Design of Low-Order Controllers using Optimization Techniques

Hast, Martin

2015

Document Version:
Publisher's PDF, also known as Version of record

Design of Low-Order Controllers using Optimization Techniques

Martin Hast

LUND
UNIVERSITY

Department of Automatic Control

PhD Thesis
ISRN LUTFD2/TFRT--1106--SE
ISBN 978-91-7623-285-9 (print)
ISBN 978-91-7623-286-6 (web)
ISSN 0280–5316

Department of Automatic Control
Lund University
Box 118
SE-221 00 LUND
Sweden

Printed in Sweden by Media-Tryck.
Lund 2015

Abstract

In many applications, especially in the process industry, low-level controllers are the workhorses of the automated production lines. The aim of this study has been to provide simple tuning procedures, either optimization-based methods or tuning rules, for design of low-order controllers.

The first part of this thesis deals with PID tuning. Design methods for both SISO and MIMO PID controllers based on convex optimization are presented. The methods consist of solving a nonconvex optimization problem by deriving convex approximations of the original problem and solving these iteratively until convergence. The algorithms are fast because of the convex approximations. The controllers obtained minimize low-frequency sensitivity subject to constraints that ensure robustness to process variations and limitations of control signal effort.

The second part of this thesis deals with tuning of feedforward controllers. Tuning rules that minimize the integrated-squared-error arising from measurable step disturbances are derived for a controller that can be interpreted as a filtered and possibly time-delayed PD controller. Using a controller structure that decouples the effects of the feedforward and feedback controllers, the controller is optimal both in open and closed loop settings. To improve the high-frequency noise behavior of the feedforward controller, it is proposed that the optimal controller is augmented with a second-order filter. Several aspects on the tuning of this filter are discussed.

For systems with PID controllers, the response to step changes in the reference can be improved by introducing set-point weighting. This can be interpreted as feedforward from the reference signal to the control signal. It is shown how these weights can be found by solving a convex optimization problem. Proportional set-point weight that minimizes the integrated-absolute-error was obtained for a batch of over 130 different processes. From these weights, simple tuning rules were derived and the performance was evaluated on all processes in the batch using five different feedback controller tuning methods. The proposed tuning rules could improve the performance by up to 45% with a modest increase in actuation.

Acknowledgments

I thank my main supervisor Tore Hägglund for letting me pursue the ideas that I found interesting, and for encouraging me in my endeavors to make something that is useful. I honestly think that I have had the pleasure of working with one of the best supervisors in the world, who always has taken time to help me with the issues that arose along the way, both research related and other problems.

I thank my co-supervisor Karl Johan Åström; your enthusiasm and spirit is a great source of inspiration. I have never met someone that is so eager to both generate interesting research problems and to solve them. I also thank Bo Bernhardsson who rushed in to my office one day and asked me if I wanted to work with him, Karl Johan and Stephen Boyd on an idea for PID design based on convex optimization. I think that turned out really nice. I thank Stephen Boyd for the work we have done on the PID parts of this thesis and the implementations of the design algorithms.

Apart for my supervisors, there are a number of persons at the department that has helped me immensely. First of all, I would like to acknowledge the department in whole, which provides an environment that is never dull, very knowledgeable and always extremely friendly. I think everyone at the department are equal contributors to this. Special thanks to the administrative and technical staff for always helping me, without hesitation, on questions concerning everything from travel claims and various computer issues (of which I usually had myself to blame), to advice on how to fix broken floor tiles and fastening shelves on walls. I thank the ”old” PhD students Erik, Kalle, Anders, P-O and Karl for welcoming me to the department, for encouraging me to apply for the PhD position and for the numerous lunch discussions filled with entertaining nonsense. I also thank Mikael for the interesting discussions and I wish you the best of luck in your endeavors to become a shoe snob.

I have either been extremely lucky or skilled when it comes to the choice of office mates. Olof, Josefin and Andreas, you have been the main reason for me to go to work every day. The days have always been full of jokes, crappy ”a-quarter-to-two” music, tomfoolery as well as research related discussions. Extra thanks to Olof who read the thesis and made many valuable suggestions on things that could be improved.

Eva Westin, the social hub of the department, has been a great support over the years. You have a fantastic ability to see to the needs of everyone around you.

Leif Andersson's expertise in everything related to computers in general and LaTeX in particular has been invaluable.

The PIC-LU people, and especially Anna, Ola and Olof G, deserves recognition for making the process-control oriented conferences and meetings interesting as well as fun and perhaps a bit bizarre.

Luckily, there are more people to thank than those at the department. Cecilia, you have been a fantastic support over the last two years and especially during the writing of this thesis. Thanks for proofreading this thesis on a subject far far from yours. I promise I will stop describing your proteins as "äggviteämne" when you stop using the term feedbackwards.

I also thank my family for always encouraging me to do what I thought was interesting and fun.

A large number of fiends have, more or less actively, helped me with my thesis. A few by forcing me to explain my research, some by reading the stuff I wrote, others by listening when I complained about the less fun parts of being a PhD student, and last but not least, all of you that did not give a damn about PhD studies at all.

Martin, Jill, Torvald, Klara, Hans, Posse, Josefine, Caro and Callum, you are a great bunch. The numerous dinners, whether its the fancy ones or the Friday taco ones, and the weekends spent in Blekinge have always been an absolute joy. A special thanks to Callum who gave me valuable suggestions on how to improve the language in this thesis. Thanks also to Rasmus, Maja, Hilda, Astrid and Andy for interesting discussions on German and Norwegian music, the Swedish Royal family, and how to properly flip over a 500 kg raft. Frida, Marcus, Josefin and Claes are much appreciated for their extraordinary taste in both music, and some of you, in beer. I thank Mia for all the fun we have had during and after the spex. Amanda and Daniel are acknowledged for always providing interesting discussions and brilliant, and subtle sarcasm. The same goes for Magnus but to claim that his sarcasm is subtle might be considered pushing it.

Arvid, Tore and Erik have helped me and given me advice on pretty much everything over the last 20 years. Thanks for not giving me much other choice than go to university to study Engineering Physics.

Finally, my thanks goes to Anna and Viktor; without your support I doubt there would have been a thesis at all.

Financial support

I acknowledge the following for financial support: the framework of the Process Industrial Centre at Lund University (PIC-LU), supported by the Swedish Foundation for Strategic Research (SSF), the Swedish Research Council through the LCCC Linnaeus Center and the Strategic Research Area ELLIIT.

Contents

1

Introduction

"Look inside, the day started right."

— Fu Manchu, *Mongoose*

The notion of control is so natural for human beings that we do not even notice it. Whether it is actions such as walking, maintaining the correct body temperature or driving a car along the road, the decisions taken by the brain are immensely complicated. External and internal signals interact in complicated ways so that without much thought, we can perform these mundane tasks. The human being in general, and its brain in particular, has an amazing capacity to continuously make decisions and to take action based on those decisions. The brain acts as the controller, which is the unit that is responsible for taking the decisions. The control systems in the human body has emerged over millions of years, an infinitesimal fraction of the time it has been used to control machines. The controllers used in modern industrial applications are today mostly implemented in computers and are nowhere near as complex as the brain. However, they share some inherent basic concepts such as the use of feedback and feedforward.

In a feedforward control system, the controller makes a decision based on some external input to the system and acts accordingly. Possible errors that arise due to the actions are not considered. This might be the only control option available, for instance in systems where it is not possible or very costly to measure the process output. In a feedback control system, decisions are instead based on an error between the controlled variable and its reference value. This makes it possible to reduce the effects of disturbances acting on the process, as well as making the system insensitive to variations in the process.

Many feedback problems can be visualized as in Figure 1.1. The controller should be designed so that it, based on measurements of the process output y and/or external signals, calculates a control signal u that drives the process output to the reference, or set-point, r. The controller should also attenuate load disturbances d, and not be sensitive to noise n. In the simplest, and perhaps most common applications, the process only has one input and one output. For that kind of system single-input single-output (SISO) controllers can be used. SISO controllers are far easier to tune and maintain than those with multiple-inputs and multiple-outputs (MIMO).

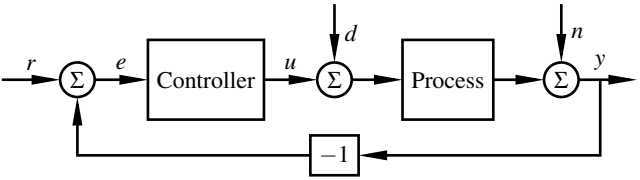

Figure 1.1 The classical control structure.

In modern factories, and especially in process industry, there can be several thousands of control loops. For these, controllers are for instance used to control pressure and flows in pipes or levels, temperatures and concentrations in tanks. Many loops are controlled by a proportional-integral-derivative (PID) controller. In its most basic form this controller has three parameters that need to be tuned so that the process output tracks the reference even when disturbances act on the system. The parameters should also be chosen so that the controller is robust to process variations and not sensitive to measurement noise.

Feedforward in combination with feedback can be used to improve the system's response to changes in set-point or measurable disturbances. In applications where PID is used for feedback control, it is often reasonable to use feedforward controllers of the same complexity. No standard controller exists for feedforward control as is the case with the PID for feedback control. Commonly, a filtered PD controller, possibly with an added time-delay is used. Such a controller has four parameters that need to be chosen so that the control system performs well.

In this thesis, methods for tuning robust PID controllers for both SISO and MIMO controllers are presented. Tuning rules for feedforward controllers that improve the performance of the control system are also presented. All these methods and tuning rules, either include optimization techniques or are derived from the solution of optimization problems. The methods are mainly intended for use in process industry where simple controllers, such as the PID are standard. The large number of control loops in a typical process plant are managed by a distributed control system (DCS) in which low-complexity controllers control a small subsection of the plant. The methods and tuning rules can, however, be used in other applications as well. All the methods assume that the plants are described by linear time-invariant models. The cost of deriving high-order and accurate models for every piece of the process is often too high compared to the profit of good control. Low-order models, for instance obtained from bump-tests or relay experiments, are simpler to derive and therefore cheaper. The tuning rules for feedforward controllers are motivated by their simplicity and once a simple model is obtained, the cost of finding good controller parameters is low.

The algorithms for design of PID controllers are more versatile and can be used to tune controllers for high-order models. The algorithms can also be extended to any controller that is linear in its parameters. On the other hand, they are based on

optimization techniques that are not readily available in current control systems.

There are both advantages and disadvantages related to the use of low-order controllers. They are limited by their simplicity in what they can achieve but on the other hand, their simplicity allows control engineers to manually retune the controller, if the need arises. Tuned properly, the performance of low-order controllers is often sufficient compared to more complex and costly alternatives, such as for instance model predictive controllers.

Contents and Contributions of the Thesis

This thesis consists of four chapters and five papers. This section briefly describes the chapters, the contributions of each paper and the specific contribution made by each author of the papers.

Chapter 2

In this chapter, a brief introduction to PID control is given. Various performance and robustness measures are introduced. A number of optimization methods for design of both PID and feedforward controllers are presented. The results presented in Papers I and II are summarized.

Chapter 3

An introduction to feedforward controller design is given in this chapter. The need for feedforward controller design using tuning rules is discussed as well as a number of controller structures. The chapter also presents a method based on convex optimization for the design of feedforward controllers. The method is used to obtain the optimal proportional set-point weight for PID controllers. The results from a large batch of processes are used to derive simple rules for the set-point weight. An overview of the work presented in Papers III-V is also provided.

Chapter 4

This chapter suggests possible future work that is related to both PID and feedforward controller design.

Paper I

Hast, M., K. J. Åström, B. Bernhardsson, and S. Boyd (2013). "PID design by convex-concave optimization". In: *Proceedings European Control Conference*, pp. 4460–4465.

In this paper a method for the design of robust SISO PID controllers is presented. The method consists of solving a sequence of convex optimization problems, called the convex-concave optimization procedure. The performance is measured

as the integrated error, and robustness constraints are specified as maximum values of the sensitivity functions. S. Boyd and K.J. Åström proposed the idea of using the convex-concave procedure to tune PID controllers. B. Bernhardsson extended the idea to processes with explicit uncertainties. M. Hast derived the constraints on the curvature of the Nyquist plot, implemented the algorithms and wrote the manuscript.

Paper II

Boyd, S., M. Hast, and K. J. Åström (2015). "MIMO PID tuning via iterated LMI restriction". Submitted to the *Internation Journal of Robust and Nonlinear Control*.

In this paper, a method for tuning robust MIMO PID controllers is presented. The PID parameters are obtained by formulating a nonconvex optimization problem that is solved by an iterative procedure. The idea stems from the work presented in Paper I. S. Boyd proposed that the robustness constraints could be expressed as linear matrix inequalities. All three authors worked on the formulation of the design problem and its solution. M. Hast implemented and tested the algorithm, formulated the examples, solved the related optimization problems, and performed the simulations. S. Boyd wrote the manuscript.

Paper III

Hast, M. and T. Hägglund (2012). "Design of optimal low-order feedforward controllers". In: *IFAC Conference on Advances in PID Control*. Brescia, Italy.

In this paper, a tuning rule for a low-order, time-delayed lead-lag, feedforward controller is presented. The tuning rule is based on first-order process models with time delay. The integrated squared error for a step load-disturbance is minimized. The optimal controllers are derived analytically, using a controller architecture having two degrees of freedom where the feedforward and feedback controllers are decoupled. T. Hägglund proposed the problem formulation. M. Hast formulated and solved the optimization problem, performed the simulations and wrote the manuscript.

Paper IV

Hast, M. and T. Hägglund (2014). "Low-order feedforward controllers: optimal performance and practical considerations". *Journal of Process Control* **24**:9, pp. 1462–1471.

In this paper, the tuning rules presented in Paper III are simplified and a number of practical issues are addressed. The optimal feedforward controller is augmented with a filter that provides high-frequency roll-off. Guidelines for choosing the filter parameter to decrease the peak of both the control signal and the Bode magnitude

are provided. A method for reducing the integrated squared error by adapting the time-delay in the filtered feedforward controller is also presented. The tuning rules and guidelines are tested on systems of high order. M. Hast refined the tuning rule, developed the filter guidelines, performed the simulations and wrote the manuscript. T. Hägglund provided valuable comments during the work.

Paper V

Hast, M. and T. Hägglund (2015). "Feedforward controller design using convex optimization and tuning rules for proportional set-point weighting". Submitted to *IET Control Theory & Applications*.

