models are used

$$P_{Y_L U_L}(s) = \frac{K_{Y_L U_L}}{Ts+1} = \frac{-4.90}{144.30s+1}, \tag{11.51}$$

$$P_{Y_L Q_i}(s) = \frac{K_{Y_L Q_i}}{Ts+1} = \frac{2.50\ 10^5}{144.30s+1} = -0.51 \times 10^5\ P_{Y_L U_L}(s), \tag{11.52}$$

with time constant $T$ in s and gains $K_{Y_L U_L}$ in %/% and $K_{Y_L Q_i}$ in %(m$^3$/s).

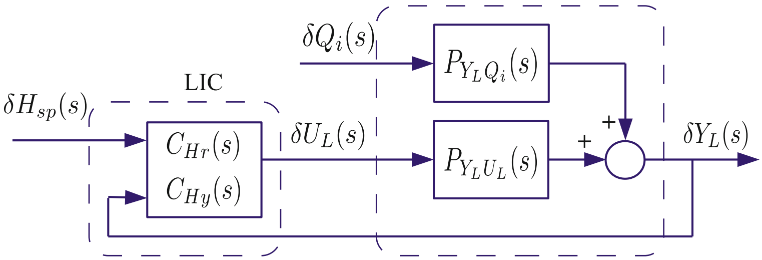



**Fig. 11.6** CSTH level control system

To control a first-order process a proportional-integral controller is adequate. For its tuning an analytical deducted rule for 2DoF PI controller is used with equations [21]:

$$\kappa_p \doteq K_p K_{Y_L U_L} = \frac{2-\tau_c}{\tau_c}, \tag{11.53}$$

$$\tau_i \doteq \frac{T_i}{T} = \tau_c(2-\tau_c), \tag{11.54}$$

$$\beta = \frac{1}{2-\tau_c}, \tag{11.55}$$

where $\tau_c < 2$ is the design parameter. Using the (11.53)–(11.55) tuning the closed-loop transfer functions are:

$$M_{Y_L Q_i}(s) = \frac{K_{Y_L Q_i}\tau_c^2 T\ s}{(\tau_c T s + 1)^2}, \tag{11.56}$$

$$M_{Y_L U_L}(s) = \frac{1}{\tau_c T s + 1}. \tag{11.57}$$

Robustness of a first-order process PI control system is not a concern, it is extremely high. Then, the level control design is made taking into account following performance indices:

- Control system speed (closed-loop time constant)

$$T_c = \tau_c T. \tag{11.58}$$

- Regulatory control maximum error (for a critical damped response)

$$E_{Lmax} = 0.368 K_{Y_L Q_i}\tau_c \Delta Q_i. \tag{11.59}$$

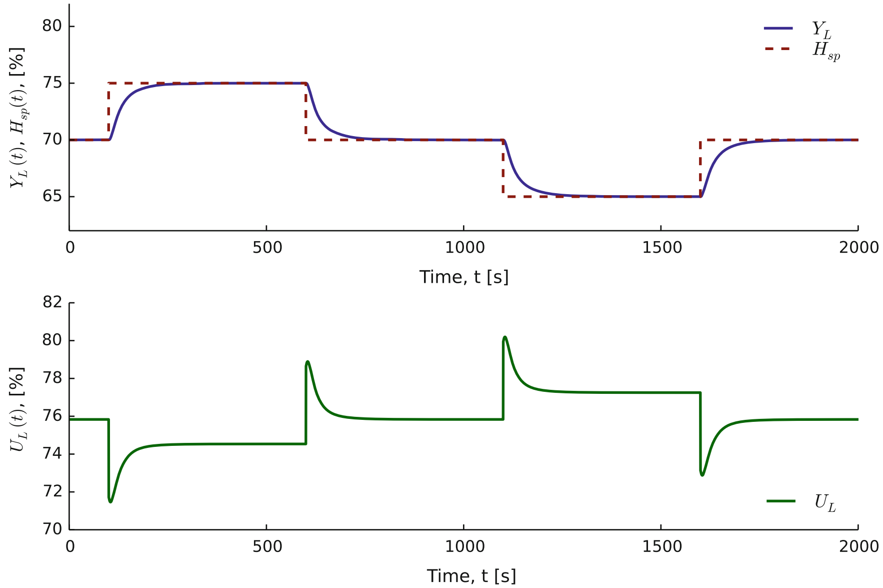



**Fig. 11.7** Level control response to set-point changes

- Controller output abrupt change on a step set-point modification

$$\Delta U_L^{0+} = K_p \beta \Delta H_{sp} = \frac{1}{K_{Y_L U_L} \tau_c} \Delta H_{sp}. \tag{11.60}$$

From (11.58) to (11.60) it is seen that if the design parameter $\tau_c \downarrow$ then $T_c \downarrow$ (faster system) and $E_{Lmax} \downarrow$ (lower maximum error) but $\Delta U_L^{0+} \uparrow$ (higher controller proportional "kick"). If we want to limit the controller output abrupt change said to be $\Delta U_L^{0+} \leq \Delta H_{sp}$ then, the design parameter must be in the range $1/K_{Y_L U_L} \leq \tau_c < 2$. Selecting $\tau_c = 0.25$ the $PI_2$ controller parameters are: $K_p = 1.43$, $T_i = 63.20$ s, and $\beta = 0.57$. The Action of the controller must be Direct. Figure 11.7 shows the tank level control operation to a set of set-point $H_{sp}$ changes.

The corresponding operation of the tank level control system to the disturbance $\delta Q_i$ is shown in Fig. 11.8.

With the level control system working in automatic mode it is necessary to conduct the tests to obtain the dynamic relations between the temperature transmitter output $Y_T$ and the possible inputs $U_T$, $T_i$, $T_{ci}$, $Q_i$, and $H_{sp}$ (see Fig. 11.4).

### CSTH Temperature Control System

To obtain the controlled process models for design the fluid temperature control system a total of ten reaction curves are obtained at the normal operating point. The

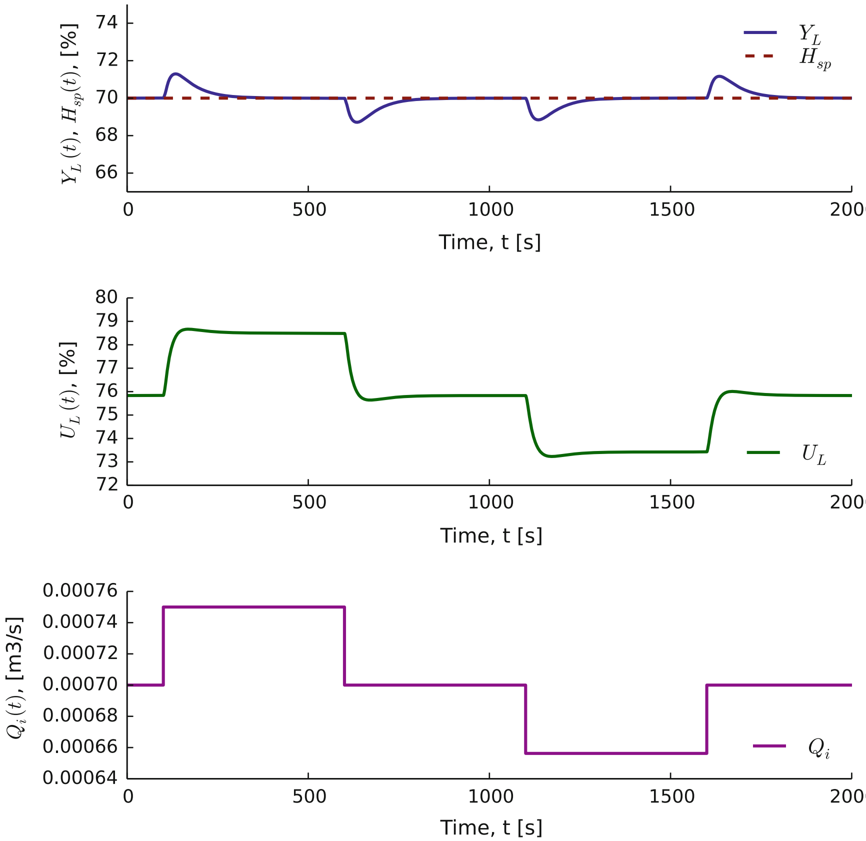



**Fig. 11.8** Level control response to disturbance changes

input changes are $\delta U_t = \pm 5\,\%$, $\delta T_i = \pm 2\,°\mathrm{C}$, $\delta T_{ci} = \pm 5\,\% = 16\,°\mathrm{C}$, $\delta Q_i = \pm 5\,\%$, and $\delta H_{sp} = \pm 5\,\%$.

The process dynamics are of second-order as shown in Fig. 11.9 for the $(\delta Y_T, \delta U_T)$ case and in Fig. 11.10 for the $(\delta Y_T, \delta T_i)$ case. The identified parameters for the models are listed in Table 11.4.

As expected, changes in the temperature controller output $U_T$ and fluid inlet temperature $T_i$ affect significantly the fluid temperature $T$ while the influence of a change in the heating fluid temperature $T_{ci}$ is approximately 20 times lower than the influence of a change in $T_i$. A change in the inlet flow $Q_i$ also has a important influence over $T$ but changes in the level control system set-point $H_{sp}$ show that they do not have any influence on the tank fluid temperature.