In this paper, a method for tuning feedforward controllers based on convex optimization is presented. Design of feedforward controllers for both measurable disturbances and the reference is considered. The proposed method is applied to a large batch of processes, resulting in a simple tuning rule for the set-point weight. M. Hast formulated and solved the problems, performed the simulations and wrote the manuscript. T. Hägglund provided valuable comments during the project.

2

PID Control

"Who controls the past controls the future; who controls the present controls the past."

— George Orwell, *1984*

The SISO PID controller is by far the most commonly used type of controller in the process industry [Desborough and Miller, 2002]. Reasons for this include that it is easy to understand, having only three parameters, and that in many applications it is sufficiently complex in order to obtain reasonable performance. Furthermore, the PID controller is well-known to most control technicians and engineers, and is available in most programmable-logic-controllers (PLCs) and DCS used in industry, thus, making it cost effective.

Many PID controllers are in fact only utilized as PI controllers and they are often poorly tuned [Bialkowski, 1993; Åström and Hägglund, 2001]. One reason that the derivative part is not used is that by increasing the number of controller parameters from two to three, the complexity of tuning the controller is increased; making hand-tuning of the controllers difficult. Furthermore, the derivative part generally increases the amount of actuation due to measurement noise. As a remedy for this, tuning rules could instead be used to find the controller parameters.

Most tuning rules are dependent on specific and often simple models of the process, while others rely on specific characteristics of the process. The perhaps most well-known tuning rules are those derived by Ziegler and Nichols [Ziegler and Nichols, 1942]. These rules rely on characterizing the process by two parameters, which then are used to tune the controller. The internal model control (IMC) framework [Rivera et al., 1986] has been used as a foundation for deriving tuning rules, see for example [Vilanova, 2008] or the the SIMC rules [Skogestad, 2003; Grimholt and Skogestad, 2012]. The lambda (λ)-method [Dahlin, 1968; Sell, 1995] that has been used extensively in the Swedish pulp and paper industry [Anonymous, 1997], is based on simple models and aims at tuning the PID controller so that the closed-loop system has a specific time constant. This can be seen as a special case of IMC. The AMIGO tuning rules [Åström and Hägglund, 2004] are derived by

solving an optimization problem that maximizes integral gain subject to robustness constraints, in a batch of over a hundred process models that are common in the process industry. The resulting controllers parameters have been parameterized in terms of first-order plus dead-time (FOTD) approximations of the plants and used as tuning rules.

The benefit of using tuning rules based on low-order process models is that those kinds of models are relatively easy to obtain, for instance by performing step-response tests. Simplicity can be important, especially in process industry where there are a large numbers of control loops. A vast amount of tuning rules exist based on different model structures and considerations. Many of them are covered in the impressive collection by [O'Dwyer, 2009]. The drawbacks of using a tuning rule are that both the performance and the robustness can be poor if the process' dynamics is not well described by the simple model. The performance and the robustness can be improved by tailoring the controller parameters to the process and utilizing all process knowledge. Optimization can be a useful means to achieve this. With fast and easy-to-use algorithms all process knowledge can be used to obtain a robust controller with good performance.

The PID controller is often used as a low-level controller in more sophisticated control algorithms such as Model Predictive Control (MPC). MPC is often used to generate reference signals for the PID controllers that in turn calculates the control signals, which are then sent to the actuators [Desborough and Miller, 2002].

The cost efficiency, the large number of commissioned controllers, and the fact that they often play an auxiliary role in more complex control schemes makes the PID research area still very active long after its first introduction. The book [Vilanova and Visioli, 2012] covers a wide variety of recent PID research. The many aspects of PID control, from modelling of processes, controller structure, tuning and implementation is presented in the book [Åström and Hägglund, 2006].

SISO PID controllers can be used to control MIMO processes by assigning one controller to each pair of inputs and outputs. Tuning of each controller is then performed in a sequential fashion that often requires iterative tuning runs. The problem could be facilitated if the interaction between the loops could be decoupled.

The MIMO PID controller is an extension of the SISO PID controller. It shares the basic controller structure but uses matrices instead of scalar parameters. Tuning controllers for processes with several inputs and outputs are therefore more challenging. For instance, a PID controller with two input and two outputs has 12 parameters that need to be chosen. Because of the large number of parameters, optimization based design is more suitable than tuning the controller by hand. The drawback of using a MIMO PID controller is that, except for benign processes, it is hard to retune if the need arises.

2.1 The PID Controller

As the abbreviation suggests, the PID controller has three parts; the Proportional, Integral and Derivative parts. The controller calculates the control signal as

$$u(t) = K\left(e(t) + \frac{1}{T_i}\int_0^t e(\tau)\mathrm{d}\tau + T_d\frac{\mathrm{d}e(t)}{\mathrm{d}t}\right), \tag{2.1}$$

where the three parameters are the gain K, the integral time T_i and the derivative time T_d. The control law (2.1) is the PID algorithm in the parallel form that will be used throughout this thesis. The controller can be represented in the Laplace domain with its transfer function

$$C(s) = K\left(1 + \frac{1}{sT_i} + sT_d\right) \tag{2.2}$$

or as

$$C(s) = k_p + \frac{k_i}{s} + k_d s. \tag{2.3}$$

The latter is preferred when performing optimization since the controller is linear in the parameters k_p, k_i and k_d. It can be noted that the formulation used in (2.3) allows for e.g., pure I-control by setting $k_p = k_d = 0$ whereas it is not possible using the former formulation. These formulations have some immediate flaws; they are nonproper and hence not realizable, and they have large gain for high frequencies, which makes them sensitive to noise. This is one of the reasons that the derivative action is not used in many commissioned controllers [Åström and Hägglund, 2001] and is commonly addressed by augmenting the controller with a filter.

The filtering of the PID controller can be performed in various ways. Augmenting the derivative part with a first-order filter with a small time constant is common, see [Åström and Hägglund, 2006] (p.73). A drawback with this approach is that the controller is not strictly proper and amplification of high-frequency noise can give rise to a jerky control signal. As a remedy for this, a second-order filter can be connected in series with the PID controller. The controller gain will then approach zero for high frequencies, a property called high-frequency roll-off. In [Larsson and Hägglund, 2011] it is concluded that PID controllers with second-order filters have near optimal performance compared to high-order controllers for low-order processes. By adding a filter, the number of parameters that needs to be tuned increases, thus making the controller harder to tune. An iterative procedure for tuning measurement filters is addressed in [Segovia et al., 2014]. The filter time-constant is chosen as a compromise between robustness, performance and control-signal activity. Tuning rules for PID controllers with first and second order filters for stable processes are presented in [Kristiansson and Lennartson, 2002].

2.2 Performance

One of the key issues when designing any controller is obtaining good performance. The controller should preferably follow changes in set-point and attenuate disturbances well. Both these requirements need to be quantified using a well defined performance measure. It is most commonly measured by an integral of some function of the norm of the control error. The perhaps most natural way to measure performance is the integrated-absolute-error (IAE) defined as

$$\text{IAE} = \int_0^\infty |e(t)|\,\mathrm{d}t \tag{2.4}$$

and evaluated for system representative external input. Both positive and negative errors will contribute to an increase in the IAE since it measures the absolute value of the error. Optimization problems that aim at minimizing the IAE are nonconvex in the controller parameters and can be time-consuming to solve. Most optimization-based tuning methods that minimize IAE find a local minimum to (2.4). PID tuning methods that minimize IAE have been presented in [Garpinger, 2009; Grimholt and Skogestad, 2015].

A closely related performance measure is the integrated error (IE) that is defined as

$$\text{IE} = \int_0^\infty e(t)\mathrm{d}t. \tag{2.5}$$

Note that the IE and the IAE are equal provided that the error does not change sign i.e., when the system is critically damped. For PID controllers, IE is easy to calculate and it is therefore a good substitute for the IAE measure provided that the system is well-damped. A pitfall with the IE measure is that it can be small when the error signal is oscillatory. For instance, an error signal that is sinusoidal will have an IE that is zero. It is therefore crucial to ensure that the error is well-damped when measuring performance by IE.

A third commonly used performance measure is the integrated-squared-error (ISE) defined as

$$\text{ISE} = \int_0^\infty e(t)^2\mathrm{d}t. \tag{2.6}$$

This measure emphasizes large errors since the error is squared. Optimization problems where ISE is minimized are nonconvex with the same disadvantages as those associated with minimization of IAE. However, minimization problems with the ISE measure can in some cases be solved analytically, for instance as in Chapter 3, which covers design of feedforward controllers. Quadratic performance measures are used extensively when designing controllers using state-feedback formulations such as e.g., Linear Quadratic Control and MPC.

In this thesis, the unit step will be used extensively as reference signal and disturbance to illustrate controller performance. It is natural to study the response of a

step since the set-point often is changed from one reference point to another. For disturbances, this is not always the case, but the step change is an established generic input used to study the impact of load disturbances.

2.3 Robustness

Performance is only one of the things that need to be taken into account when a controller is designed. It is also crucial that the designed controller works well even if the process model is inaccurate or if the dynamics of the physical process changes over time. The most vital aspect is ensuring that the controller will not render the closed-loop system unstable when the model is inaccurate. Robustness is a measure of the distance from instability that the closed-loop system will be with a given controller. Design of robust controllers has been an active area of research for a long time, and has given rise to a wide range of design methods, see [Zhou and C. Doyle, 1998].

Robustness for SISO processes

Amplitude and phase margins are classical measures of robustness. The amplitude margin specifies how much the gain of the process can increase before the system becomes unstable. The phase margin is a measure of how much the phase at a certain frequency can be allowed to vary in order for the system to remain stable. However, these measures do not capture what happens if the phase and gain change simultaneously. To better capture how much a system can change before it becomes unstable, the notion of maximum sensitivities is preferred. The sensitivity function

$$S(s) = \frac{1}{1 + P(s)C(s)}, \tag{2.7}$$

where $P(s)$ is a linear time-invariant model of the process, has several interpretations. It is the transfer function from measurement noise to system output, and the ratio between a system's open-loop and closed-loop responses. It thereby reflects both how noise will influence the system and how disturbances are attenuated. Furthermore, for a given frequency ω, $|S(i\omega)|$ is the inverse of the distance from a point on the Nyquist curve to the critical point -1. The maximum sensitivity of a system defined as

$$M_S = \max_{\omega} |S(i\omega)|, \tag{2.8}$$

is thus a measure of the smallest distance to the critical point, and thereby instability. It is common to design controllers so that M_S is in the range from 1.2 to 2.0 [Åström and Hägglund, 2006].

The transfer function from reference and load disturbance to the system output and control signal, respectively, is given by

$$T(s) = \frac{P(s)C(s)}{1 + P(s)C(s)}, \tag{2.9}$$

and is known as the complementary sensitivity function. It is desirable to design a controller so that $|T(i\omega)|$ is close to one up to a certain frequency in order to efficiently follow reference signals. It is also desirable to make $|T(i\omega)|$ small for high frequencies in order to reduce high-frequency components in the control signal since this can increase the wear of the actuator. The complementary sensitivity function can also be seen as a measure of how large additive changes in the process dynamics can be before the system becomes unstable. Assume that the process dynamics change from $P(s)$ to $P(s)+\Delta P(s)$ where $\Delta P(s)$ is stable. A condition for stability is [Åström and Hägglund, 2006]

$$\left|\frac{\Delta P(i\omega)}{P(i\omega)}\right| < \left|\frac{1}{T(i\omega)}\right|. \tag{2.10}$$

This expression implies that if $|T(i\omega_o)|$ is small, the controller will still be able to handle large changes in the process dynamics at the frequency ω_o. By designing a controller that makes $|T(i\omega)|$ small for high frequencies, the need for an accurate high-frequency model is less important. It is therefore reasonable to limit the maximum value of the magnitude of the complementary sensitivity function

$$M_T = \max_{\omega} |T(i\omega)| \tag{2.11}$$

by some upper bound, usually in the same range as for the maximum sensitivity.

Graphical interpretation of robustness in the Nyquist plot

Upper bounds on the sensitivity functions can be illustrated in the Nyquist plot. Both bounds can be represented by circles, with radii and centers that depend on M_S and M_T, see [Åström and Hägglund, 2006] for the derivations and Table 2.1 for the expressions. For $M_T = 1$, the bound on the complementary sensitivity becomes a vertical line. The circle interpretation will be used extensively in the method for design of SISO PID controllers as presented in Section 2.5 and in Paper I. Figure 2.1 shows an example of the circle representation in a Nyquist plot for $M_S = M_T = 1.4$.

Table 2.1 Radii and centers for the circles that represents the upper bounds on the sensitivity and the complementary sensitivity functions in the Nyquist plot.

Contour	Center	Radius
M_S	-1	$\frac{1}{M_S}$
M_T	$-\frac{M_T^2}{M_T^2-1}$	$\frac{M_T}{M_T^2-1}$

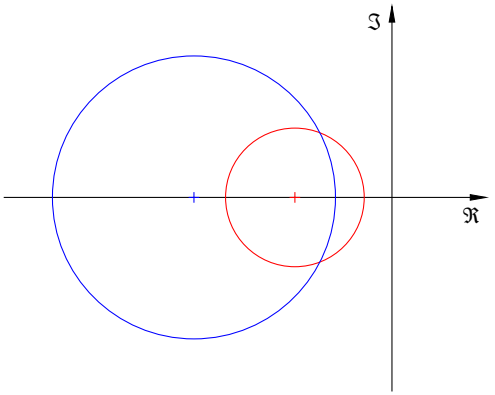

Figure 2.1 Maximum sensitivity $M_S = 1.4$ represented as a red circle with center in $(-1,0)$ in the complex plane. The blue circle represents the maximum complementary sensitivity $M_T = 1.4$ with center in $(-2.04,0)$

Robustness for MIMO processes

For MIMO processes, the sensitivity and the complimentary sensitivity are defined analogously to (2.7) and (2.9). Consider a process $P(s)$ with m inputs and p outputs and a controller $C(s)$ with p inputs and m outputs. The MIMO equivalent of the sensitivity and the complementary sensitivity functions, (2.7) and (2.9), are expressed as

$$S = (I + PC)^{-1}, \quad T = PC(I + PC)^{-1}, \tag{2.12}$$

respectively, and where the identity matrix is denoted by I. The MIMO equivalent of the maximum sensitivities are captured by the H_∞-norm of the two functions given by

$$M_S = \sup_{\omega \geq 0} \|S(i\omega)\|, \quad M_T = \sup_{\omega \geq 0} \|T(i\omega)\| \tag{2.13}$$

where $\|\cdot\|$ denotes the spectral norm, i.e., the maximum singular value. The MIMO definitions of sensitivity and complementary sensitivity are used to formulate the MIMO PID design problem as an optimization problem in Paper II.