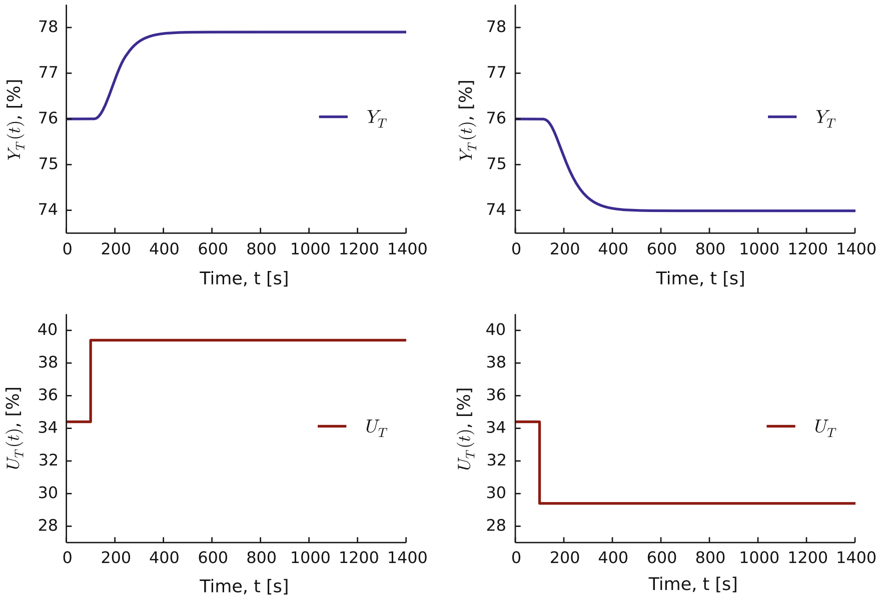



**Fig. 11.9** Process reaction curve ($\delta Y_T/\delta U_T$)

Table 11.4 also includes the sum of all the model time related parameters $T_T = (1 + a + \tau_L)T$ in s as an indication of the speed of the particular dynamics. At the normal operating point the $\delta U_T \rightarrow \delta Y_T$, $\delta T_i \rightarrow \delta Y_T$, $\delta T_{ci} \rightarrow \delta Y_T$, and $\delta Q_i \rightarrow \delta Y_T$, average models of the tank fluid temperature control system depicted in Fig. 11.11 are:

$$\tilde{P}_{Y_T U_T}(s) = \frac{0.39\mathrm{e}^{-27.59s}}{(48.89s + 1)(48.89s + 1)}, \tag{11.61}$$

$$\tilde{P}_{Y_T T_i}(s) = \frac{1.91\mathrm{e}^{-4.64s}}{(46.38s + 1)(31.54s + 1)}, \tag{11.62}$$

$$\tilde{P}_{Y_T T_{ci}}(s) = \frac{0.095\mathrm{e}^{-24.87s}}{(48.65s + 1)(48.65s + 1)}, \tag{11.63}$$

$$\tilde{P}_{Y_T Q_i}(s) = \frac{-3.82\ 10^4\mathrm{e}^{-4.63s}}{(46.48 + 1)(31.35s + 1)}. \tag{11.64}$$

To analyze changes in the process dynamics two extreme cases are considered as was made for the steady-state conditions. The lowest heating demanding condition ($Q_i^m$, $T_i^M$) and the highest heating demanding condition ($Q_i^M$, $T_i^m$). Then, 16 additional reaction curves are obtained. Parameters of the obtained models are listed in Table 11.5.

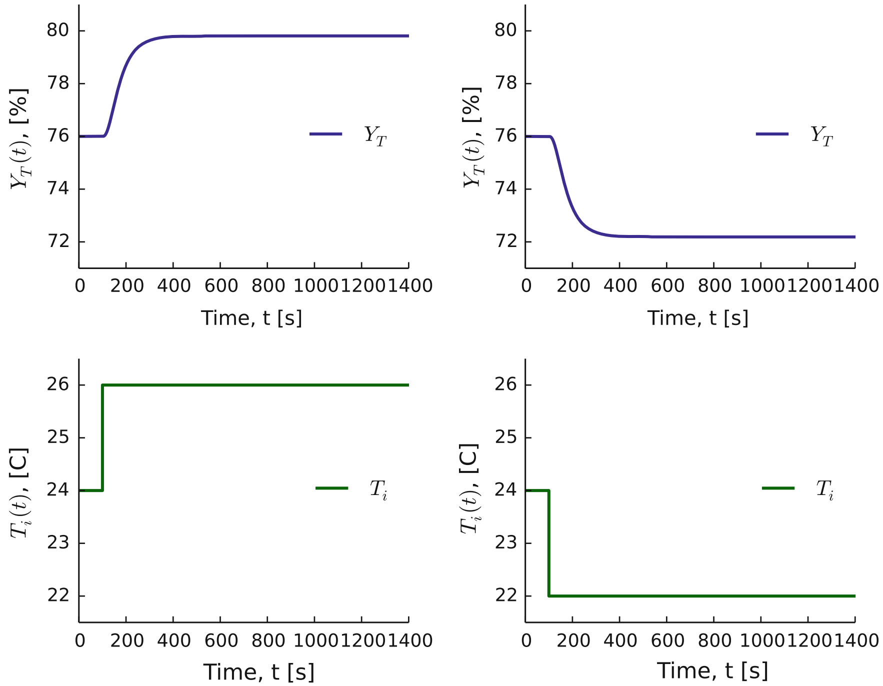



**Fig. 11.10** Process reaction curve ($\delta Y_T / \delta T_i$)

**Table 11.4** CSTH models for temperature control (normal operation point)

| | $K$ | $T$ | $a$ | $\tau_L$ | $T_T$ |
|---|---|---|---|---|---|
| $+\Delta U_T$ | 0.38 | 46.78 | 1.0 | 0.58 | 120.69 |
| $-\Delta U_T$ | 0.40 | 51.00 | 1.0 | 0.55 | 130.05 |
| $+\Delta T_i$ | 1.91 | 46.38 | 0.68 | 0.10 | 82.56 |
| $-\Delta T_i$ | 1.91 | 46.38 | 0.68 | 0.10 | 82.56 |
| $+\Delta T_{ci}$ | 0.095 | 48.65 | 1.0 | 0.47 | 120.17 |
| $-\Delta T_{ci}$ | 0.095 | 48.65 | 1.0 | 0.47 | 120.17 |
| $+\Delta Q_i$ | $-3.64 \times 10^4$ | 44.68 | 0.69 | 0.11 | 80.42 |
| $-\Delta Q_i$ | $-4.00 \times 10^4$ | 48.28 | 0.66 | 0.09 | 84.49 |

The control system stability depends only on the $C_y(s)$ controller parameters and on the $P_{Y_T U_T}(s)$ model parameters. From the relative stability point of view the worst case corresponds to $P_{Y_T U_T}(s)$ highest gain $K_{Y_T U_t}$, lowest sum of time constants $(1 + a)T_{Y_T U_t}$, and highest dead-time $L_{Y_T U_t}$.

From Table 11.5 the $P_{Y_T U_T}(s)$ model highest deviations are: $\Delta K_{Y_T U_T} = +23\,\%$, $\Delta(1 + a)T_{Y_T U_t} = -33.2\,\%$, and $\Delta L_{Y_T U_t} = +6.6\,\%$ but not all of them occur at the

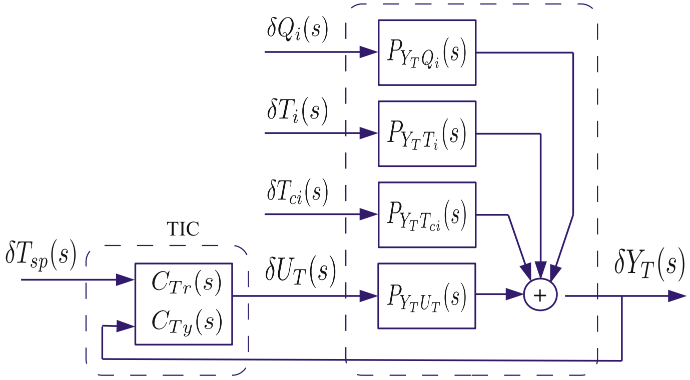



**Fig. 11.11** CSTH temperature control system

**Table 11.5** CSTH models for temperature control (extreme cases)

| | $K$ | $T$ | $a$ | $\tau_L$ | $T_T$ |
|---|---|---|---|---|---|
| $(Q_i^m, T_i^M)$ | | | | | |
| $+\Delta U_T$ | 0.48 | 58.79 | 1.0 | 0.49 | 146.39 |
| $-\Delta U_T$ | 0.48 | 67.80 | 0.93 | 0.38 | 156.62 |
| $+\Delta T_i$ | 1.92 | 63.27 | 0.35 | 0.14 | 94.27 |
| $-\Delta T_i$ | 1.92 | 63.27 | 0.35 | 0.14 | 94.27 |
| $+\Delta T_{ci}$ | 0.082 | 61.20 | 1.0 | 0.40 | 146.88 |
| $-\Delta T_{ci}$ | 0.082 | 61.20 | 1.0 | 0.40 | 146.88 |
| $+\Delta Q_i$ | $-3.66 \times 10^4$ | 60.11 | 0.38 | 0.14 | 91.37 |
| $-\Delta Q_i$ | $-4.03 \times 10^4$ | 66.81 | 0.31 | 0.14 | 96.87 |
| $(Q_i^M, T_i^m)$ | | | | | |
| $+\Delta U_T$ | 0.26 | 41.22 | 0.98 | 0.52 | 103.05 |
| $-\Delta U_T$ | 0.29 | 42.59 | 1.0 | 0.60 | 110.73 |
| $+\Delta T_i$ | 1.89 | 42.39 | 0.72 | 0.13 | 78.42 |
| $-\Delta T_i$ | 1.89 | 42.39 | 0.72 | 0.13 | 78.42 |
| $+\Delta T_{ci}$ | 0.11 | 43.52 | 0.88 | 0.46 | 101.84 |
| $-\Delta T_{ci}$ | 0.11 | 43.52 | 0.88 | 0.46 | 101.84 |
| $+\Delta Q_i$ | $-3.56 \times 10^4$ | 40.94 | 0.73 | 0.14 | 76.56 |
| $-\Delta Q_i$ | $-4.23 \times 10^4$ | 44.01 | 0.70 | 0.12 | 80.10 |

same operating point. Considering this for the temperature control system design a robustness target level $M_S^t = 2.0$ is used.

From Tables 11.4 and 11.5 it is seen that the main disturbances of the fluid temperature $T$ are $T_i$ and $Q_i$ and that the $\delta T_i \rightarrow \delta Y_T$ and $\delta Q_i \rightarrow \delta Y_T$ dynamics are the same except for the gains due only to the disturbances different physical nature. The controlled process models for the temperature controller tuning are:

$$P_{Y_T U_T}(s) = \frac{0.39\mathrm{e}^{-27.59s}}{(48.89s+1)(48.89s+1)}, \tag{11.65}$$

$$P_{Y_T T_i}(s) = \frac{1.91\mathrm{e}^{-4.64s}}{(46.38s+1)(31.54s+1)}, \tag{11.66}$$

$$P_{Y_T Q_i}(s) = \frac{-3.82\times 10^4\mathrm{e}^{-4.63s}}{(46.48+1)(31.35s+1)}, \tag{11.67}$$

with time constants and dead-time in s, $K_{Y_T U_T}$ in %/°C, and $K_{Y_T Q_i}$ in $\%/(\mathrm{m}^3/\mathrm{s})$.

To produce a temperature change $\Delta T = 1$ °C ($\Delta Y_T = 2\,\%$) it is required a $\Delta U_T = 5.138\,\%$, a $\Delta T_i = 1.047$ °C, or a $\Delta Q_i = 5.236\times 10^{-5}$ $\mathrm{m}^3\mathrm{s}^{-1}$. The temperature control system controlled variable is given by following relation:

$$\delta Y_T(s) = \frac{C_{Tr}(s)P_{Y_T U_T}(s)}{1+C_{Ty}(s)P_{Y_T U_T}(s)}\delta T_{sp}(s) + \frac{P_{Y_T T_i}(s)}{1+C_{Ty}(s)P_{Y_T U_T}(s)}\delta T_i(s) + \frac{P_{Y_T Q_i}(s)}{1+C_{Ty}(s)P_{Y_T U_T}(s)}\delta Q_i(s). \tag{11.68}$$

Considering that the two disturbances $\{\delta T_i, \delta Q_i\}$ have the same dynamics, the temperature controller can be designed considering only one of them. Using the process gain $K = 0.39$ and main time constant $T = 48.89$ s the processes normalized models used for the controller design are:

$$\hat{P}_{Y_T U_T}(\hat{s}) = \frac{\mathrm{e}^{-0.56\hat{s}}}{(\hat{s}+1)^2}, \tag{11.69}$$

$$\hat{P}_{Y_T T_i}(\hat{s}) = \frac{4.90\mathrm{e}^{-0.095\hat{s}}}{(0.95\hat{s}+1)(0.65\hat{s}+1)}. \tag{11.70}$$

In this case, $P_d(s) \neq P_u(s)$ and the closed-loop target transfer functions must be selected using (10.24) and (10.26) presented in Sect. 10.2.

Based on normalized models (11.69) and (11.70) the target closed-loop transfer functions are selected as:

$$M^t_{Y_T T_{sp}}(\hat{s}) = \frac{\mathrm{e}^{-0.56\hat{s}}}{(\tau_c\hat{s}+1)^2}, \tag{11.71}$$

$$M^t_{Y_T T_i}(\hat{s}) = \frac{4.90(\hat{s}+1)^2(\tau_i/\kappa_p)\hat{s}\mathrm{e}^{-0.095\hat{s}}}{(0.95\hat{s}+1)(0.65\hat{s}+1)(\tau_c\hat{s}+1)^3}, \tag{11.72}$$

to obtain non-oscillating responses.

Starting with controller parameters $\kappa_p^0 = 1.0$, $\tau_i^0 = 1.0$, $\tau_d^0 = 0.25$, and $\beta^0 = 1.0$, and a design parameter $\tau_c^0 = 1.0$ it is found the the resulting controller normalized parameters are $\kappa_p = 1.440$, $\tau_i = 2.099$, $\tau_d = 0.617$, and $\beta = 0.485$, and the control system robustness $M_S = 1.553$ that is much higher than the selected target robustness. Then, we can speed-up the control system, and consequently reduce its robustness, by using a lower design parameter.

**Table 11.6** CSTH MoReRT temperature controller parameters and robustness

| $\tau_c$ | $\kappa_p$ | $\tau_i$ | $\tau_d$ | $\beta$ | $M_S$ |
|---|---|---|---|---|---|
| 0.700 | 2.477 | 1.893 | 0.625 | 0.393 | 2.457 |
| 0.725 | 2.379 | 1.924 | 0.626 | 0.398 | 2.336 |
| 0.750 | 2.282 | 1.952 | 0.626 | 0.404 | 2.226 |
| 0.775 | 2.186 | 1.978 | 0.626 | 0.410 | 2.126 |
| 0.800 | 2.093 | 2.001 | 0.627 | 0.416 | 2.036 |
| 0.825 | 2.002 | 2.022 | 0.627 | 0.423 | 1.954 |
| 0.875 | 1.827 | 2.056 | 0.626 | 0.438 | 1.813 |
| 0.975 | 1.512 | 2.096 | 0.620 | 0.475 | 1.596 |
| 1.000 | 1.440 | 2.099 | 0.617 | 0.485 | 1.553 |
| 1.100 | 1.179 | 2.087 | 0.602 | 0.533 | 1.411 |
| 1.200 | 0.956 | 2.032 | 0.575 | 0.595 | 1.306 |

The MoReRT controller parameters and control system robustness obtained with different $\tau_c$ design parameters are listed in Table 11.6.

Selecting ($\tau_c = 0.80$) $\kappa_p = 2.093$, $\tau_i = 2.001$, $\tau_d = 0.627$, and $\beta = 0.416$, the parameters for the temperature controller (TIC) tuning are $K_p = 5.37$, $T_i = 97.83$ s, $T_d = 30.65$ s, $\beta = 0.42$, and Reverse Action.

The temperature ($Y_T$) and level ($Y_L$) control system responses and controller outputs ($U_T, U_L$) to changes in the temperature set-point ($T_{sp}$), and inlet fluid temperature ($T_i$) and flow rate ($Q_i$) disturbances are shown in Fig. 11.12.

## 11.4 Chapter Remarks

The two-degree-of-freedom proportional integral derivative control algorithms implemented in commercial controllers of eight manufactures are presented. Most of they made use of the PID Standard algorithm that can be aggregated with set-point and controlled variable input signal filters. Approximately half of these manufactures allow to use a proportional set-point factor $\beta > 1$, but the others constraint it to be in the range $0 \leq \beta \leq 1$. This constraint would reduce the achievable servo-control performance in cases that a highly robust control system is required.

The MoReRT tuning equations for 2DoF PI and PID control algorithms to control a diversity of controlled process are summarized for easy using.

The steps to follow to applied the MoReRT design methodology to and specific controlled process/controller combination not covered by the available tuning rules are presented. Based on these the user may implement its own design routine.

The controlled process dynamics analysis and identification, and the robust controller design stages are exemplify using a continuous stirred-tank heater (CSTH).

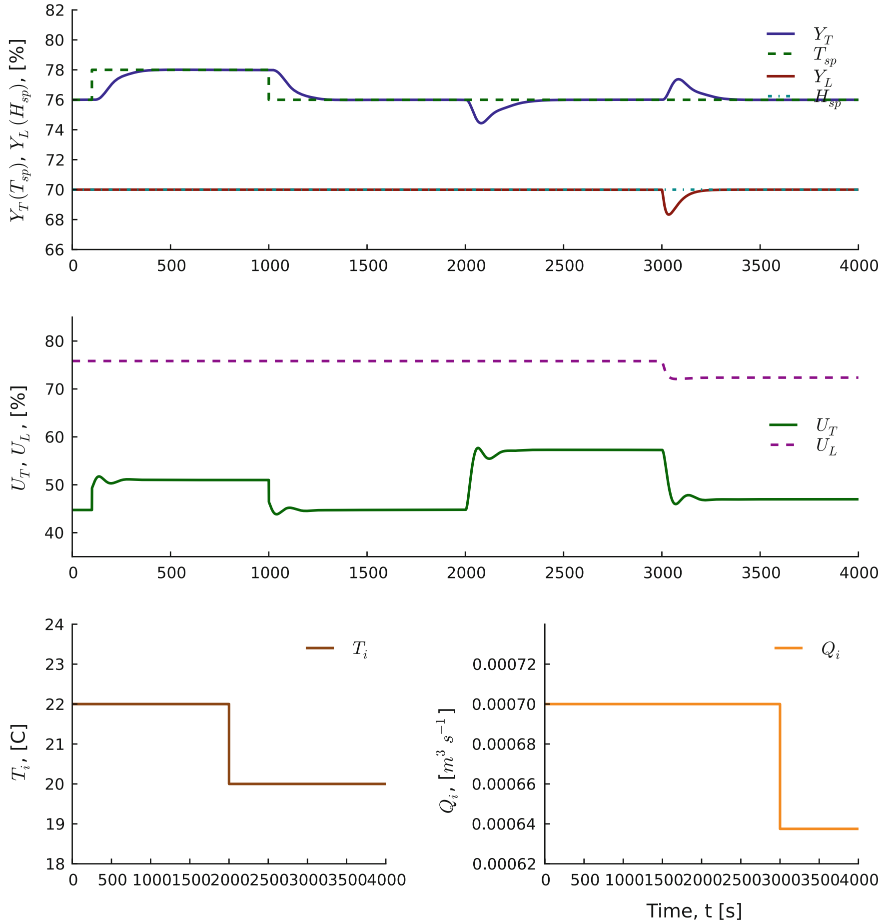



**Fig. 11.12** CSTH control system responses

The control system design made use of a analytically deducted robust tuning method for the tank level $PI_2$ controller and of the MoReRT design for the fluid temperature $PID_2$ controller.