2.4 Optimization

When designing controllers, both performance and robustness need to be taken into account. Formulating and solving an optimization problem is often a good way to proceed in these cases. All of the results presented in this thesis rely heavily on

optimization. A mathematical optimization problem can be formulated as

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & f_0(x) \\ \text{subject to} \quad & f_i(x) \leq b_i, \quad i = 1,\ldots,m \\ & h_j(x) = c_j \quad j = 1,\ldots,k, \end{aligned} \tag{2.14}$$

where the vector $x = (x_1,\ldots,x_n)$ is the optimization variable, and the function $f_0 : \mathbf{R}^n \to \mathbf{R}$ is the objective function that we want to minimize. The inequality constraints are expressed by the functions $f_i : \mathbf{R}^n \to \mathbf{R}, i = 1,\ldots,m$ that should be less than the bounds $b_1,\ldots,b_m$. The functions $h_j : \mathbf{R}^n \to \mathbf{R}, j = 1,\ldots,k$ are the equality constraints, which should be equal to constants c_j. A vector x^* is called a *global* optimum (minimum) if it satisfies

$$f_0(x^*) \leq f_0(x) \tag{2.15}$$

for all x that satisfies the constraints. The vector x^* is a *local* optimum if there exists a $\delta > 0$ such that (2.15) is satisfied for all x that satisfies the constraints in $\|x^* - x\| < \delta$. Solving (2.14) efficiently is in general difficult. However, classes of optimization problems exist, such as the convex problems, that can be efficiently solved for the optimal global solution.

It should be mentioned that optimal solutions are not necessarily good solutions. Care must be taken when formulating an optimization problem so that it captures the most reasonable aspects.

Convex Optimization

Mathematical optimization problems can be divided into a large number of classes. One important class is the convex optimization problems. An optimization problem posed as (2.14) is convex if the equality constraints h_j are affine in x and all the functions $f_0,\ldots,f_m$ are convex, i.e., they satisfy

$$f_i(\theta x + (1-\theta)y) \leq \theta f_i(x) + (1-\theta) f_i(y) \tag{2.16}$$

for all x, y that belongs to the convex domain of f_i, and $0 \leq \theta \leq 1$. A function $h(x)$ is concave if $-h(x)$ is convex. Many optimization problems encountered in engineering can be formulated as convex optimization problems. Examples include linear, quadratic, least-squares, second-order cone and semidefinite programs, and are used in a wide range of applications.

The theory of convex functions and optimization are well understood [Rockafellar, 1970; Boyd and Vandenberghe, 2004] and there are several benefits of optimization problems that are convex. A fundamental property of a convex optimization problem is that any local optimum is also a global optimum. This implies that it is sufficient to find a local optimum, which is simpler.

A number of easy-to-use tools for formulating convex optimization are available, for instance CVXOPT [M. S. Andersen et al., 2013; M. Andersen et al., 2011]

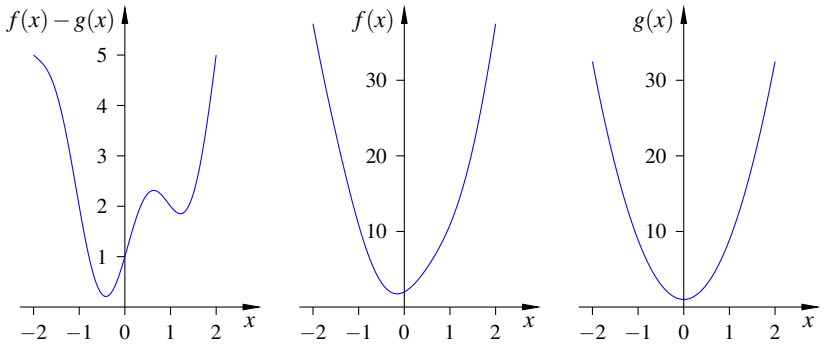

Figure 2.2 Illustration of a convex-concave function $f(x) - g(x)$ composed as a difference between the two convex functions $f(x)$ and $g(x)$.

for Python, and YALMIP [Lofberg, 2004] for MATLAB [MATLAB, 2012]. All convex optimization problems in this thesis are formulated using CVX [Research, 2012; Grant and Boyd, 2008] and solved with the SDPT3 solver [Toh et al., 1999; Tütüncü et al., 2003].

Convex optimization is used as a foundation for the design methods presented in Paper I and II. The design of feedforward controllers presented in Paper V is formulated as pure convex optimization problems.

The convex-concave procedure

The problem of finding the PID parameters so that the controller has good performance and robustness can unfortunately not be formulated as a convex optimization problem. Nevertheless, it can be formulated as a *difference of convex program* (DCP), which can be formulated as

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & f_0(x) - g_0(x) \\ \text{subject to} \quad & f_i(x) - g_i(x) \leq 0, \quad i = 1, \ldots, m. \end{aligned} \tag{2.17}$$

where the functions f_i, g_i for $i = 0 \ldots m$ are convex functions. From (2.17) it can be seen that convex optimization problems are special cases of DCPs where the functions g_i, $i = 0 \ldots m$ are affine. Any function with a bounded Hessian can be decomposed into a difference between two convex functions [Yuille and Rangarajan, 2003], although the decomposition is not unique.

An illustration of a function that is clearly nonconvex can be seen in the leftmost plot of Figure 2.2. It can, however, be decomposed into the difference of the functions displayed in the middle and rightmost plots.

The convex-concave procedure [Yuille and Rangarajan, 2003] is a heuristic method for finding a local minimum to (2.17). The method is an iterative procedure that in each iteration solves a convex optimization problem. The procedure in its

basic version requires a feasible initial point x_0. A convex formulation is obtained by linearizing the functions g_i around the current solution point x_k. The resulting convex problem is solved, the function is linearized around the obtained solution point and the procedure is repeated until convergence. A good summary of the procedure, for differentiable f_i and g_i, can be found in Algorithm 1.1 in [Lipp and Boyd, 2014] and it is restated here in Algorithm 1. All the iterates x_k will be feasi-

Algorithm 1: Convex-concave procedure.

given an initial feasible point x_0.
$k := 0$.
repeat

1. Form $\hat{g}_i(x;x_k) := g_i(x_k) + \nabla g_i(x_k)^T (x - x_k)$ for $i = 0, \ldots m$.
2. Set the value of x_{k+1} to a solution of the convex problem

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & f_0(x) - \hat{g}_0(x;x_k) \\ \text{subject to} \quad & f_i(x) - \hat{g}_i(x;x_k) \leq 0, \quad i = 1, \ldots, m. \end{aligned}$$

3. $k := k + 1$.

until *stopping criterion is satisfied.*

ble and the sequence of objective values will be nonincreasing, and converge [Lipp and Boyd, 2014]. If not initialized at a local maximum the procedure will converge to a local minimum or saddle point. The final point found can depend on the given initial point. The time it takes to solve the PID design problems encountered in this thesis are a few seconds for the SISO design, and a few minutes for the MIMO design, and it is therefore possible to test a number of initial parameters and solve the corresponding problems, taking the best of these parameters as the solution.

The stopping criterion

$$(f_0(x_k) - g_0(x_k)) - (f_0(x_{k+1}) - g_0(x_{k+1})) \leq \delta \tag{2.18}$$

with δ chosen as a small positive number is used in the design of PID controllers. The procedure is thus stopped when when there is little progress in the last iteration.

The convex-concave procedure has some advantages compared to similar algorithms. In the convexification step, all information from the convex parts is retained. This can be compared to sequential quadratic programming [Boggs and Tolle, 1995] where the original program, in each iteration, is approximated as a quadratic optimization problem. Furthermore, using the convex-concave procedure there is no need to include trust regions where it is ensured that the approximation is valid [Lipp and Boyd, 2014].

Several extensions to the convex-concave procedure such as; elimination of a feasible initial point, augmentation of a line search, and vector inequalities, can be found in [Lipp and Boyd, 2014].

2.5 Tuning of SISO PID Controllers

In Paper I, a method that utilizes the convex-concave procedure for design of robust PID controllers, which minimizes IE subject to robustness constraints, is presented. The objective is to find the controller parameters $x = [k_p\ k_i\ k_d]^T$ so that the performance measure is minimized while the robustness constraints are satisfied.

Performance

It can be shown [Åström and Hägglund, 2006] that for a process, controlled with a stabilizing PID controller

$$\text{IE} = \frac{1}{k_i} \tag{2.19}$$

when a unit-step disturbance enters at the process input. Minimizing IE is, thus, equivalent to maximizing the integral gain. Provided that the system is well-damped this is in turn equivalent to minimizing IAE. From (2.19) it is easy to see that minimizing IE is equivalent to the optimization criterion

$$\text{maximize} \quad k_i. \tag{2.20}$$

The criterion is clearly linear in the optimization variables and it is therefore convex. From an optimization point-of-view IE is thus simpler to work with than for instance IAE. Furthermore, for stable processes the first term in a Taylor series expansion of $S(s)$ around zero is $S(s) \approx \frac{s}{P(0)k_i}$. Thus, maximization of integral gain can be interpreted as minimizing low-frequency sensitivity.

Robustness constraints

To ensure that the system is robust, constraints on the maximum sensitivity functions are imposed. As discussed in Section 2.3, the robustness constraints can be interpreted as the Nyquist curve being outside the circles defined in Table 2.1. The PID controller (2.3) can be written as

$$C(s,x) = \begin{bmatrix} 1 & \frac{1}{s} & s \end{bmatrix} x. \tag{2.21}$$

Let the open-loop transfer function be $L(s,x) = P(s)C(s,x) = \mathbf{L}(s)x$, which is linear in the controller parameters. The point $\mathbf{L}(i\omega)x$ on the Nyquist plot is outside a circle with center c and radius r if

$$r - |\mathbf{L}(i\omega) - c| \leq 0. \tag{2.22}$$

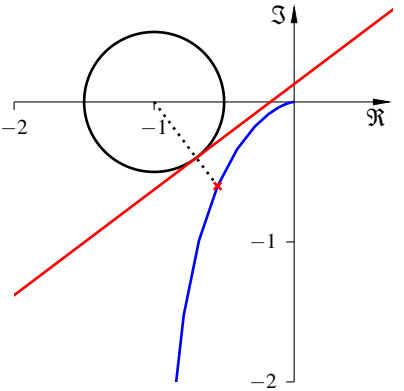

Figure 2.3 Nyquist plot of an open-loop system (blue) together with a black circle corresponding to a constraint on the maximum sensitivity function. For the point on the Nyquist plot that is marked with the red cross, the red line corresponds to the boundary of the convexified constraint in the next iteration. The point is constrained to lie in the half plane to the right of this line in the next iteration.

This is a convex-concave, or difference of convex, constraint with $f(x) = r$ and $g(x) = |\mathbf{L}(i\omega) - c|$. Linearizing $g(x)$ around a solution point x_k we obtain the convex constraint

$$f(x) - \hat{g}(x;x_k) = r - \Re\left(\frac{\overline{\mathbf{L}x_k - c}}{|\mathbf{L}x_k - c|}(\mathbf{L}x - c)\right) \leq 0 \tag{2.23}$$

where $\Re$ and the bar denote the real part and the conjugate transpose, respectively. Graphically, this can be interpreted as a half-plane in the complex plane that the point on the Nyquist plot is restricted to, see Figure 2.3.

Process uncertainties

The robustness constraints can be generalized to processes with a certain class of uncertainties. Consider an uncertain process model $\tilde{P}$ for which each point on the Nyquist plot is known to lie within a circle with a frequency dependent radius ρ, i.e.,

$$\tilde{P}(i\omega) = P(i\omega) + \Delta(i\omega), \quad |\Delta(i\omega)| \leq \rho(\omega). \tag{2.24}$$

The constraint

$$r - |PC + \Delta C - c| \leq 0 \tag{2.25}$$

specifies that the Nyquist plot for all processes in the uncertainty set is outside the circle. The constraint can be rewritten as

$$r - |L - c| + \rho|C| \leq 0. \tag{2.26}$$

This constraint is the difference of the two convex functions $r+\rho|C|$ and $|L-c|$ where the second function can be handled in the same way as the robustness constraints in the previous section.

Curvature constraints

Minimization of IE can give Nyquist curves with very small curvature. The step response of such a system is often oscillatory [Åström and Hägglund, 2006]. This could be avoided by using another cost function that penalizes oscillatory responses or by including constraints on the curvature as was done in [Panagopoulos et al., 2002]. Let the loop transfer function be decomposed by its real and imaginary part i.e., $L(i\omega) = x(\omega) + iy(\omega)$. The curvature of L is then given by

$$\kappa = \frac{\dot{x}\ddot{y} - \dot{y}\ddot{x}}{\left(\dot{x}^2 + \dot{y}^2\right)^{3/2}}, \tag{2.27}$$

where the dot and the double dot correspond to first and second derivative with respect to ω. Constraining the curvature by γ can then formulated as

$$x^T Q_p x - x^T Q_n x - \gamma |Zx|^2 \leq 0 \tag{2.28}$$

where Q_p and Q_n are positive semidefinite matrices and Z is a complex vector. By linearizing the concave part, a quadratic convex constraint is obtained.

Properties and extensions

The constraints are formulated for all frequencies, which leads to a semi-infinite optimization problem. To obtain a tractable optimization problem a frequency grid is introduced, and the constraints are imposed for each point in the grid. In each iteration, a convex optimization problem is solved, therfore a dense grid can be used. In the optimization problems in Paper I, 1000 grid points were used. It is vital to ensure that the grid captures the relevant frequency interval for the process.

The algorithm must be initialized with controller parameters which satisfy the constraints. If the process is asymptotically stable, initialization with $k_p = k_i = k_d = 0$ will satisfy the robustness constraints. For integrating and unstable processes, care must be taken to ensure that the initial controller is stabilizing and satisfies the constraints.

There are several benefits of using the convex-concave procedure for design of PID controllers. For instance, other convex constraints can easily be included in the optimization problem. In each iteration, a convex optimization problem formulated as a second-order cone program is solved. Interior point methods are known to solve such problems quickly [Boyd and Vandenberghe, 2004]. For the design problems presented in Paper I, no more than 11 iterations where needed, which on an ordinary desktop computer takes a few seconds. One could argue that this is in fact less time than it takes to find a tuning rule, appropriate for the process at hand, and calculating the controller parameter from that.