## References

1. ABB Control Technologies: System 800xA 5.1 Product Catalog (FP4 included). Västeras, Sweden (2013)
2. ABB Control Technologies: Functional Description—PID01 Closed Loop Controller. Västeras, Sweden (2012). Document number. 3BTG811792-3032, Version 5.2-0

3. ABB Control Technologies: Funtional Description, PP_ FunctionLib. Västeras, Sweden (2012). Document number: 3BTG811792-3011, Version 5.2-0
4. ABB Control Technologies: Functional Description: PID01A—Closed Loop Controller with Auto Tuner, Version 5.2-0 . Västeras, Sweden (2012). Document number: 3BTG811792-3033
5. ABB Automation Technology Products: Industrial IT 800xA—System, 800xA for Advant Master, System Version 4.0, Graphic Library. Västeras, Sweden (2004). Document number: 3BSE030430R4001
6. Beall, J.: Interesting and Useful Features of the DeltaV PID Controller (n.a.). Emerson Global Users Exchange
7. Emerson Process Management: Key Features of the DeltaV PID Function Block (2011). Delta V Whitepaper
8. Fisher-Rosemount Systems, Inc.: DeltaV Books Online (2006). Version 9.3
9. Mitsubishi Electric Corporation, Industrial Automation: MELSEC System Q - Programmable Logic Controllers, Programming Manual (Process Control Instructions, QnPH CPUs) (2002). Art. no. 149256, SH(NA)-080316, Version A
10. National Instruments Corporation: LabVIEW 2014 Help. 11500 North Mopac Expressway, Austin, Texas, USA (2014). Document No. 371361L-01
11. OMRON Corporation: SYSMAC CS/CJ Series CS1W-LCB01/CS1W-LCB05 Loop Control Boards, CS1D-CPU P Process-control CPU Units, CJ1G-CPU P Loop-control CPU Units—Function Blocks Reference Manual. Control Devices Division H.Q., Shiokoji Horikawa, Shimogyo-ku, Kyoto, 600-8530, Japan (2005). Cat. No. W407-E1-06
12. Balda, P., Schlegel, M.: Advanced PID control algorithms built into the REX control system. In: IFAC Conference on Advances in PID Control (PID'12). 28–30 March, Brescia, Italy (2012)
13. REX Controls s.r.o.: Function Blocks of the REX Control System: Reference manual. Pilsen, Czech Republic (2015). Version 2.10.6
14. Siemens AG: SIMATIC S7-1200 Easy Book. Nürnberg, Germany (2011). A5E02486774-02
15. Siemens AG: SIMATIC S7-1200 Programmable controller, System Manual. Nürnberg, Germany (2012). A5E02486680-06
16. Siemens AG: PID Control with Gain Scheduling and PID Tuning, SIMATIC PCS 7—Application Example. Nürnberg, Germany (2009). 38755162
17. Siemens AG: SIMATIC Process Control System PCS 7–PCS 7 Advanced Process Library V8.0, Function Manual. Nürnberg, Germany (2011). A5E03709256-01
18. Toshiba Corporation: Unified Controller nv series & Integrated Controller V series - Programming Instructions (LD/FBD/SFC/ST). 72–34, Horikawa-Cho, Saiwai-Ku, Kawasaki 212-8585, Japan, 6th edn. (2014)
19. Toshiba Corporation: Unified Controller - Single-Loop Controller LC531/LC532 - Instruction Manual. 72-34, Horikawa-cho, Saiwai-Ku, Kawasaki 212-8585, Japan, 4th edn. (2014)
20. Alfaro, V.M., Vilanova, R., Arrieta, O.: Considerations on set-point weight choice for 2-DoF PID controllers. In: IFAC International Symposium on Advanced Control on Chemical Process (ADCHEM 2009). 12–15 July, Istambul, Turkey (2009)
21. Alfaro, V.M., Vilanova, R., Arrieta, O.: Analytical robust tuning of PI controllers for First-Order-Plus-Dead-Time processes. In: 13th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA2008). Sept 15–18, Hamburg, Germany (2008)
22. Alfaro, V.M., Vilanova, R., Arrieta, O.: A single-parameter robust tuning approach for Two-Degree-of-Freedom PID controllers. In: European Control Conference (ECC2009). Aug 23–26, Budapest, Hungary (2009)
23. Alfaro, V.M., Balaguer, P., Arrieta, O.: Robustness considerations on pid tunins for regulatory control of inverse response processes. In: IFAC Conference on Advanced in PID Control (PID'12). IFAC Conference on Advanced in PID Control (PID'12), Bresia, Italy, 28–30 March (2012)
24. Bequette, B.W.: Process Dynamics: Modeling, Analysis, and Simulation. Prentice Hall, Inc., Upper Saddle River, New Jersey 07458, USA (1998)
25. Coughanowr, D.R., LeBlanc, S.E.: Process systems analysis and control, 3rd edn. McGraw-Hills, Inc., 1221 Avenue of the Americas, New York, NY 10020, USA (2009)

26. Luyben, W.L.: Process Modeling, Simulation, and Control for Chemical Engineers, 2nd edn. McGraw-Hill, Inc. (1999)
27. Seborg, D.E., Edgar, T.F., Mellichamp, D.A.: Process Dynamics and Control, 2nd edn. Wiley, Inc., Hoboken, NJ 07030, USA (2004)
28. Alfaro, V.M.: Low-Order Models Identification from the Process Reaction Curve. Ciencia y Tecnología (Costa Rica) **24(2)**, 197–216 (2006). (in Spanish) Available at http://revista-ciencia-tecnologia.ucr.ac.cr/index.php/ciencia-tecnologia/article/view/35

# Appendix A
# MoReRT Controllers Design Demo Software

The use of the proposed Model-Reference Robust Tuning (MoReRT) design methodology, described in Chap. 4, to tune a two-degree-of-freedom (2DoF) proportional integral derivative (PID) controller requires of an optimization program, as outlined in Sect. 11.2.2. In order to facilitate the implementation of the MoReRT approach, a MATLAB® based software package has been developed. The provided routines just require the user to input the process information data and desired controller structure. The software will perform the required optimizations and show the closed-loop responses for the obtained controller.

In the following, a simple MoReRT software package implemented in MATLAB® is described. The user interface and software usage are explained by means of developing some design examples. This software can be obtained directly from the authors.

## A.1 Introduction

Considered the general closed-loop control system depicted in Fig. A.1 where the controlled process is given by

$$y(s) = P_u(s)u(s) + P_d(s)d(s), \tag{A.1}$$

and the controller output signal by the expression

$$u(s) = C_r(s)r(s) + C_y(s)y(s). \tag{A.2}$$

From (A.1) and (A.2) the closed-loop control system output is then

$$y(s) = \frac{C_r(s)P_u(s)}{1 + C_y(s)P_u(s)}r(s) + \frac{P_d(s)}{1 + C_y(s)P_u(s)}d(s), \tag{A.3}$$



V.M. Alfaro and R. Vilanova, *Model-Reference Robust Tuning of PID Controllers*,
Advances in Industrial Control, DOI 10.1007/978-3-319-28213-8

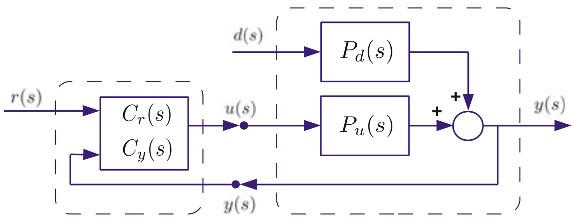



**Fig. A.1** General 2DoF closed-loop control system block diagram

that can be rewrite in compact form as

$$y(s) = M_{yr}(s)r(s) + M_{yd}(s)d(s). \tag{A.4}$$

If $P_d(s) = P_u(s)$ the disturbance $d(s)$ acts as an *input disturbance* $d_i(s)$, or load disturbance. In case that the disturbance input dynamics $P_d(s) = 1$ the disturbance $d(s)$ is an *output disturbance* $d_o(s)$. In general, if $P_d(s) \neq P_u(s)$ the path from the disturbance to the controlled variable $d \rightarrow y$ is different to the path from the controller output signal to the controlled variable $u \rightarrow y$.

The controlled process model $P_u(s)$ captures the controlled variable dynamics to a change in the controller output signal and the model $P_d(s)$ the corresponding dynamics to a change in the process disturbance signal. These two dynamics can be very different. Therefore, (A.3) allows to analyze different alternatives or control problems.

## A.2 Controlled Process Models and Control Algorithm

We consider here the general case of over-damped first- and second-order plus dead-time controlled process models, (F)SOPDT, and a controller with a 2DoF Standard PID control algorithm, $PI(D)_2$.

***Controlled Process Models***

The controlled process dynamics are given by following models:

$$P_u(s) = \frac{K_{pu}\mathrm{e}^{-L_{pu}s}}{(T_{pu}s+1)(a_{pu}T_{pu}s+1)}, \tag{A.5}$$

$$P_d(s) = \frac{K_{pd}\mathrm{e}^{-L_{pd}s}}{(T_{pd}s+1)(a_{pd}T_{pd}s+1)}, \tag{A.6}$$

where their parameters are $\theta_{pu} = \{K_{pu} \neq 0,\ T_{pu} > 0,\ 0 \leq a_{pu} \leq 1,\ L_{pu} \geq 0\}$ and $\theta_{pd} = \{K_{pd} \neq 0,\ T_{pd} \geq 0,\ 0 \leq a_{pd} \leq 1,\ L_{pd} \geq 0\}$. Here the usual default parameter values for the derivative part have been taken.

Gains of the controlled process models $P_u(s)$ and $P_d(s)$ can be any number different from zero. If $K_{pu} > 0$ the controller Action must be set to Reverse (+). In case that $K_{pu} < 0$ a Direct (−) Action is required for the controller.

### *Control Algorithm*

The 2DoF Standard PID control algorithm given by the equation

$$u(t) = K_p \left\{ \beta r(t) - y(t) + \frac{1}{T_i} \int_0^t [r(\xi) - y(\xi)] \mathrm{d}\xi + \frac{\mathrm{d}[\gamma r(t) - y(t)]}{\mathrm{d}t} \right\}, \quad \text{(A.7)}$$

or

$$u(s) = K_p \left\{ \beta r(s) - y(s) + \frac{1}{T_i s}[r(s) - y(s)] + \frac{T_d s}{\alpha T_d s + 1}[\gamma r(s) - y(s)] \right\} \quad \text{(A.8)}$$

is selected for the controller. Then, the controller parameters to tune are $\theta_c = \left\{ K_p > 0,\ T_i > 0,\ T_d \geq 0,\ \beta \geq 0,\ \alpha = 0.1,\ \gamma = 0 \right\}$.

The set-point and feedback controllers are given by the transfer functions:

$$C_r(s) = K_p \left( \beta + \frac{1}{T_i s} \right), \quad \text{(A.9)}$$

$$C_y(s) = K_p \left( 1 + \frac{1}{T_i s} + \frac{T_d s}{0.1 T_d s + 1} \right). \quad \text{(A.10)}$$

### *Normalized Controller Process Models and Controllers*

As described in Chap. 5 it is convenient to normalize the controlled process models and the controller. Then, using $P_u(s)$ parameters $K_{pu}$ and $T_{pu}$ and the transformation $\hat{s} = T_{pu} s$, the normalized versions of A.5, A.6, A.9, and A.10 are:

$$\hat{P}_u(\hat{s}) = \frac{\mathrm{e}^{-\tau_{Lpu} \hat{s}}}{(\hat{s} + 1)(a_{pu} \hat{s} + 1)}, \quad \text{(A.11)}$$

$$\hat{P}_d(\hat{s}) = \frac{\kappa_{pd} \mathrm{e}^{-\tau_{Lpd} \hat{s}}}{(\tau_{pd} \hat{s} + 1)(a_{pd} \tau_{pd} \hat{s} + 1)}, \quad \text{(A.12)}$$

$$\hat{C}_r(\hat{s}) = \kappa_p \left( \beta + \frac{1}{\tau_i \hat{s}} \right), \quad \text{(A.13)}$$

$$\hat{C}_y(\hat{s}) = \kappa_p \left( 1 + \frac{1}{\tau_i \hat{s}} + \frac{\tau_d \hat{s}}{0.1 \tau_d \hat{s} + 1} \right), \quad \text{(A.14)}$$

where

$$\tau_{Lpu} \doteq \frac{L_{pu}}{T_{pu}},$$
$$\kappa_{pd} \doteq \frac{K_{pd}}{K_{pu}}, \ \tau_{pd} \doteq \frac{T_{pd}}{T_{pu}}, \ \tau_{Lpd} \doteq \frac{L_{pd}}{T_{pu}}, \tag{A.15}$$
$$\kappa_p \doteq K_{pu} K_p, \ \tau_i \doteq \frac{T_i}{T_{pu}}, \ \tau_d \doteq \frac{T_d}{T_{pu}},$$

are the new normalized (dimensionless) parameters.

## A.3 Closed-Loop Transfer Functions Targets and Cost Functionals

Following the general procedure described in Chap. 4 and in Sect. 10.2 for the particular case when $P_d(s) \neq P_u(s)$, the target closed-loop servo-control and regulatory control transfer functions for (A.4) are selected as:

$$M_{yr}^t(\hat{s}) \doteq \frac{(\tau_c \hat{s} + 1)\, \mathrm{e}^{-\tau_{Lpu}\hat{s}}}{(\tau_c^2 \hat{s}^2 + 2\zeta\tau_c \hat{s} + 1)(a_{pu}\tau_c \hat{s} + 1)}, \tag{A.16}$$

$$M_{yd}^t(\hat{s}) \doteq \frac{(\hat{s} + 1)(a_{pu}\hat{s} + 1)}{(\tau_{pd}\hat{s} + 1)(a_{pd}\tau_{pd}\hat{s} + 1)} \frac{(\kappa_{pd}/\kappa_{pu})(\tau_i/\kappa_p)\, \hat{s}\, \mathrm{e}^{-\tau_{pd}\hat{s}}}{(\tau_c^2 \hat{s}^2 + 2\zeta\tau_c \hat{s} + 1)(a_{pu}\tau_c \hat{s} + 1)}, \tag{A.17}$$

where $\theta_d = \{\zeta, \tau_c\}$ are the design parameters. According to the pursued studies and analysis, presented in the corresponding chapters, there are recommended fixed values for $\zeta$ in order to guarantee a good compromise between the performance and control signal usage. On the other hand, $\tau_c$ should be adjusted in order to provide the fastest response for the desired robustness.

## A.4 MoreRT Controllers Design Software Implementation

As shown in Fig. 11.1 to implement the proposed MoReRT design procedure a minimum of five routines are required: a *main program* for data entry, a *design function* to call the *optimization function* and for iteration control, a *cost function*, and a *simulation and plotting function*.

The MoReRT cost function optimization is done using MATLAB `fminsearch` function which uses the Nelder-Mead simplex method.

The control system simulation can be made directly in MATLAB or using a Simulink block diagram.

***Data Entry***

The user must provide the following information:

- The controlled process models $P_u(s)$ and $P_d(s)$ parameters: $\theta_{pu}$ and $\theta_{pd}$, respectively.
- The controller initial normalized parameters: $\theta_c^0$ (i.e. $\kappa_p^0 = 1, \tau_i^o = 1, \tau_d^0 = 0.25, \beta = 1$).
- The design parameters $\theta_d = \{\zeta, \tau_c\}$. For nearly non-oscillating responses $\zeta = 1$ must be used. For under damped responses it is recommended to use $\zeta = 0.80$ for the PI and $\zeta = 0.7$ for the PID. The closed-loop relative speed $\tau_c$ can be a single value (i.e. $\tau_c = 1$), a set of discrete values (i.e. $\tau_c = 0.8,\ 1.0,\ 1.05$), or a sequence of values (i.e. $\tau_c = 0.8 : 0.1 : 1.5$).
- The simulation time control (total simulation time and the discretization (sampling) time). The simulation span must cover the time required for the target servo-control and the regulatory control response to reach to a new steady-state operation. The discretization time selection impacts the total CPU time required for the cost function optimization process but also the numerical solution accuracy and stability.

***Cost Function***

The cost function evaluation include following steps:

- Regulatory control $M_{yd}(s)$ step response, $y_{yd}^m(t)$.
- Regulatory control target $M_{yd}^t(s)$ step response, $y_{yd}^t(t)$.
- Regulatory control cost functional, $J_d = \int [y_{yd}^m(t) - y_{yd}^t(t)]^2 \mathrm{d}t$.
- Servo-control $M_{yr}(s)$ step response, $y_{yr}^m(t)$.
- Servo-control target $M_{yr}^t(s)$ step response, $y_{yr}^t(t)$.
- Servo-control cost functional, $J_r = \int [y_{yr}^m(t) - y_{yr}^t(t)]^2 \mathrm{d}t$.
- Total cost functional evaluation, $J_T = J_d + J_r$.

***Controller Parameters***

For each $\tau_c$ given, the optimization routine prints the obtained controller normalized parameters $\hat{\theta}_c$. With these, the corresponding control system robustness $M_S$ is evaluated.

For the last design parameter $\tau_c$ analyzed, the controller MoReRT tuning parameters (not the normalized ones) $K_P = \kappa_p/K_{pu}$, $T_i = \tau_i T_{pu}$, $T_d = \tau_d T_{pu}$, $\beta$, $\alpha = 0.1$, and $\gamma = 0$ are returned. Finally, the control system responses to a $\Delta r$ set-point change followed by a $\Delta d$ disturbance change are shown with its robustness plot.

The MoReRT designed controller parameters $\{K_p,\ T_i,\ T_d,\ \beta\}$ are available at the MATLAB Workspace as the variables `Kpo`, `Tio`, `Tdo`, and `bo`, respectively.

***Demo Software Files***

- `MoReRTcPID2pSOPuPd.p`: main program, data entry and program execution control.


**Fig. A.2** Program main menu user interface

- `mrtfpcPID2pSOPuPd.p`: design parameter $\tau_c$ iteration control, optimization function call.
- `mrtfccPID2pSOPuPd.p`: cost function evaluation.
- `mrtfscPID2pSOPuPd.p`: control systems simulation, step responses and robustness plot.
- `mrtfgNyquist.p`: Nyquist with $M_S$ circles plot.
- MATLAB functions: `bode`, `disp`, `fminsearch`, `lsim`, `nyquist`, `plot`, `step`.