The ideas presented in Paper I can easily be extended. The algorithm is nonparametric, since it only needs the frequency response of the process and the controller. Fractional order PID controllers, see for instance [Padula and Visioli, 2011], on the form

$$C(s) = k_p + \frac{k_i}{s^\lambda} + k_d s^\mu \tag{2.29}$$

where λ and μ are fixed scalars can easily be handled since the controller is linear in the parameters. However, implementing such a controller in a DCS is not trivial and it is often done by approximating the fractional controller by an integer one, see [Monje et al., 2008] and the references therein.

The presented method for design of PID controllers can also be used for systems with a static nonlinearity in the feedback loop, by use of the circle criterion [Khalil and Grizzle, 1996]. The criterion implies that the closed-loop system is absolutely stable if the Nyquist curve is outside a circle, which is defined by bounds on the nonlinearity.

2.6 Tuning of MIMO PID Controllers

In many MIMO processes, the interactions between the control loops are sufficiently decoupled so that each process output can be controlled by a SISO controller. However, for some processes the loops are so tightly coupled that this cannot be disregarded. The MIMO PID tuning method presented in Paper II will be briefly outlined in this section.

Consider a linear time-invariant process with m inputs and p outputs and given by its frequency response $P(i\omega) \in \mathbf{C}^{p\times m}$ or its transfer function $P(s)$. The process is assumed to be stable and strictly proper, but not necessarily rational. It is further assumed that $m \leq p$ and that $P(0)$ has full rank.

The MIMO PID controller is given by

$$C(s) = K_P + \frac{1}{s}K_I + \frac{s}{1+\tau s}K_D, \tag{2.30}$$

where the controller parameters are real matrices, i.e., $K_P, K_I, K_D \in \mathbf{R}^{m\times p}$. The derivative part of the controller is filtered by a first-order low-pass filter with a fixed time-constant $\tau > 0$. A MIMO PID controller has $3mp$ parameters to tune, which makes manual tuning in the best of cases, cumbersome. Ensuring good robustness and closed-loop stability can even for simple plants be hard.

The constraints, which are imposed on the closed-loop system, are expressed using the H_∞ norm. For a $p \times q$ transfer function H, $\|H\|_\infty$ is its H_∞ norm

$$\|H\|_\infty = \sup_{\Re s \geq 0} \|H(s)\|, \tag{2.31}$$

where $\|H\|$ is the spectral norm, i.e., the maximum singular value. For a stable H the infinity norm can be expressed as

$$\|H\|_\infty = \sup_{\omega \geq 0} \|H(i\omega)\|. \tag{2.32}$$

The aim of the optimization procedure is to obtain a robust controller that has small low-frequency sensitivity and does not use too much actuation. Robustness is captured by imposing upper bounds on the infinity norm of the sensitivity functions S and T defined in (2.12). The transfer function from the reference to the controller output or more importantly, from measurement noise to controller output is given by $Q = C(I+PC)^{-1}$. Requiring $\|Q\|_\infty \leq Q_{\max}$ limits the size of the control signal's response to a change in the reference signal.

Perfect zero steady-state error for constant input will be obtained if $S(0) = 0$, which is implied by $P(0)K_I$ being nonsingular. A Taylor series expansions of $S(s)$ around $s = 0$ gives $S(s) \approx s(P(0)K_I)^{-1}$ for small s. By minimizing $\|(P(0)K_I)^{-1}\|$ the low-frequency sensitivity is reduced. The design problem can be stated as the following, nonconvex, optimization problem

$$\begin{array}{ll} \underset{K_P, K_I, K_D}{\text{minimize}} & \|(P(0)K_I)^{-1}\| \\ \text{subject to} & \|S\|_\infty \leq S_{\max}, \\ & \|T\|_\infty \leq T_{\max}, \\ & \|Q\|_\infty \leq Q_{\max}, \end{array} \tag{2.33}$$

where $S_{\max}$, $T_{\max}$ and $Q_{\max}$ are upper bounds on the largest singular value of the sensitivity functions.

Formulation of the design problem using LMI restriction

The constraint on the sensitivity function can be expressed as

$$\|S(i\omega)\| \leq S_{\max}, \quad \forall \omega \geq 0, \tag{2.34}$$

which is a semi-infinite constraint in the frequency parameter ω. By introducing a large set of N frequency samples $0 < \omega_k < \omega_N$ and replacing the constraints in (2.33) with one constraint for each frequency sample

$$\|S(i\omega_k)\| \leq S_{\max}, \quad k = 0, \ldots, N \tag{2.35}$$

is obtained. The same procedure is applied to the constraints on T and Q. A dense enough grid needs to be chosen so that it captures the relevant changes in the functions. It should also be chosen so that the range of frequencies captures the asymptotic values of $S(i\omega)$, $T(i\omega)$ and $Q(i\omega)$ for both low and high frequencies. Let the subscript k denote a transfer function evaluated at $s = i\omega_k$ to keep the notation un-

cluttered. A sampled version of (2.33) is

$$\begin{array}{ll} \underset{K_P,K_I,K_D}{\text{minimize}} & \|(P(0)K_I)^{-1}\| \\ \text{subject to} & \|S_k\| \le S_{\max}, \\ & \|T_k\| \le T_{\max}, \\ & \|Q_k\| \le Q_{\max}, \\ & k = 1,\ldots,N. \end{array} \tag{2.36}$$

This problem can be reformulated to a problem where the cost function and all the constraints can be written as quadratic matrix inequalities on the form

$$Z^*Z \succeq Y^*Y, \tag{2.37}$$

where Z^* denotes the Hermitian transpose of Z. The symbol $\succeq 0$ is used to denote matrix inequality, $Z \succeq 0$ means that Z is Hermitian and positive semidefinite. Constraints on this form are convex in Y but not in Z. The cost function can be rewritten as $\|(P(0)K_i)^{-1}\| = 1/\sigma_{\min}(P(0)K_I)$, where $\sigma_{\min}$ denotes the smallest singular value of the matrix. Minimization of the cost function in (2.33) is therefore equivalent to maximization of the smallest singular value. By introducing and maximizing a scalar variable t, this can be formulated as

$$\sigma_{\min}(P(0)K_I) \ge t \Leftrightarrow (P(0)K_I)^*(P(0)K_I) \succeq t^2 I, \tag{2.38}$$

which is on the same form as (2.37) where $Z = P(0)K_I$ and $Y = tI$. The constraint on the sensitivity function can be reformulated in a similar way

$$\|S_k\| \le S_{\max} \Leftrightarrow ((I + P_kC_k)^{-1})^*(I + P_kC_k)^{-1} \preceq S_{\max}^2 I \tag{2.39}$$

By multiplying this expression with $1/S_{\max}(I + P_kC_k)^*$ and $1/S_{\max}(I + P_kC_k)$ from the left and right, respectively, the constraint can be expressed as

$$(I + P_kC_k)^*(I + P_kC_k) \succeq 1/S_{\max}^2 I. \tag{2.40}$$

This is a constraint on the form (2.37) with $Z = I + P_kC_k$ and $Y = 1/S_{\max}I$. The same procedure can be applied to the constraints for T and Q and an optimization problem equivalent to (2.36) can be expressed as

$$\begin{array}{ll} \underset{t,K_P,K_I,K_D}{\text{maximize}} & t \\ \text{subject to} & Z_k^*Z_k \succeq Y_k^*Y_k, \quad k = 1,\ldots,M \end{array} \tag{2.41}$$

where the number of constraints is $M = 3N + 1$ and $Z_k = I + P_kC_k$ for all constraints apart from the one connected to the cost function.

The problem stated in (2.41) is nonconvex and to solve it a linear matrix inequality (LMI) restriction is imposed on all the constraints. The LMI restriction is obtained by first noting that for any choice of Z and $\tilde{Z}$

$$0 \preceq (Z - \tilde{Z})^*(Z - \tilde{Z}) = Z^*Z - Z^*\tilde{Z} - \tilde{Z}^*Z + \tilde{Z}^*\tilde{Z} \tag{2.42}$$

is valid. From this inequality it follows that

$$Z^*Z \succeq Z^*\tilde{Z} + \tilde{Z}^*Z - \tilde{Z}^*\tilde{Z} \tag{2.43}$$

and subsequently, the matrix inequality

$$Z^*\tilde{Z} + \tilde{Z}^*Z - \tilde{Z}^*\tilde{Z} \succeq Y^*Y \tag{2.44}$$

implies $Z^*Z \succeq Y^*Y$, for any matrix $\tilde{Z}$. The constraint in (2.44) is convex in Z and Y. The inequality can be formulated as the LMI

$$\begin{bmatrix} Z^*\tilde{Z} + \tilde{Z}^*Z - \tilde{Z}^*\tilde{Z} & Y^* \\ Y & I \end{bmatrix} \succeq 0. \tag{2.45}$$

A convex optimization problem is obtained by applying the LMI restriction on all the constraints in (2.41). The problem is formulated as

$$\begin{aligned} &\underset{t,K_P,K_I,K_D}{\text{maximize}} \quad t \\ &\text{subject to} \quad \begin{bmatrix} Z_k^*\tilde{Z}_k + \tilde{Z}_k Z_k - \tilde{Z}_k\tilde{Z}_k & Y_k^* \\ Y_k & I \end{bmatrix} \succeq 0, \quad k = 1,\ldots,M, \end{aligned} \tag{2.46}$$

which is a semidefinite program.

Using the LMI restriction to obtain MIMO PID controllers

The optimization problem (2.46) can be solved iteratively to obtain the controller parameters. The algorithm is initialized with a pure integrating controller with a low gain, i.e.,

$$K_P = K_D = 0, \quad K_I = \varepsilon P(0)^\dagger \tag{2.47}$$

where $P(0)^\dagger$ is the pseudo-inverse of the stationary gain of the process and $\varepsilon > 0$ is a small scalar chosen such that the constraints in (2.33) are satisfied. The method can be summarized in the following steps. First, the LMI restriction is formed using $\tilde{Z}_k = Z_k^{\text{curr}}$ where Z_k^{curr} is the current value of Z_k. The optimization problem (2.46) is then solved and the procedure is then repeated.

Choosing $\tilde{Z}_k = Z_k^{\text{curr}}$ guarantees that the LMI restriction in each step is feasible. The cost function t is nonnegative and nonincreasing and will therefore converge. The iterations are stopped when the progress in each step is small according to (2.18). The method is summarized in Algorithm 2.

Design of continuous-time MIMO PID controllers using iterative LMI approaches have previously been presented in [Bianchi et al., 2008; Lin et al., 2004; Zheng et al., 2002], where the design is done using a state-space representation and the problem is transformed to that of a static output feedback. However, these methods do not consider plants with time delays. A similar method for discrete-time controllers has been presented in [Lim and Lee, 2008].

Algorithm 2: MIMO PID tuning via Iterated LMI Restriction.

given process frequency data P_k and an $\varepsilon > 0$ such that the controller $C = (\varepsilon/s)P(0)^{\dagger}$ satisfies the constraints.
$i := 0$.
repeat

1. *Form LMI restriction.* Calculate $\tilde{Z}_k = Z_k^{\text{curr}}$ for $k = 1 \ldots M$
2. *Solve.*

$$\begin{aligned} &\underset{t,K_P,K_I,K_D}{\text{maximize}} \quad t \\ &\text{subject to} \quad \begin{bmatrix} Z_k^* \tilde{Z}_k + \tilde{Z}_k Z_k - \tilde{Z}_k \tilde{Z}_k & Y_k^* \\ Y_k & I \end{bmatrix} \succeq 0, \quad k = 1, \ldots, M \end{aligned}$$

3. *Update iteration.* $i := i + 1$

until *stopping criterion is satisfied.*

The method presented in [Galdos et al., 2010] uses convex optimization to obtain MIMO PID controllers. The generalized Nyquist stability criterion is approximated with convex constraints and the norm between the actual and a desired open-loop transfer function is minimized. Design of robust multiloop PID controllers, i.e., diagonal MIMO PID controllers using semidefinite programming have been proposed in [Bao et al., 1999]. The closest prior work is presented in [Saeki et al., 2010] where LMI restrictions are used iteratively but where S, T and Q are lumped together into one function wheras the method presented in this thesis allows for individual constraints on the functions. The method presented in this thesis minimizes the low-frequency sensitivity and provides an initialization procedure that works for all stable plants.

Example and comparison with other PID design methods.

To illustrate how the PID controllers obtained using the LMI restriction method perform, the method will be applied on the well-known Wood-Berry binary distillation column [Wood and Berry, 1973].

The process is given by the transfer function

$$P(s) = \begin{bmatrix} \dfrac{12.8e^{-s}}{16.7s+1} & \dfrac{-18.9e^{-3s}}{21.0s+1} \\ \dfrac{6.6e^{-7s}}{10.9s+1} & \dfrac{-19.4e^{-3s}}{14.2s+1} \end{bmatrix}. \tag{2.48}$$

There is substantial interaction between the inputs and outputs even at steady-state

which can be seen from the relative gain array [Åström and Hägglund, 2006],

$$\text{RGA} = \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix}. \tag{2.49}$$

Tuning a PID controller by hand is difficult because of the coupled dynamics.

The design parameters were chosen as

$$S_{\max} = 1.4, \quad T_{\max} = 1.4, \quad Q_{\max} = 3/\sigma_{\min}(P(0)) = 0.738 \tag{2.50}$$

and the filter time-constant was chosen as $\tau = 0.3$. For details about sampling and initialization, see Section 8 in Paper II. The obtained controller parameters are

$$K_P = \begin{bmatrix} 0.1750 & -0.0470 \\ -0.0751 & -0.0709 \end{bmatrix}, \quad K_I = \begin{bmatrix} 0.0913 & -0.0345 \\ 0.0402 & -0.0328 \end{bmatrix},$$
$$K_D = \begin{bmatrix} 0.1601 & -0.0051 \\ 0.0201 & -0.1768 \end{bmatrix}.$$

To illustrate how the obtained controller performs, it is compared with three previously presented PID controllers designed for the Wood-Berry process. It should be stressed that the compared methods all have different approaches and focuses. The aim of this comparison is to show that the obtained controller gives good performance, since this is not explicitly handled in the optimization problem. Only PID controllers that do not include any additional decoupling filters are included in the comparison in order to make the comparison fair.

The first controller in the comparison is taken from [Tan et al., 2002]. The presented method involves solving an optimization problem using a loop-shaping H_∞ approach, resulting in a high-order controller. A PID controller is then obtained by reducing the high-order controller via a truncated Maclaurin approximation.

The second controller is taken from [Wang et al., 1997]. A relay auto-tuning procedure, which gives information about the stationary gain and the frequency response at the cross-over frequency is used to design a PID controller with a specific gain and phase margin. The obtained controller consists in PI-type controllers in the diagonal elements of C and PID controllers in the off-diagonals.