***Program Main Menu***

The MoReRT demo software user interface (`[MAIN MENU]`) is shown in Fig. A.2.

Using the `MAIN MENU` push buttons the user can input the problem data (design parameters, controlled process (F)SOPDT models parameters, PI(D) controller initial parameters, and simulation control), run the controller tuning routine, and/or exit the control system design process.

***Input Data Error Detection***

The MoReRT demo software implements a simple verification of the input data provided by the user to prevent an eventual program malfunction or hanging. Some of the error messages are shown in Fig. A.3.

## A.5 MoReRT Controllers Design Demo Software Usage

For use of the MoReRT controllers design demo software consider the following example

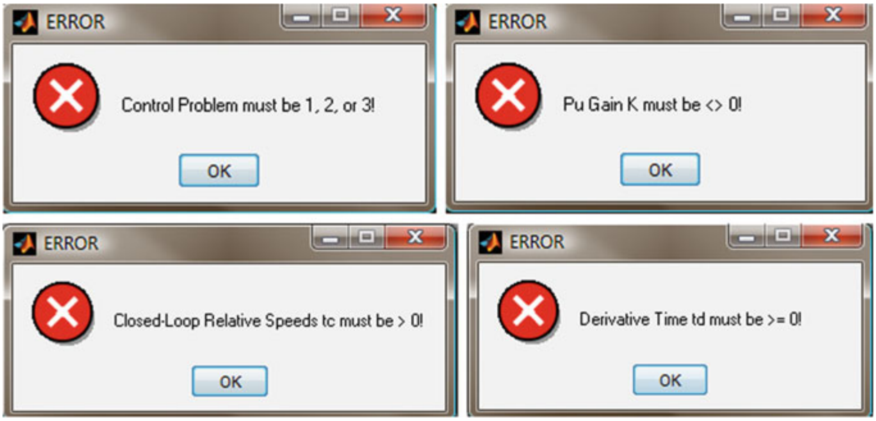



**Fig. A.3** Input data error messages

- Controlled process models:

$$P_u(s) = \frac{2e^{-2.0s}}{(5.0s+1)(2.5s+1)},$$
$$P_d(s) = P_u(s).$$

- Controller: 2DoF PI.
- Design criteria: $M_S^t \approx 1.80$, $\zeta = 0.80$, $\tau_c = 1.0$ (only one initial test).
- Controller initial normalized parameters: $\kappa_p^0 = 1.0$, $\tau_i^0 = 1.0$, $\tau_d^0 = 0$, $\beta = 1.0$.
- Simulation control: $t_u = 100.0$ $\delta t = 0.02$ (for simulations during optimization), $t_{us} = 200.0$ $\delta t_s = 0.2$, $\Delta r = 10\,\%$, $\Delta d = 5\,\%$ (for final control system simulation).

***Design for Input Disturbance***

For the first example, the disturbance transfer function is selected as $P_d(s) = P_u(s)$. Then, we are considering an *input disturbance* $d_i(s)$.

The Execution of the MoReRT Design Demo Software maim program is started writing the main file name at MATLAB Command Window:
`»MoReRTcPID2pSOPuPd [Enter]`
After that, the program Main Menu shown in Fig. A.2 is displayed.

**Data Input**

User must provide the control problem data.

The problem data input windows are selected from the `MAIN MENU`:

- `[DESIGN Parameters]`, opens the window shown in Fig. A.4.
- `[CONTROLLED PROCESSES Parameters]`, opens the window shown in Fig. A.5.

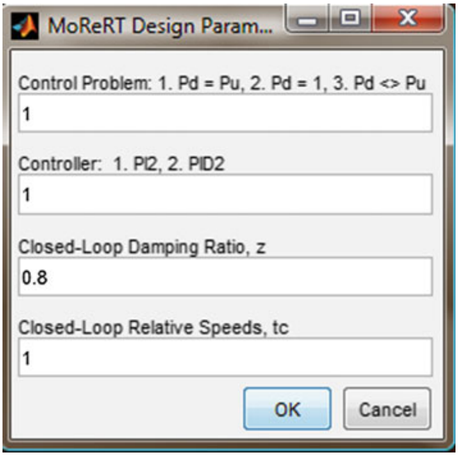



**Fig. A.4** Design parameters input window

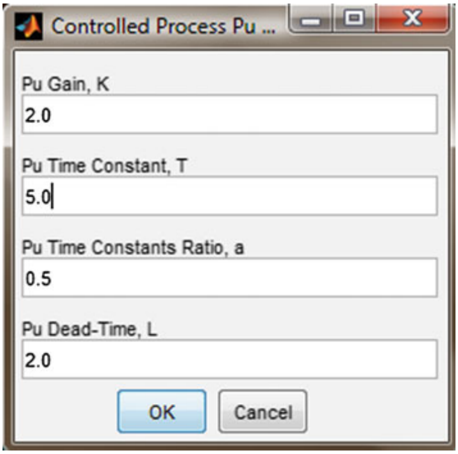



**Fig. A.5** Controlled process $P_u(s)$ parameters input window

- `[CONTROLLER Parameters]`, opens the window shown in Fig. A.6.
- `[SIMULATION Control]`, opens the window shown in Fig. A.7.

## Program Output

The program output is:

```
== MoReRT Controllers Design (cPID2/pSOPDT) ==
==============================================
- Controlled process models parameters -
Pu(s):
Kpu = 2
Tpu = 5
apu = 0.5
Lpu = 2
```

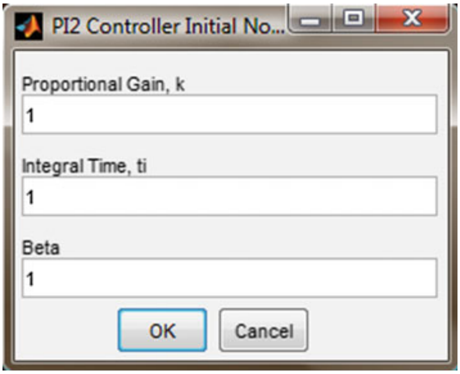



**Fig. A.6** PI Controller initial normalized parameters input window

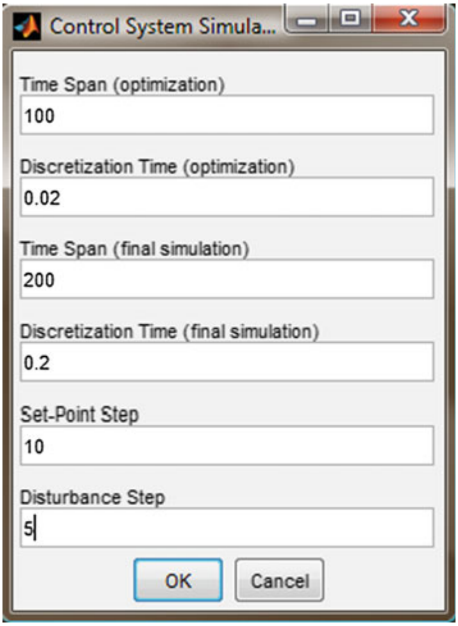



**Fig. A.7** Control system simulation control input window

```
Pd(s) = Pu(s)

- Design parameters -
controller: PI2
z   = 0.8
tco = 1

- MoReRT controller normalized parameters -
tc = 1
```

```
kp = 0.81863
ti = 1.2085
td = 0
ba = 0.90726
Ms = 1.573

== MoReRT PID2 controller parameters ==
Action = reverse (+)
Kp     = 0.40931
Ti     = 6.0425
Td     = 0
alpha  = 0.1
beta   = 0.90726
gamma  = 0
Ms     = 1.573
»
```

The robustness of the control system $M_S = 1.57$ is higher that the design requirement. Then, we can increase the control system speed and use `tcv = 0.95:-0.05:0.85`, to obtain three controllers. Therefore looking for a better performance on the basis of the desired robustness.

Selecting from the `MAIN MENU` the `[DESIGN Parameters]` input window the new closed-loop relative speed design parameters are typed as shown in Fig. A.8

### Design Progress Bar

During the design process its progress is indicated by a length changing red bar as shown in Fig. A.9. The design process can be interrupted with the `[Cancel]` push button. It will be stopped after that the optimization with the next $\tau_c$ design parameter is finished.

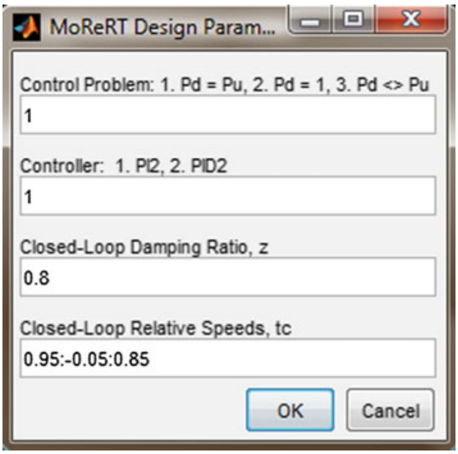



**Fig. A.8** Change of the closed-loop relative speeds

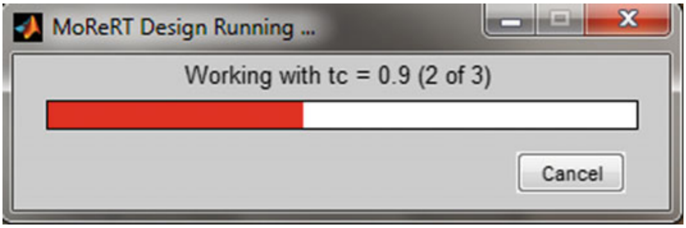



**Fig. A.9** Controller design progress bar

## New Results

The program output is now:

```
== MoReRT Controllers Design (cPID2/pSOPDT) ==
==============================================
...

- MoReRT controller normalized parameters -
tc = 0.95
kp = 0.90509
ti = 1.2411
td = 0
ba = 0.85694
Ms = 1.6367

tc = 0.9
kp = 0.99883
ti = 1.2708
td = 0
ba = 0.81243
Ms = 1.712

tc = 0.85
kp = 1.1
ti = 1.2976
td = 0
ba = 0.77317
Ms = 1.8013

== MoReRT PID2 controller parameters ==
Action = reverse (+)
Kp     = 0.55
Ti     = 6.4878
Td     = 0
alpha  = 0.1
```

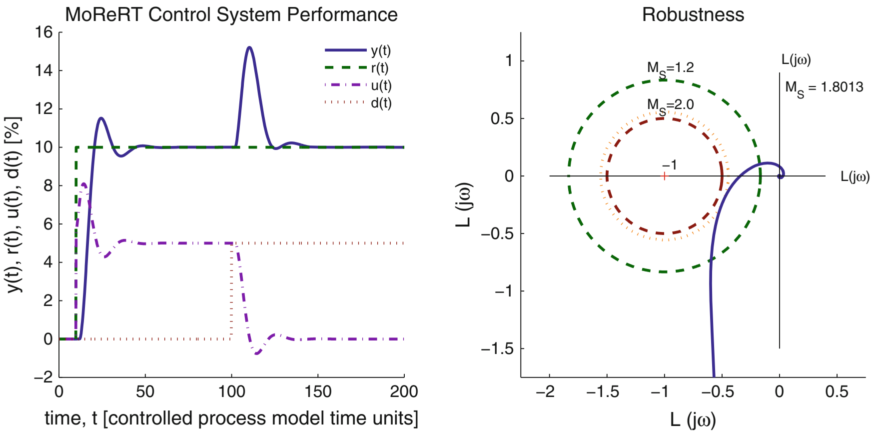



**Fig. A.10** Demo example ($d_i$)—System responses and robustness ($\tau_c = 0.85$)

```
beta    = 0.77317
gamma   = 0
Ms       = 1.8013
»
```

For $\tau_c = 0.85$ the control system robustness is $M_S = 1.80$. Then, we select a $PI_2$ controller with parameters $K_p = 0.55$, $T_i = 6.49$, and $\beta = 0.77$. The resulting control systems responses and robustness are shown in Fig. A.10.

***Design for Output Disturbance***

To design the MoReRT controller considering an output disturbance $d_o$ the transfer function $P_d(s) = 1$. Then, `Kpd = 1`, `Tpd = 0`, `apd = 0`, and `Lpd = 0` are automatically set.

Selecting at the `[DESIGN Parameters]` input window the control problem 2 (`Pd = 1`) and using `z = 0.8` and `tcv = 0.85` as shown in Fig. A.11, we have

```
== MoReRT Controllers Design (cPID2/pSOPDT) ==
==============================================
...
- MoReRT controller normalized parameters -
tc = 0.85
kp = 1.1936
ti = 1.2878
td = 0
ba = 0.67589
Ms = 1.9124

== MoReRT PID2 controller parameters ==
Action = reverse (+)
```

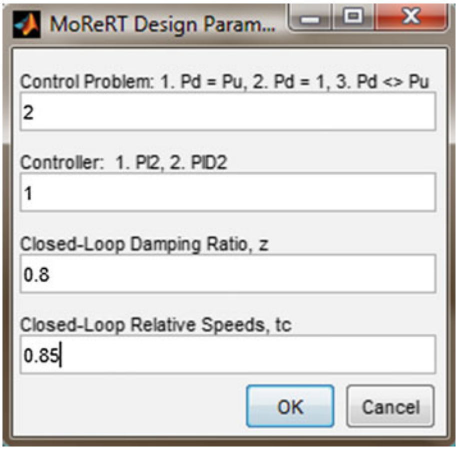



**Fig. A.11** Output disturbance selection

```
Kp      = 0.59682
Ti      = 6.439
Td      = 0
alpha   = 0.1
beta    = 0.67589
gamma   = 0
Ms      = 1.9124
»
```

Using the same closed-loop system relative speed design parameter, the resulting control system for an output disturbance is more oscillating, with higher peak error, and less robust than the one designed by considering an input disturbance. The new control systems responses and robustness are shown in Fig. A.12.

To compare, at the same design robustness level, the control system obtained considering an output disturbance with the one obtained earlier considering an input disturbance the control system relative speed must be decreased.

Adjusting `tc` we finally obtain:

```
== MoReRT Controllers Design (cPID2/pSOPDT) ==
==============================================
- Controlled process models parameters -
Pu(s):
Kpu = 2
Tpu = 5
apu = 0.5
Lpu = 2

Pd(s) = 1
```

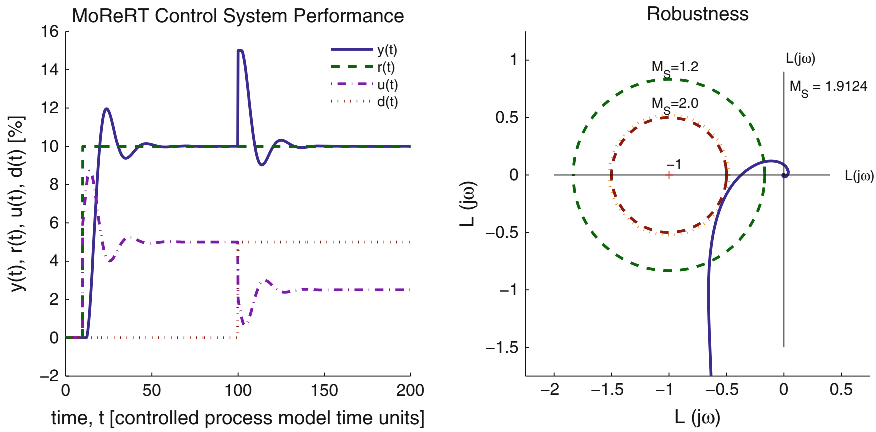



**Fig. A.12** Demo example ($d_o$)—System responses and robustness ($\tau_c = 0.85$)

```
- Design parameters -
controller: PI2
z    = 0.8
tco = 0.895

- MoReRT controller normalized parameters -
tc = 0.895
kp = 1.0841
ti = 1.2707
td = 0
ba = 0.72126
Ms = 1.8004

== MoReRT PID2 controller parameters ==
Action = reverse (+)
Kp     = 0.54206
Ti     = 6.3537
Td     = 0
alpha  = 0.1
beta   = 0.72126
gamma  = 0
Ms     = 1.8004
»
```

The responses of the new controller are shown in Fig. A.13.

The parameters of the two $PI_2$ controllers are listed in Table A.1. These two controller parameters sets are different but produce control systems with the same robustness level.

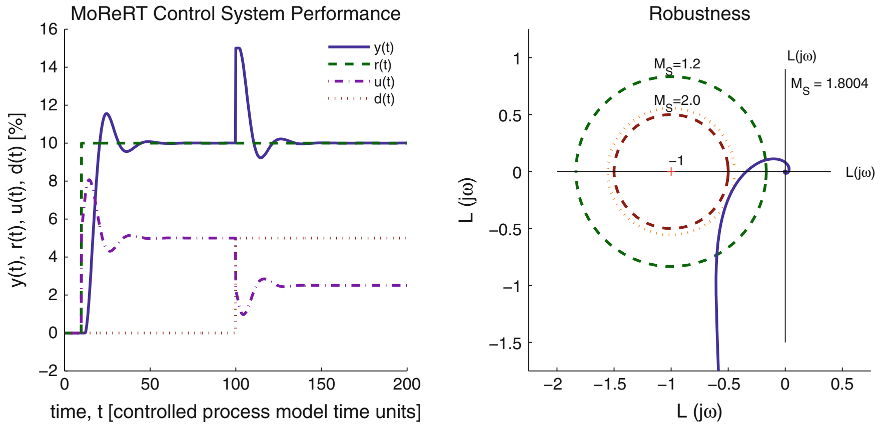



**Fig. A.13** Demo example ($d_o$)—New system responses and robustness ($\tau_c = 0.895$)

**Table A.1** MoReRT $PI_2$ parameters

| Disturbance | $\tau_c$ | $K_p$ | $T_i$ | $\beta$ | $M_S$ |
|---|---|---|---|---|---|
| $d_i$ | 0.850 | 0.550 | 6.488 | 0.773 | 1.801 |
| $d_o$ | 0.895 | 0.542 | 6.354 | 0.721 | 1.800 |

### *Design for a Slow Disturbance*

Consider now that the main disturbance dynamics has been identified and that it is slower than early assumed and with a negative and lower gain. It is given by the model

$$P_d(s) = \frac{-0.5e^{-4.0s}}{(7.5s+1)(5.625s+1)}.$$

The robustness requirement is increased to $M_S^t = 1.60$. The only data that needs to be changed is shown if Figs. A.14 (`[DESIGN Parameters]`) and A.15 (`[CONTROLLED PROCESSES Parameters]`).