In [Dong and Brosilow, 1997] a third method for tuning PID controllers is presented. The method constists in finding an IMC controller, designed to provide decoupled control. The IMC controller is then reduced to a PID controller by a Maclaurin series expansion around $s = 0$. The obtained PID unstabilized the closed-loop system and comparison will therefore be done against the obtained PI controller.

An extra low-pass filter was added to the derivative term in the cases where the PID controllers where given in nonproper form. The filter time-constant was chosen as $\tau = 0.01$. The system and controller output responses to unit step changes in the reference signals can be seen in Figure 2.4 and Figure 2.5, respectively. All methods

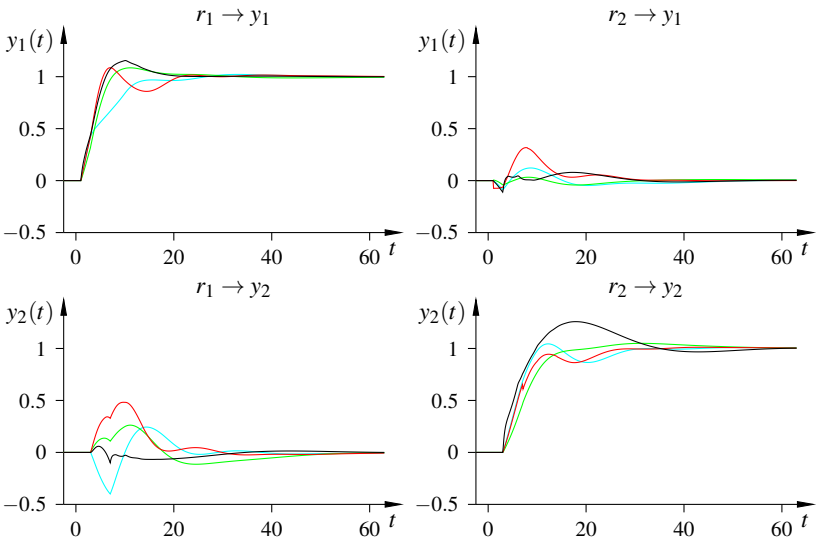

Figure 2.4 Unit step output-responses for the Wood-Berry process. The left column shows the responses of y_1 (upper) and y_2 (lower) for a unit-step in r_1. The right column shows the responses to a unit-step in r_2. The black curves correspond to the responses with the controller obtained using the LMI restriction method. The red, green and cyan correspond to the controllers presented in [Tan et al., 2002], [Wang et al., 1997] and [Dong and Brosilow, 1997], respectively.

give similar responses with approximately the same closed-loop time-constants. The LMI restriction method gives larger overshoots but also more decoupling between the channels. The overshoots could be reduced by introducing set-point weighting. Some performance metrics are presented in Table 2.2 where the IAE is defined as

$$\text{IAE} = \sum_{i,j} \int_0^\infty |e_{ij}(t)| \mathrm{d}t, \quad i = 1,2,\ j = 1,2 \tag{2.51}$$

and the largest peak in the control signal is denoted

$$u_{\max} = \max_{i,j} \max_t |u_{ij}(t)|, \quad i = 1,2,\ j = 1,2. \tag{2.52}$$

All four controllers have comparable performance, judged with the IAE measure, with the LMI restriction method being 11% better than the second best controller by [Dong and Brosilow, 1997]. From Figure 2.4 it can be seen that the LMI restriction method has the least interaction between the loops but a larger overshot than the other methods. It should be noted that the PID controller presented in [Tan et al., 2002] is much more aggressive than the others. The PI controller in [Dong and Brosilow, 1997] is the least aggressive, which is reasonable since it does not have derivative action.

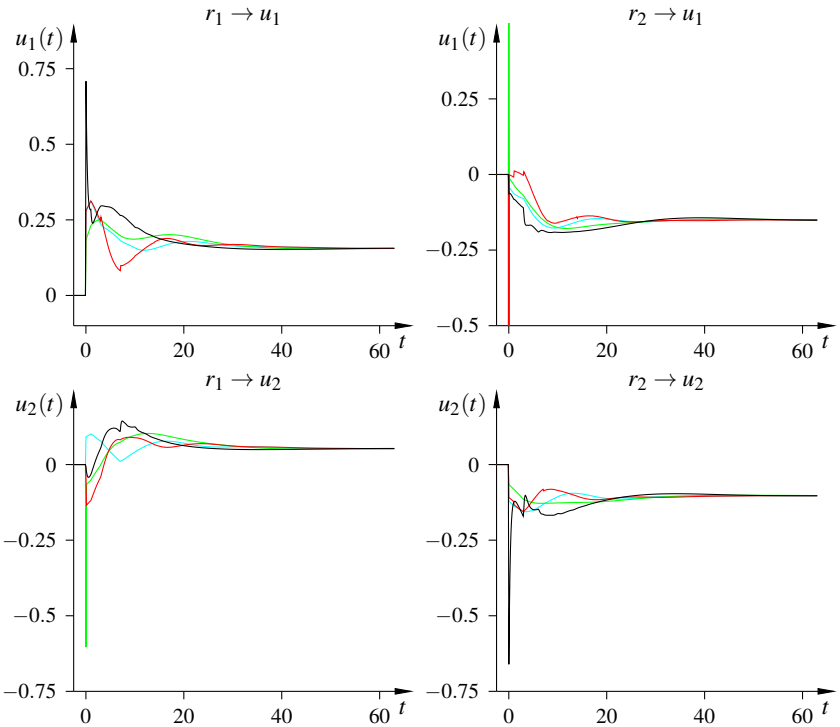

Figure 2.5 Unit step control-signal responses for the Wood-Berry process. The left column shows the responses of u_1 (upper) and u_2 (lower) for a unit-step in r_1. The right column shows the responses to a unit-step in r_2. The black curves correspond to the responses with the controller obtained using the LMI restriction method. The red, green and cyan correspond to the controllers presented in [Tan et al., 2002], [Wang et al., 1997] and [Dong and Brosilow, 1997], respectively.

Table 2.2 Performance measures and control signal peaks for the compared PID controllers.

Design	IAE	$u_{\max}$
LMI	17.7	0.71
Tan et al.	21.2	9.81
Wang et al.	20.1	0.81
Dong et al.	19.9	0.31

2.7 Contributions Overview

The contributions of the work presented in Paper I and II are summarized as methods for designing SISO and MIMO PID controllers that minimize low-frequency sensitivity subject to robustness constraints. The methods rely on the convex-concave procedure that iteratively solves a number of convex optimization problems. In each iteration, the nonconvex constraints, and in the MIMO case also the cost function, are approximated by convex constraints. Since the problem solved in each iteration is convex, the algorithms are fast. Designing a SISO controller takes roughly 5 seconds and a two-input two-output MIMO takes roughly 2 minutes. Examples and comparisons with other methods have shown that the method produces PID controllers which perform well.

3

Feedforward Control

"I was beat before I even had a chance to begin"

— Nicke Andersson, *Time got no time to wait for me*

Feedforward control can be described as taking action based solely on external inputs, without regard to the effect of the action taken. If measurements of the effect are not available it can be the only means of control. However, feedforward controllers are mostly used together with feedback controllers, where their main purposes are to improve reference tracking, to decouple the interaction effects in MIMO control structures, or to improve rejection of measurable disturbances.

Feedforward controllers are in general easier to design than feedback controllers since the stability issue is straightforward. A stable feedforward controller will not destabilize a stable control loop. Stability, and robustness with respect to stability, are two of the main concerns when designing feedback controllers whereas the focus when designing feedforward controllers is on performance.

For many process industry control applications, the PID controller is the standard controller. Apart from static feedforward, no default feedforward controller exists. In the work presented in this thesis, the emphasis is on simple controllers, in the sense that they have few parameters. This also allows us to derive tuning rules from which the controller parameters can easily be calculated. More complex feedforward controllers could possibly give better performance but the aim has been to provide tuning rules for controllers that are of comparable complexity with the PID controller. These simple controllers can provide substantial improvements in performance.

This chapter will review and serve as a summary of the results presented in Paper III-V.

3.1 Feedforward Controllers

Consider the feedforward control structure in Figure 3.1. The objective is to design the feedforward controller $F_d(s)$ so that the effect of the disturbance d on the output y is as small as possible. The transfer function from d to y is given by

$$G_{yd}(s) = P_d(s) - P_u(s)F_d(s), \tag{3.1}$$

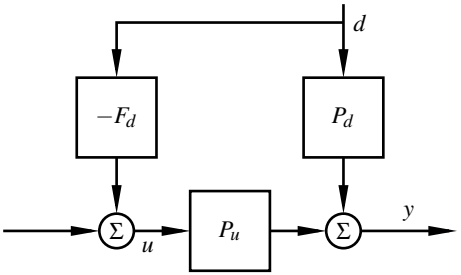

Figure 3.1 Open-loop feedforward controller structure.

where P_u is the process model and P_d is the disturbance model. The controller

$$F_d(s) = \frac{P_d(s)}{P_u(s)} \tag{3.2}$$

completely attenuates the disturbance. This will be referred to as the ideal feedforward controller. There are, however, a number of issues with this controller. The controller is unstable if P_u has zeros in the right-half plane. The controller could be nonrealizable if the time-delay in P_u is larger than that of P_d or if the number of zeros is larger than the number of poles. Furthermore, even if the controller is stable and realizable, the variations of the control signal u could be too large for common disturbances. There is therefore more to the design of feedforward controllers than simply inverting the process dynamics. The controller (3.2) is also, as every open-loop control scheme, sensitive to variations in the process.

As mentioned above, feedforward control is often done in combination with feedback, see Figure 3.2. The transfer function from the disturbance to the process

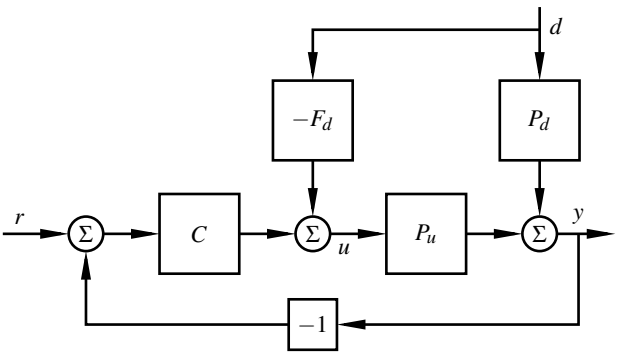

Figure 3.2 Classical feedback structure with feedforward from a measurable disturbance.

output is given by

$$G_{yd} = \frac{P_d - P_u F_d}{1 + P_u C}. \tag{3.3}$$

The ideal feedforward controller (3.2) will completely reject the disturbance also in closed-loop. Focus will therefore be on scenarios where the controller (3.2) is nonrealizable. For such a feedforward controller, the feedback interconnection will reduce the the sensitivity to process variations for frequencies where $|S(i\omega)| < 1$. With the structure in Figure 3.2, the two controllers will interact and both act on the disturbance. For a feedforward controller that is designed for optimal performance in the open-loop setting, the interaction might have a severe impact on the overall performance, usually with large overshoots as the result. If, on the other hand, the feedforward controller is designed based on the closed-loop dynamics, the interaction is taken into account. However, should the feedback controller be retuned, the feedforward controller also needs to be retuned.

Tuning rules for low-order feedforward controllers using the conventional controller structure have been presented in [Guzmán and Hägglund, 2011]. This work has served as inspiration for the tuning methods presented in Paper III.

3.2 Disturbance Rejection

Feedback controllers are often designed to attenuate load disturbances. However, the controller acts on the disturbances only when a control error arises and it could, e.g., for robustness reasons, give a sluggish response. Attenuation of measurable disturbances can be improved by the addition of a feedforward controller.

A structure that facilitates the tuning of the feedforward controller and decouple the feedback and the feedforward controllers is shown in Figure 3.3 where a decoupling filter $H(s)$ has been added. This is equivalent to the decoupling feedforward structure presented in [Brosilow and Joseph, 2002]. The transfer function from d to y is

$$G_{yd} = \frac{P_d - P_u F_d + P_u C H}{1 + P_u C}. \tag{3.4}$$

By choosing the decoupling filter as

$$H = P_d - P_u F_d, \tag{3.5}$$

the transfer function becomes $G_{yd} = P_d - P_u F_d$, which is identical to the open-loop transfer function (3.1). Thus, H provides a second degree of freedom that decouples the feedback and feedforward controllers and allows design of the feedforward controller to be done by only considering the open-loop structure in Figure 3.1. The decoupling filter is not necessary if the ideal controller (3.2) is implemented but it provides advantages when this cannot be done.

The decoupling filter subtracts the open-loop response of the measurable disturbance from the control error. To see what happens in the case of model errors let $\bar{P}_d$

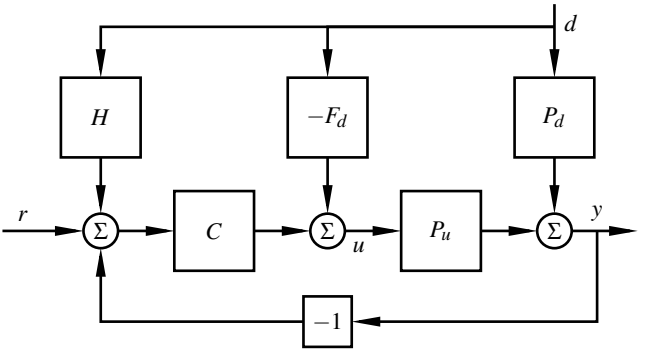

Figure 3.3 Decoupled feedback/feedforward structure.

and $\bar{P}_u$ be approximate models of P_d and P_u, respectively. Furthermore, let F_d and H be designed based on the approximate models, i.e.,

$$H = \bar{P}_d - F_d \bar{P}_u. \tag{3.6}$$

The transfer function from d to y is then given by

$$G_{yd} = S\left(P_d - \bar{P}_d - F_d\left(P_u - \bar{P}_u\right)\right) + \bar{P}_d - F_d \bar{P}_u, \tag{3.7}$$

where S is the sensitivity function. The transfer function is a sum of the expected open-loop response and an additional term that governs the response due to the model mismatch. If the model is perfect, i.e., $\bar{P}_d = P_d$ and $\bar{P}_u = P_u$, the mismatch term vanishes. For any robust, and reasonably designed feedback controller, S is small for low frequencies and close to one for high frequencies. A model mismatch in the low-frequency region will therefore have little impact on the closed-loop response. This implies that the controllers will reject constant disturbances even if the stationary gains of $\bar{P}_u$ and $\bar{P}_d$ are not equal to those of P_u and P_d, respectively. For processes with small high-frequency gain, the impact of a mismatch at those frequencies will also be negligible. However, care must be taken so that the models capture the frequency characteristics in the mid-frequency region where the gains of S, P_u and P_d are largest.

Design of low-order feedforward controllers using the decoupled controller structure has also been addressed in [Rodríguez et al., 2013], where a number of tuning rules for reducing IAE, ISE and overshoot are presented. Tuning rules for optimal feedforward design for systems with right-half plane zeros, using the decoupling structure, has been addressed in [Rodríguez et al., 2014].

ISE minimizing controllers

The tuning rules presented in Paper III and IV are based on the structure in Figure 3.3, which allows design of the feedforward controller by considering the open-loop transfer function (3.1).

Let the processes be defined as

$$P_i(s) = \frac{K_i}{1+sT_i} \mathrm{e}^{-L_i s}, \tag{3.8}$$

where P_i denotes either the disturbance model P_d or the process model P_u with corresponding parameters. The processes are assumed to be stable i.e., T_d and T_u are nonnegative. The low-order feedforward controller is given by

$$F_d(s) = K_{\mathrm{ff}} \frac{1+sT_z}{1+sT_p} \mathrm{e}^{-sL_{\mathrm{ff}}}. \tag{3.9}$$

This can be interpreted as a lead/lag filter with a time-delay, or as a filtered and time-delayed PD controller. Inclusion of an integrator in the controller is not desired since constant disturbances would cause the control signal to grow large over time. The parameters are the static gain K_{ff}, the filter time-constant T_p, the time-delay L_{ff} and the derivative time T_z.

For a unit-step disturbance the output, in time-domain, is given by

$$y(t) = \mathscr{L}^{-1}\left((P_d - F_d P_u)/s\right) \tag{3.10}$$

where $\mathscr{L}^{-1}$ denotes the inverse Laplace operator.

The objective is to find the controller parameters that minimize ISE, defined in (2.6). The optimal controller is the solution to the following optimization problem

$$\begin{aligned} \text{minimize} \quad & \int_0^\infty y(t)^2 \mathrm{d}t \\ \text{subject to} \quad & T_p \geq 0 \\ & L_{\mathrm{ff}} \geq 0 \end{aligned} \tag{3.11}$$

where the constraints ensure that the controller is causal and stable. If the time-delay of the disturbance model is greater than that of the process model, the optimal controller is obtained from (3.2) as

$$F_d(s) = \frac{K_d}{K_u} \frac{1+sT_u}{1+sT_d} \mathrm{e}^{-s(L_d - L_u)} \tag{3.12}$$

for which (3.10) is zero regardless of the disturbance. The remainder of this section will focus on the case where perfect disturbance rejection is not possible.

The derivation of the solution to the optimization problem is presented in Paper III and the solution is simplified in Paper IV. The optimal static gain and time-delay are easily derived and intuitively simple. For the open-loop response to converge to zero, the optimal stationary gain must be the same as in the ideal case i.e.,

$$K_{\mathrm{ff}} = \frac{K_d}{K_u}. \tag{3.13}$$

The disturbance affects the output before any action from the controller can pass through the dynamics of the process. Any additional time-delay in the controller would only increase the ISE and

$$L_{\text{ff}} = 0 \tag{3.14}$$

is therefore the optimal choice. The optimal choice of the two remaining parameters are conveniently expressed by the auxiliary parameters

$$a = \frac{T_u}{T_p}, \quad \text{and} \quad b = a(a+1)\mathrm{e}^{\frac{L_u - L_d}{T_d}}. \tag{3.15}$$

Using these parameters, the optimal choice of the filter parameter is

$$T_p = \begin{cases} \dfrac{3a - 1 - b + (a-1)\sqrt{1+4b}}{b-2} T_d & \text{if } b < 4a^2 + 2a \text{ or } b < a + \sqrt{a} \\ 0 & \text{otherwise.} \end{cases} \tag{3.16}$$

The optimal choice of the derivative time is

$$T_z = (T_p + T_u)\left(1 - \frac{2T_u}{b(T_d + T_p)}\right) \tag{3.17}$$

where T_p corresponds to the optimal parameter from (3.16). The choice of $T_p = 0$ is optimal for certain processes but unfortunately such a controller is nonrealizable. A remedy for this could be to choose a small T_p and then calculate T_z based on that choice. Another approach that will be described in the next section is to add a filter, which also reduces the sensitivity to measurement noise.

Practical considerations

The ISE optimal controller is derived assuming noise-free measurements and no control-signal limitations, which of course is an idealized picture of reality. In an industrial setting, the measurements of the disturbances are likely corrupted by noise, which the optimal controller can be sensitive to. Figure 3.4 shows a block diagram where noise n is added to the measurement of the disturbance. High-frequency noise can result in unwanted control activity that will increase the wear of the actuator. To reduce the impact of such noise, the controller can be fitted with a low-pass filter. In the work presented in Paper IV, it is proposed that the optimal controller is fitted with a second order low-pass filter, i.e., the feedforward controller is given by

$$F_d(s) = K_{\text{ff}} \frac{1 + sT_z}{(1 + sT_p)(1 + sT_f)^2} \mathrm{e}^{-sL_{\text{ff}}}. \tag{3.18}$$

The filter time-constant of the low-pass filter T_f is tuned so that the controller has satisfactory control signal characteristics. This sacrifices performance in favor of noise suppression and reduced actuator wear. The order of the filter was chosen so that the controller would have roll-off also for $T_p = 0$. A number of guidelines on how to choose T_f are also presented in Paper IV, and they are summarized in this section.

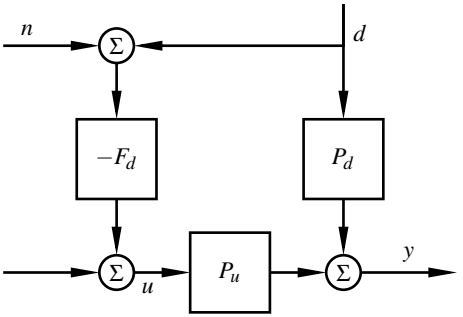

Figure 3.4 Open-loop structure with additive noise on the measurements of the disturbance.

Effects of control signal filtering Assuming that n is white noise, the variance of the control signal u, with respect to the noise, is given by

$$\mathrm{var}(u) = \frac{K_{\mathrm{ff}}^2}{4T_f}\left(1 + \frac{T_z^2 - T_p^2}{(T_f + T_p)^2}\right). \tag{3.19}$$

The expression for the variance is equal to the integrated squared control signal

$$\mathrm{ISU} = \int_0^\infty \left(\frac{\mathrm{d}u(t)}{\mathrm{d}t}\right)^2 \mathrm{d}t \tag{3.20}$$

for a unit-step in d. This is a measure of the control signal activity from which it can be seen that increasing the filtering will give a smoother control signal. With the low-pass filtered controller, the derivative of the control signal will be smaller than it is with the optimal controller. Additive white noise corruption of the measurements is of course an idealized assumption of the noise properties, but the expressions (3.19) and (3.20) serve as good illustrations of how the filtering affects the properties of the control signal.

Control signal peak For processes such that the optimal T_p is zero, the filtering is necessary in order to get a realizable controller. From a control signal perspective this can be seen as the worst-case since $T_p > 0$ provides an extra order of filtering. For $T_p = 0$, a number of control-signal related measures can be analytically derived. For a unit-step disturbance, the peak in the control signal will be

$$u_{\mathrm{peak}} = -K_{\mathrm{ff}}\left(1 + \frac{T_z - T_f}{T_f}\mathrm{e}^{T_z/(T_f - T_z)}\right). \tag{3.21}$$

Limiting the peak to be a factor Δ of K_{ff}, i.e.,

$$u_{\mathrm{peak}} = -K_{\mathrm{ff}}\Delta \tag{3.22}$$

and solving for T_f gives

$$T_f = \frac{T_z}{1 + \dfrac{1}{W_0\left(\frac{\mathrm{e}^{-1}}{\Delta - 1}\right)}} \tag{3.23}$$

where W_0 is the principal branch of the Lambert W-function [Corless et al., 1996].

Controller gain reduction Limiting the peak of the controller signal and using (3.23) to calculate a suitable filter time-constant is one design approach. Another is to chose T_f so that

$$\max_{\omega} |F_d(i\omega)| = \lambda K_{\mathrm{ff}}, \quad \lambda > 1 \tag{3.24}$$

and thereby limiting the largest frequency gain of the controller. The lower bound on λ is introduced since the structure of the controller (3.18) does not allow the maximum gain to be less than the stationary gain. The largest magnitude of the frequency function has a peak, i.e., is larger than K_{ff} for some frequencies, if $T_z > T_p$ and

$$0 < T_f < \hat{T}_f, \tag{3.25}$$

where $\hat{T}_f = \sqrt{\frac{T_z^2 - T_p^2}{2}}$ is the smallest filter time-constant for which $\max_\omega |F_d(i\omega)| \geq K_{\mathrm{ff}}$. For controllers with $T_p = 0$, an expression for how to choose T_f can be found analytically whereas for $T_p \neq 0$ an approximate expression was derived in Paper IV. The expressions for T_f are given by

$$T_f = \begin{cases} \dfrac{T_z}{\sqrt{2}} \sqrt{1 - \sqrt{1 - \dfrac{1}{\lambda^2}}} & \text{if } T_p = 0 \\ \dfrac{\hat{T}_f(1+\lambda)}{2\lambda} \sqrt{1 - \dfrac{2\lambda(T_z - \lambda T_p)(T_z + T_p)}{(1+\lambda)^2 \hat{T}_f^2}} & \text{if } T_p \neq 0. \end{cases} \tag{3.26}$$

Precompensation If $L_u \leq L_d$, the ideal controller given in (3.12) inverts the system dynamics and will completely attenuate any disturbance. However, this optimal controller can also be sensitive to noise and have a large high-frequency gain, which gives rise to aggressive actuation. The addition of a low-pass filter is therefore advisable. To counteract the lag introduced by the filter the time-delay of the controller can be adjusted so that the performance loss is small. Let the controller be given by (3.18) and the time-delay in the controller be

$$L_{\mathrm{ff}} = L_d - L_u + \delta \tag{3.27}$$

where δ is a time-delay shift. Let $y_f(t)$ be the step-response obtained through (3.10) and J_f be the ISE obtained with this controller. Then J_f is a convex function of δ

since $\frac{\mathrm{d}^2 J_f}{\mathrm{d}\delta^2} \geq 0$. The optimal time-delay shift is thus given by the unique solution to $\frac{\mathrm{d}J_f}{\mathrm{d}\delta} = 0$, which is

$$\delta^* = \ln\left(\frac{2T_d^3(T_d+T_z)}{(T_f+T_d)^2(T_p+T_d)(T_u+T_d)}\right)T_d. \tag{3.28}$$

For the nonfiltered controller, the optimal choice of parameters was $T_p = T_d$ and $T_z = T_u$ for which the optimal time-delay shift simplifies to

$$\delta^* = 2\ln\left(\frac{T_d}{T_f+T_d}\right)T_d. \tag{3.29}$$

The total time-delay (3.27) must be positive, which together with (3.29) gives an upper bound

$$T_f \leq T_d\left(\mathrm{e}^{(L_d-L_u)/2T_d} - 1\right) \tag{3.30}$$

on the time constant in the low-pass filter. This inequality states how much filtering can be applied before the time delay in the controller is zero. Some of the performance losses introduced by filtering can therefore be counteracted by reducing the time-delay in the controller.

3.3 Set-Point Weighting

The control system should, apart from stabilizing and rejecting disturbances, also track changes in the reference signal. A well-designed feedback controller can accomplish this, but to boost performance feedforward techniques should be applied. In many available PID implementations, feedforward is included through set-point weighting, which means that the reference is weighted in the P and D part of the controller. In such a PID controller, the control signal is calculated as

$$u(t) = K\left(br(t) - y(t) + \frac{1}{T_i}\int_0^t e(t)\mathrm{d}t + T_d\frac{\mathrm{d}}{\mathrm{d}t}\left(cr(t) - y(t)\right)\right) \tag{3.31}$$

where b and c are the set-point weights. Note that the standard PID controller (2.1) is obtained for $b = c = 1$. By increasing the parameters, the controller will be more aggressive when the reference changes and vice versa. A control system with the control law (3.31) can be represented as depicted in Figure 3.5 where $C(s)$ is the transfer function of the standard PID controller defined in (2.2) and

$$F_r(s) = K\big((b-1) + (c-1)T_d s\big) \tag{3.32}$$

is a feedforward controller related to the set-point weights.

A method for finding the optimal set-point weights is presented in Paper V. The method is applied on a batch of processes to obtain tuning rules for how to tune the b parameter. In this section, the method is described briefly and the tuning rules for b are evaluated with PID controllers tuned using five different methods.

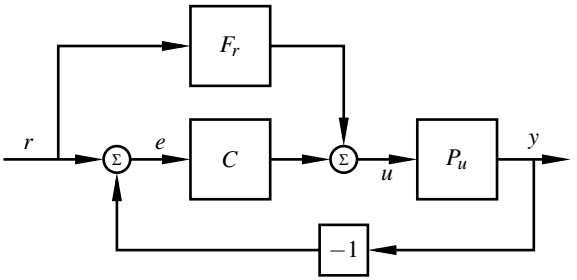

Figure 3.5 Controller structure with feedforward from the reference to the control signal.

Preliminaries

Given the system in Figure 3.5, the error and control signal are given by

$$E = S(1 - P_u F_r)R \tag{3.33}$$
$$U = S(C + F_r)R, \tag{3.34}$$

respectively. The signals are affine in b and c since F_r is an affine function of the parameters. The time-domain signals, which can be calculated by the inverse Laplace transformation of (3.33) and (3.34), can be expressed as

$$e(t) = e_0(t) + Z(t)x \tag{3.35}$$
$$u(t) = u_0(t) + W(t)x, \tag{3.36}$$

where $x = [b \;\; c]^T$, $Z(t) = \mathscr{L}^{-1}(SP_uKR[1 \;\; T_d s])$ and $W(t) = \mathscr{L}^{-1}(SKR[1 \;\; sT_d])$. The functions $e_0(t)$ and $u_0(t)$ denote the error and control signals for $b = c = 0$.