The program output data is:

```
» MoReRT_cPID2pSOPuPd

== MoReRT Controllers Design (cPID2/pSOPDT) ==
==============================================
- Controlled process models parameters -
Pu(s):
Kpu = 2
Tpu = 5
apu = 0.5
Lpu = 2
```

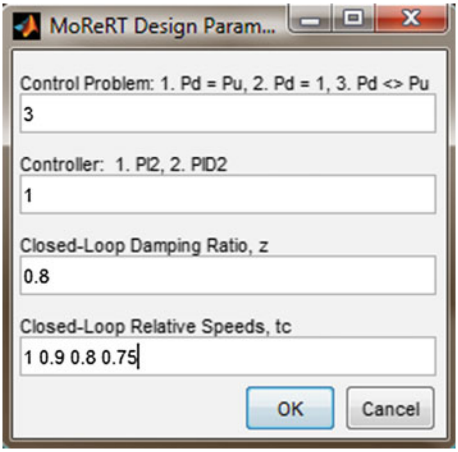



**Fig. A.14** New desing parameters

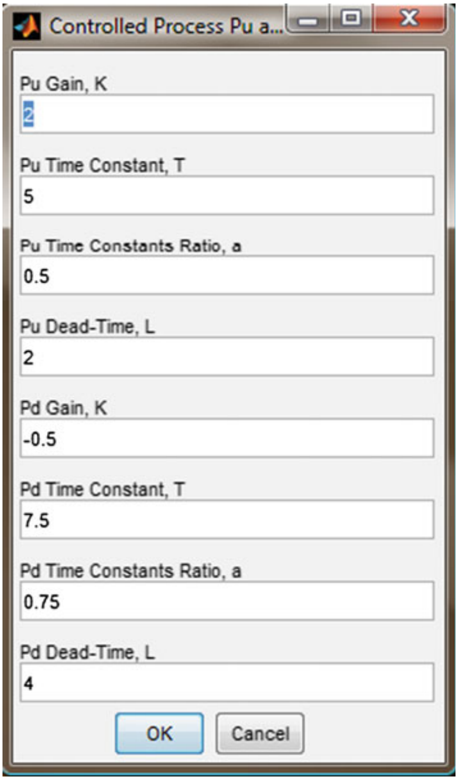



**Fig. A.15** Controlled process data $(P_d(s) \neq Pu(s))$

```
Pd(s):
Kpd = -0.5
Tpd = 7.5
apd = 0.75
Lpd = 4

- Design parameters -
controller: PI2
z   = 0.8
tco = 1

- MoReRT controller normalized parameters -
tc = 1
kp = 0.67112
ti = 1.2083
td = 0
ba = 1.2127
Ms = 1.447

tc = 0.9
kp = 0.78159
ti = 1.3145
td = 0
ba = 1.175
Ms = 1.4985

tc = 0.8
kp = 0.90764
ti = 1.4377
td = 0
ba = 1.1521
Ms = 1.5616

tc = 0.75
kp = 0.97763
ti = 1.5089
td = 0
ba = 1.1449
Ms = 1.5988

== MoReRT PID2 controller parameters ==
Action = reverse (+)
Kp     = 0.48882
Ti     = 7.5444
Td     = 0
```

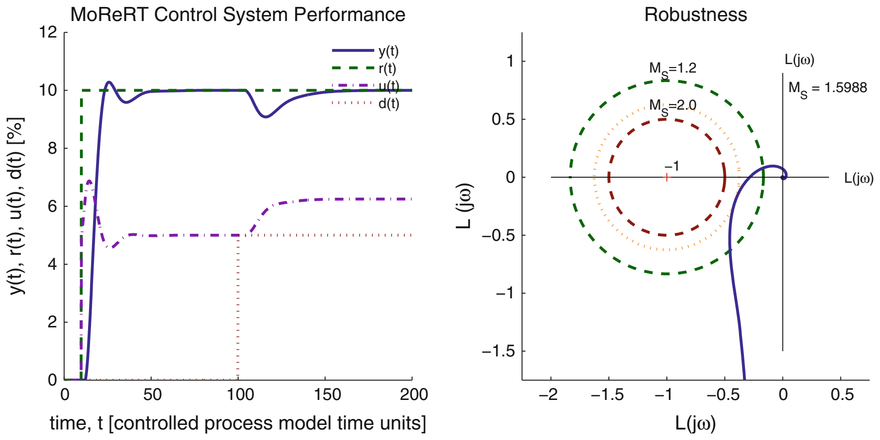



**Fig. A.16** Demo example $P_d(s) \neq P_u(s)$ (slow disturbance)—System responses and robustness

```
alpha   = 0.1
beta    = 1.1449
gamma   = 0
Ms      = 1.5988
»
```

The $PI_2$ control system response obtained with $\tau_c = 0.75$ has the required robustness $M_S = 1.60$. Its response is shown in Fig. A.16.

It is noted that in this case the controller proportional set-point weight needs a value $\beta > 1$ (`beta =1.145`).

### *MoReRT* $PID_2$ *Controller*

For comparison with the $PI_2$ controller $PID_2$ controllers are obtained using `z = 0.70` and `tcv = 0.65:0.05:0.75`.

The changes in the input data are shown in Figs. A.17 (`[DESIGN Parameters]`) and A.18 (`[CONTROLLER Parameters]`).

The PID design gives:

```
== MoReRT controllers design (cPID2/pSOPDT) ==
==============================================
...
- Design parameters -
controller: PID2
z   = 0.7
tco = 0.65
...

- MoReRT controller normalized parameters -
tc = 0.65
```

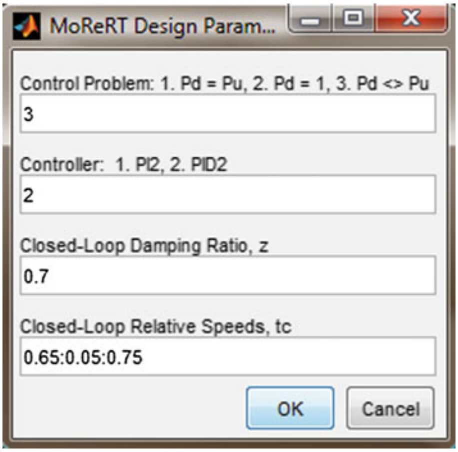



**Fig. A.17** New design parameters for $PID_2$

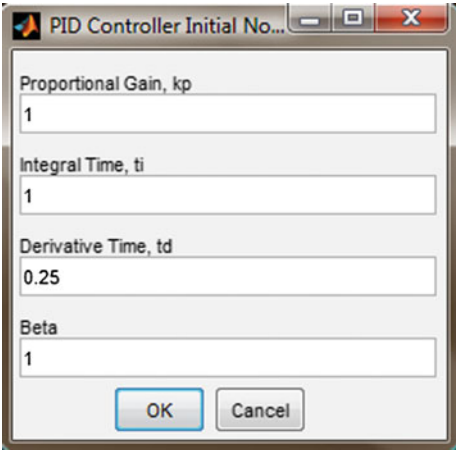



**Fig. A.18** PID controller normalized initial parameters

```
kp = 2.0063
ti = 1.3091
td = 0.35564
ba = 0.74978
Ms = 1.7647

tc = 0.7
kp = 1.8588
ti = 1.2729
td = 0.35816
ba = 0.73195
Ms = 1.6799
```

```
tc = 0.75
kp = 1.7116
ti = 1.2472
td = 0.36019
ba = 0.72474
Ms = 1.6019

== MoReRT PID2 controller parameters ==
Action = reverse (+)
Kp      = 0.85578
Ti      = 6.2359
Td      = 1.8011
alpha   = 0.1
beta    = 0.72474
gamma   = 0
Ms      = 1.6019
»
```

Parameters of the $PI_2$ and $PID_2$ controllers are listed in Table A.2. For the same robustness, the $PID_2$ control system provides a faster disturbance recovery and with

**Table A.2** MoReRT $PI_2$ and $PID_2$ parameters

| Controller | $K_p$ | $T_i$ | $T_d$ | $\beta$ | $M_S$ |
|---|---|---|---|---|---|
| $PI_2$ | 0.499 | 7.544 | 0 | 1.145 | 1.599 |
| $PID_2$ | 0.856 | 6.236 | 1.801 | 0.725 | 1.602 |

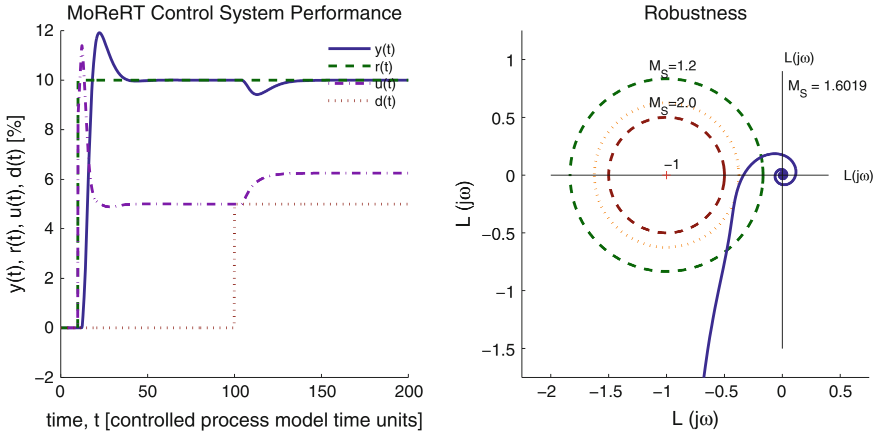



**Fig. A.19** Demo example $PID_2$ ($P_d(s) \neq P_u(s)$)—System responses and robustness lower maximum error than the $PI_2$. Its set-point response is also faster but with a higher overshoot, as seen in Fig. A.19.

Select [EXIT] from the MAIN MENU to close the program and return to MATLAB.

# Index



 
V.M. Alfaro and R. Vilanova, *Model-Reference Robust Tuning of PID Controllers*,
Advances in Industrial Control, DOI 10.1007/978-3-319-28213-8