To be able to formulate a tractable optimization problem that can be solved by a numerical solver, the time signals are sampled over a grid of N time instances. By evaluating the functions (3.35) and (3.36) at the grid points, their sampled counterparts are obtained and given by

$$\mathbf{e} = \mathbf{e}_0 + \mathbf{Z}x \tag{3.37}$$
$$\mathbf{u} = \mathbf{u}_0 + \mathbf{W}x \tag{3.38}$$

where $\mathbf{e}$, $\mathbf{e}_0$, $\mathbf{u}$ and $\mathbf{u}_0$ are vectors of length N, and $\mathbf{Z}$ and $\mathbf{W}$ are $N \times 2$ matrices. These expressions allow formulations of a wide variety of optimization problems where basically any convex function can be applied to $\mathbf{e}$, $\mathbf{u}$ and be used to form cost functions or constraints suitable for the application at hand. Examples of such formulations include minimization of norms of the error with constraints on the magnitude of the control signal. For convex optimization problems, a large number of grid points can be used. The same method can be used for feedforward disturbance

rejection and the control signal derivative could be derived in a similar fashion. This is described in Paper V.

The method of expressing the error and control signals as linear functions of the controller parameters and using that to formulate an optimization problem with a quadratic cost function, which can be solved analytically, was presented in [Leva and Bascetta, 2006].

Tuning rules for the proportional set-point weight

The method described in the previous section can be applied and tailored to fit any process, provided that it is linear and time-invariant. However, an optimization problem must be solved for every control loop in the plant. The aim has therefore been to find simple tuning rules for the set-point weights that give close to optimal performance.

For PI controllers, T_d is zero and the derivative set-point c is therefore superfluous. For PID controllers, it was observed that the optimal c depended heavily on the filtering of the derivative term. To avoid aggressive control action when the set-point is changed, the parameter c is often set to zero [Åström and Hägglund, 2006]. Therefore, two rules for the proportional set-point weight has been developed, one for PI control and another for PID control with $c = 0$.

The objective is to find the b that minimizes IAE for a given process P_u and controller C, either a PI or PID with $c = 0$. The control error can be calculated by (3.33)-(3.37) and used to formulate the optimization problem

$$\underset{b}{\text{minimize}} \; \|\mathbf{e}\|_1 \tag{3.39}$$

The AMIGO tuning rules [Åström and Hägglund, 2006] were derived by applying the MIGO design method [Panagopoulos et al., 2002] on a large batch of processes. Expressions that approximately described the optimal PID parameters in terms of the parameters from a first-order time-delayed model were used to form the tuning rules. Inspired by this work, (3.39) is solved for all processes in that batch and the obtained optimal solutions will be used to derive tuning rules for b.

Tuning rules for the proportional set-point weight that reduces overshoot have been presented in, e.g., [Chidambaram, 2000], where the analysis is restricted to FOTD processes, and [Hang et al., 1991] where the rules are based on the product of the steady-state gain and the ultimate gain. Tuning rules for b with the aim of making the set-point response be that of a specified FOTD are presented in [Vilanova, V. Alfaro, et al., 2012]. In [Taguchi and Araki, 2000] a batch of processes, similar to the AMIGO batch, were used to derive separate tuning rules for all process types.

The AMIGO batch of processes. The batch used to derive the AMIGO tuning rules is divided into the following nine process types

$$P_1 : \frac{e^{-s}}{1+sT}, \qquad P_4 : \frac{1}{(s+1)^n}, \qquad P_7 : \frac{T\,e^{-sL_1}}{(1+sT)(1+sT_1)},$$

$$P_2 : \frac{e^{-s}}{(1+sT)^2}, \qquad P_5 : \prod_{i=0}^{3} \frac{1}{(1+\alpha^i s)}, \qquad P_8 : \frac{1-\alpha s}{(s+1)^3},$$

$$P_3 : \frac{1}{(s+1)(1+sT)^2}, \quad P_6 : \frac{e^{-sL_1}}{s(1+sT_1)}, \qquad P_9 : \frac{1}{(s+1)((sT)^2+1.4sT+1)}$$

where T, T_1, L_1, n and α take different values, adding up to 134 processes in total; see [Åström and Hägglund, 2006] for a complete description of all the processes used in the batch. The model orders range from one to eight and the batch includes both integrating processes and processes with nonminimum phase-behavior. The process types 1-7 have monotone step responses and 8-9 have an *essentially* monotone step-response [Åström and Hägglund, 2006], which is defined as

$$\frac{\int_0^\infty g(t)\mathrm{d}t}{\int_0^\infty |g(t)|\mathrm{d}t} > 0.8, \tag{3.40}$$

where $g(t)$ is the impulse response of the process.

Derivations of the tuning rules. The optimization problem (3.39) was solved for all processes in the batch with a PI and PID controller tuned using the AMIGO rules. The D-part is filtered with a first order low-pass filter with the time constant $T_f = 10^{-5}$ in order to obtain a proper PID controller. The optimal b parameters that were obtained can be seen in Figure 3.6, where they are plotted against the product of the proportional gain of the controller K and the static gain of the process $P(0)$. The integrating processes have infinite $P(0)$ and are not plotted but are included in the analysis. As can be seen in the figure, there is a clear trend that as $KP(0)$ increases, the optimal set-point weight decreases. Functions on the form

$$\frac{1}{\alpha KP(0)} + \beta \tag{3.41}$$

were fitted to the data. The best fit for the PI case was $\alpha = 1.9$ and $\beta = 0.7$. For the integrating processes, the optimal b was approximately 0.745. Based on this and favoring simplicity, the tuning rule for how to choose b for PI controllers is given by

$$b^*_{PI} = \frac{1}{2KP(0)} + 0.75. \tag{3.42}$$

For systems controlled by PID controllers, with $c = 0$, the best fit was obtained for $\alpha = 2$ and $\beta = 0.55$ and the tuning rule is given by

$$b^*_{PID} = \frac{1}{2KP(0)} + 0.55. \tag{3.43}$$

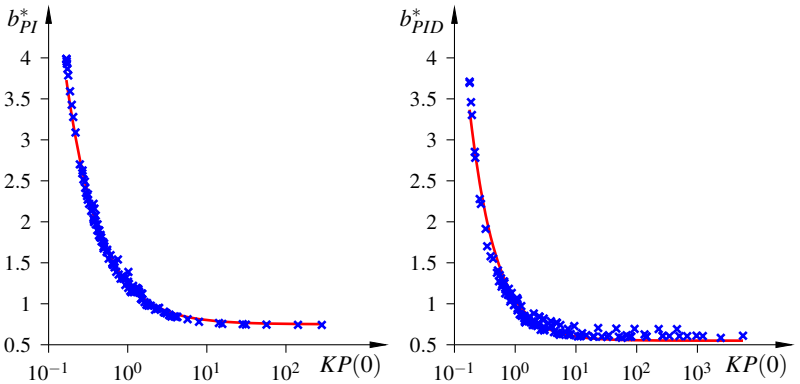

Figure 3.6 Optimal proportional set-point weights for all processes in the AMIGO batch in blue and the tuning rules in red. The left plot shows the optimal b for PI controllers and the right plot shows the parameter for PID controllers.

For a PID controller with a first order low-pass filter on the derivative term and a unit step in the reference, the initial control signal will be

$$\lim_{t\to 0} u(t) = K\left(b + cT_d/T_f\right) \tag{3.44}$$

where T_f is the filter time constant. The tuning rules for b are derived with either T_d or c equal to zero and hence, the initial control signal will be a factor b larger compared with the default $b = 1$.

Allowing the b parameter to be larger than one could be perceived as unorthodox and is in many cases not permitted in commercially available PID controllers [V. M. Alfaro et al., 2010]. However, as reported in [V. M. Alfaro et al., 2009], it can be needed in order to obtain good step-responses. For controllers where $KP(0)$ is small, a large b will decrease the IAE as will be shown in the next section.

Evaluation of the tuning rule

The tuning rules (3.42) and (3.43) were derived using PI(D) controllers tuned by the AMIGO method. To evaluate how the tuning rules (3.42) and (3.43) influence the performance and control signal activity, they are tested with PID controllers tuned by four other methods. Two of the methods use FOTD models of the process to calculate the PID parameters and two are optimization based methods. The next sections briefly describes the feedback tuning methods.

Lambda tuning The first method is the well-known λ-method [Sell, 1995]. Based on an FOTD model

$$\frac{K_p}{1+sT}\mathrm{e}^{-sL} \tag{3.45}$$

of the process, the PI parameters are given by

$$\begin{aligned} K &= \frac{1}{K_p}\frac{T}{L+T_{cl}} \\ T_i &= T. \end{aligned} \tag{3.47}$$

where T_{cl} is a design parameter that reflects the closed-loop time constant. The PID parameters are given by

$$\begin{aligned} K &= \frac{1}{K_p}\frac{L/2+T}{L/2+T_{cl}} \\ T_i &= T+L/2 \\ T_d &= \frac{TL}{L+2T}, \end{aligned} \tag{3.49}$$

For all the processes in the batch, the closed-loop time constant is chosen as $T_{cl} = L+T$. This is a conservative choice, but the aim is to illustrate the effects of the tuning rules for b rather than an evaluation of the lambda tuning method. The integrating processes were excluded in the evaluation since the lambda tuning method is based on FOTD models and hence does not provide PID parameters for such processes.

SIMC The second method consists of variations of the SIMC tuning rules. For processes that are not integrating, the PI controllers are tuned using the rules presented in [Skogestad and Grimholt, 2012] with the parameters given by

$$\begin{aligned} K &= \frac{1}{K_p}\frac{L/3+T}{L+T_{cl}} \\ T_i &= \min\left\{T+L/3, 4(T_{cl}+L)\right\}. \end{aligned} \tag{3.51}$$

The SIMC PID tuning rules based on FOTD models [Grimholt and Skogestad, 2015] are derived for PID controllers on serial or cascade form i.e.,

$$C(s) = K'\left(1+\frac{1}{sT_i'}\right)(1+sT_d') \tag{3.52}$$

with parameters given by

$$\begin{aligned} K' &= \frac{1}{K_p}\frac{T}{L+T_{cl}} \\ T_i' &= \min\left\{T, 4(T_{cl}+L)\right\} \\ T_d' &= L/3. \end{aligned} \tag{3.54}$$

These parameters were used for the nonintegrating processes in the batch. For the integrating processes, type P_6, the feedback controllers are tuned using the rule presented in [Skogestad, 2003]. The proportional and integral gains for both the PI and

the PID controllers are given by

$$\begin{aligned} K' &= \frac{1}{T_{cl}+L} \\ T_i' &= 4(T_{cl}+L). \end{aligned} \tag{3.56}$$

The derivative gain for the PID controllers is $T_d' = T$.

All PI and PID controllers were tuned with the same choice of T_{cl} as in the Lambda method.

Convex-concave procedure The third method is the one presented in Paper I. PI and PID controllers are obtained by solving an optimization problem that maximizes integral gain subject to constraints on the maximum sensitivity functions. The optimization problem was formulated so that the obtained controllers gave maximum sensitivities $M_S \leq 1.4$ and $M_T \leq 1.4$. It should be mentioned that for some processes in the batch, the obtained controller gave oscillatory step responses. However, the purpose of this section is to illustrate the benefits of using the set-point weighting tuning rule and not to evaluate the feedback controller performances.

IAE optimal controllers The fourth method is the optimization based tuning presented in [Garpinger and Hägglund, 2008]. For each process, PI and PID controllers are obtained by solving optimization problems with the objective of minimizing IAE subject to constraints on the maximum sensitivities. The obtained controllers have $M_S \leq 1.4$ and $M_T \leq 1.4$.

Results and summary of evaluation PI and PID feedback controllers were tuned with each method for all processes in the batch. The performance, measured by the IAE, defined in (2.4), with b chosen according to the tuning rules (3.42) and (3.43) was then compared to the default value $b = 1$. The reference signal is a unit-step. In order to verify that the control signals are not excessively amplified when using the tuning rules, the control signal ratio

$$\Delta u_{\max} = \frac{\max\limits_t |u_{b=b^*}(t)|}{\max\limits_t |u_{b=1}(t)|} \tag{3.57}$$

is calculated. The tuning rules are tested on 1320 control systems with an even split of PI and PID feedback controllers.

The increase in performance for PI controllers can be seen in Figure 3.7. On each box in the figure, the red mark is the median, the edges of the box are the 25th and 75th percentiles and the whiskers extend to the most extreme data points. The increase is as large as 45 % for controllers tuned with the AMIGO method. For 36 out of the 660 systems, the performance deteriorated when the tuning rule (3.42) was used. The worst decrease in performance was a modest 0.5 %. The median improvement was larger for the Lambda and SIMC methods than for the optimization based methods. One reason for this is the conservative choice of the closed-loop

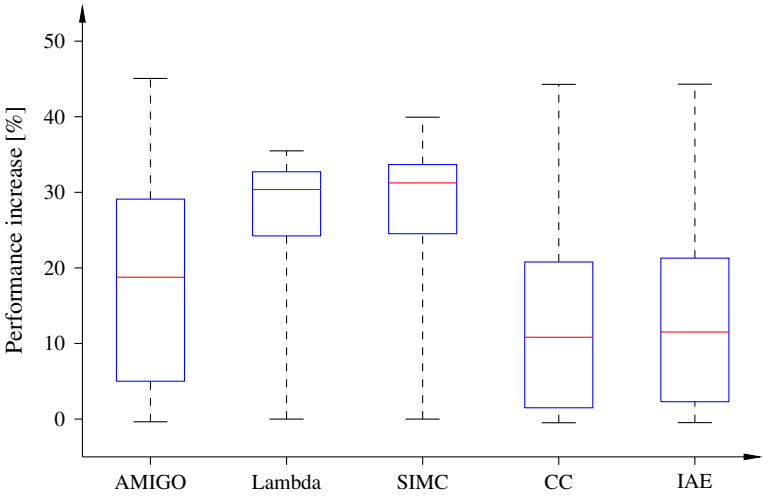

Figure 3.7 Performance increase using the tuning rules (3.42) with PI controllers tuned with five different methods. The performance is compared with the default $b = 1$. In each box, the red mark is the median, the edges of the box are the 25th and 75th percentiles and the whiskers extend to the most extreme data points. CC denotes controllers obtained from the the convex-concave procedure and IAE denotes controllers obtained from the IAE minimization.

time constant, $T_{cl} = T + L$, which implies that there is further room for improvement compared to the optimization-based tuning methods.

The relative increase in control signal, $\Delta u_{\max}$, can be seen in Figure 3.8. The Lambda and SIMC method stand out with larger changes in $\Delta u_{\max}$. This is yet another result of the conservative tuning of the feedback controllers. For the optimization-based design methods and the AMIGO method, the largest change in control signal is within ± 25 %. This is to be considered a reasonable price to pay for a possible 45 % increase in performance. For the controller tuned by the AMIGO or optimization based techniques, approximately 25 % of the cases experienced a decrease in the maximum control effort, but still achieved an improvement in performance.

The performance increase and $\Delta u_{\max}$ for PID controllers, with $c = 0$, are shown in Figure 3.9 and 3.10. As for the PI controllers, the increase in performance is up to 45 %. For 46 out of the 660 systems, the performance deteriorates when the tuning rule (3.43) was used, with a worst case loss of 1.15 %. For the optimization-based methods, approximately 75 % of the systems experienced a decrease in $\Delta u_{\max}$.

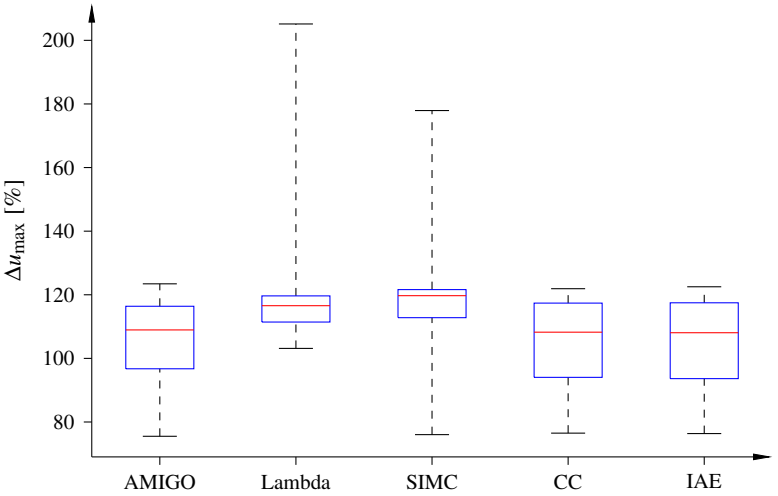

Figure 3.8 Increase in control signal peak when using the tuning rules (3.42) with PI controllers tuned with five different methods. In each box, the central red mark is the median, the edges of the box are the 25th and 75th percentiles and the whiskers extend to the most extreme data points. CC denotes controllers obtained from the the convex-concave procedure and IAE denotes controllers obtained from the IAE minimization.

Extensions and variations

The method used for finding the optimal proportional set-point weight can be extended to any feedforward controller that is linear in the parameters i.e., the denominator is fixed. Constraints on the control signal and its derivative could be included in the optimization problem in which the cost functions and constraints could be tailored with a vast number of possible formulations. The aim was to derive a simple tuning rule that provides a substantial increase in performance. The set-point weights are often tuned so that there is no overshoot when doing step-changes in the reference. Inclusion of this into the optimization problem is readily done since such a constraint is convex in the set-point weights. Constraints that limit the control signal and its derivative could also easily be added. When deriving the tuning rules, IAE was used as the performance measure, but any norm of the error (or control signal) could have been used. The method could also be used for design of feedforward controllers from measurable disturbances.

3.4 Contributions Overview

In Paper III, tuning rules that minimize the impact of measurable disturbances for a system of FOTD processes are presented. The ISE-optimal controller is derived

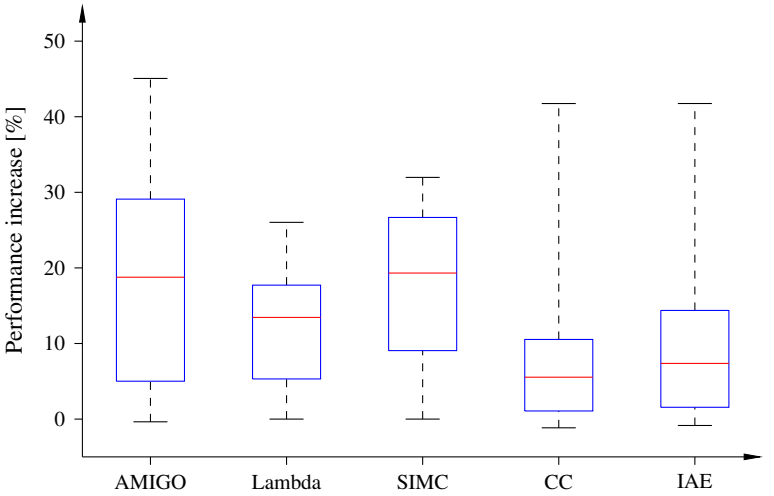

Figure 3.9 Performance increase using the tuning rules (3.43) with PID controllers tuned with five different methods. The derivative set-point weight is $c = 0$. The performance is compared with the default $b = 1$. In each box, the red mark is the median, the edges of the box are the 25th and 75th percentiles and the whiskers extend to the most extreme data points. CC denotes controllers obtained from the the convex-concave procedure and IAE denotes controllers obtained from the IAE minimization.

for an open-loop setting and the optimal solution is used as a tuning rule. A controller structure that decouples the feedforward and the feedback action allows not only the controllers to be tuned individually, but also enables the open-loop optimal controller to be used in closed-loop.

The optimal controller derived in Paper III can be sensitive to measurement noise or not be realizable. In Paper IV, it is suggested that the optimal controller is augmented with a second-order low-pass filter in order to cope with these deficiencies. Several guidelines and aspects regarding the choice of the filter time constant are discussed. For controllers with time-delays, it was shown how to adjust the delay when filtering is introduced.

In Paper V, it is shown how feedforward controllers that are linear in the parameters can be tuned by solving a convex optimization problem. Optimal proportional set-point weights are obtained by solving an optimization problem in which IAE is minimized for a large batch of processes. From the optimal parameters tuning rules for PI and PID controllers were derived. The rule for PID controllers apply to controllers with the derivative set-point weights equal to zero. The tuning rules were evaluated on 1320 different systems with five different PI and PID tuning methods. The evaluation shows that the tuning rules increase the performance by up to 45 % with a modest increase in actuation.

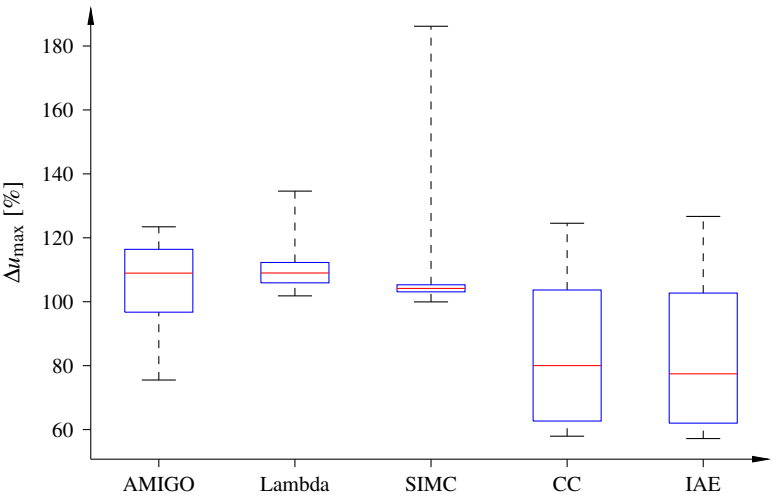

Figure 3.10 Increase in control signal peak when using the tuning rules (3.43) with PID controllers tuned with five different methods. The derivative set-point weight is $c = 0$. In each box, the central red mark is the median, the edges of the box are the 25th and 75th percentiles and the whiskers extend to the most extreme data points. CC denotes controllers obtained from the the convex-concave procedure and IAE denotes controllers obtained from the IAE minimization.

4

Future Work

"The more you found, the less you've been around."

— Josh Homme*, The Bronze*

The algorithm for design of SISO PID controllers presented in Paper I maximizes the integral gain, which is equivalent to minimizing IE. Since this cost function does not penalize oscillatory responses, it would be interesting to investigate if a similar optimization routine could be used to minimize IAE instead. Especially when the derivative term is used, it is known that the algorithm can produce controllers that give oscillatory responses. Minimizing IAE the oscillations could be reduced and thus the constraints of the curvature of the Nyquist curve could be removed.

The SISO design procedure is both fast and efficient and allows exploration of controller behavior subject to different constraints. It would be interesting, and straight-forward, to include constraints on the frequency function relating to the impact of noise on the control signal. Along with this, it would be of use to incorporate the design of measurement filters. Since the PID design method is fast, it could be used together with an iterative filter design, for instance the one presented in [Segovia et al., 2014], and thereby provide complete and useful controllers with good disturbance attenuation. The method for finding good set-point weights presented in Paper V could then be used to obtain a two-degree-of-freedom PID controller with good reference tracking.

The design method for MIMO PID controllers could be used in a similar fashion as its SISO counterpart. With an implementation that removes the relatively large overhead when CVX parses the problem, the algorithm would be very fast, which would allow for the same type of exploration as for the SISO PID. Additionally, the method could be used to compare the achievable robustness and performance of different controller structures. This could be used to investigate when diagonal MIMO controllers i.e., multiple SISO controllers, give comparable results to full structured MIMO PID controllers. Furthermore, the impact of the structure of each element in the MIMO PID controllers could be explored. Research related to the effects of, for instance PD controllers in the off-diagonal elements, could be performed using the presented method as a foundation.

It is much harder to design controllers for MIMO systems and it would therefore be convenient to have tuning rules for these in the same way as for SISO systems.

Developing such rules for a general MIMO plant is most definitely not tractable but simple systems with two inputs and two outputs, where each element is a FOTD model, could be a starting point. Another suggestion could be to use a batch of different process types in order to develop tuning rules based on FOTD approximations. However, this would require that such a batch is available or is developed.

The two methods for PID tuning could be adapted so that the optimization problem is solved for a set of processes. This could be used to find a controller that ensures robustness and has a reasonable performance for all the processes in the set.

An optimal controller is not necessarily the same as good a controller. This is somewhat reflected in the ISE minimizing feedforward controller presented in Paper III, which can be very aggressive. Some of that aggressiveness is counteracted by the filtering and considerations related to filtering that were presented in Paper IV. When designing feedforward controllers, from both reference and disturbances, noise and the control signal activity should be taken into account. These could be incorporated using the method presented in Paper V. Using a controller structure for the disturbance attenuating feedforward controller, similar to set-point weighting, these aspects could be investigated. The result would be simple feedforward controllers that have good noise rejection properties and reasonable actuation.

The work related to the tuning rules for the set-point weights have some apparent next steps. Set-point weights are often tuned so that there is no overshoot in the step response. Formulating and solving an optimization problem that minimizes IAE with the constraints on the overshoot could be applied to the AMIGO batch. The results could be used to formulate tuning rules that give small or no overshoot.

To find tuning rules for the derivative set-point weight c, the problem of how the ideal PID controller should be filtered has to be solved first. For a specific process with a given filtered PID controller, it is easy to find optimal set-point weights using the method in Paper V but developing tuning rules in the general case is harder. It was noticed during the work preceding that paper that the optimal c depends on the both the type of filter and also on the time-constants of the filter.

Bibliography

Alfaro, V. M., R. Vilanova, and O. Arrieta (2010). "Maximum sensitivity based robust tuning for two-degree-of-freedom proportional- integral controllers". *Industrial & Engineering Chemistry Research* **49**:11, pp. 5415–5423.

Alfaro, V. M., R. Vilanova, and O. Arrieta (2009). "Considerations on set-point weight choice for 2-DoF PID controllers". In: *IFAC International Symposium on Advanced Control on Chemical Process (ADCHEM 2009)*, pp. 12–15.

Andersen, M. S., J. Dahl, and L. Vandenberghe (2013). "CVXOPT: a Python package for convex optimization, version 1.1.6". *Available at cvxopt. org*.

Andersen, M., J. Dahl, Z. Liu, and L. Vandenberghe (2011). "Interior-point methods for large-scale cone programming". *Optimization for machine learning*, pp. 55–83.

Anonymous (1997). *Regleroptimering SSG-5253*. Tech. rep. Skogsindustriernas Teknik AB, Pulp and Paper Industries' Engineering Co, Sundsvall, Sweden.

Åström, K. J. and T. Hägglund (2001). "The future of PID control". *Control Engineering Practice* **9**:11, pp. 1163–1175.

Åström, K. J. and T. Hägglund (2004). "Revisiting the Ziegler-Nichols step response method for PID control". *Journal of Process Control* **14**:6, pp. 635–650.

Åström, K. J. and T. Hägglund (2006). *Advanced PID Control*. Instrumentation, Systems, and Automation Society.

Bao, J., J. Forbes, and P. McLellan (1999). "Robust multiloop PID controller design: a successive semidefinite programming approach". *Industrial & engineering chemistry research* **38**:9, pp. 3407–3419.

Bialkowski, W. (1993). "Dreams versus reality: a view from both sides of the gap: manufacturing excellence with come only through engineering excellence". *Pulp & Paper Canada* **94**:11, pp. 19–27.

Bianchi, F. D., R. J. Mantz, and C. F. Christiansen (2008). "Multivariable PID control with set-point weighting via BMI optimisation". *Automatica* **44**:2, pp. 472–478.

Boggs, P. T. and J. W. Tolle (1995). "Sequential quadratic programming". *Acta numerica* **4**, pp. 1–51.

Boyd, S. and L. Vandenberghe (2004). *Convex Optimization*. Cambridge University Press.

Brosilow, C. and B. Joseph (2002). *Techniques of Model-Based Control*. Prentice Hall PTR, pp. 221–239.

Chidambaram, M. (2000). "Set point weighted PI/PID controllers". *Chemical Engineering Communications* **179**:1, pp. 1–13.

Corless, R. M., G. H. Gonnet, D. E. Hare, D. J. Jeffrey, and D. E. Knuth (1996). "On the Lambert-W function". *Advances in Computational mathematics* **5**:1, pp. 329–359.

Dahlin, E. (1968). "Designing and tuning digital controllers". *Instruments and Control Systems* **41**:6, pp. 77–83.

Desborough, L. and R. Miller (2002). "Increasing customer value of industrial control performance monitoring–Honeywell's experience". In: *AIChE symposium series*. New York; American Institute of Chemical Engineers; 1998, pp. 169–189.

Dong, J. and C. Brosilow (1997). "Design of robust multivariable PID controllers via IMC". In: *Proceedings of the American Control Conference*. Vol. 5, pp. 3380–3384.

Galdos, G., A. Karimi, and R. Longchamp (2010). "H_∞ controller design for spectral MIMO models by convex optimization". *Journal of Process Control* **20**:10, pp. 1175–1182.

Garpinger, O. (2009). *Design of Robust PID Controllers with Constrained Control Signal Activity*. Licentiate Thesis ISRN LUTFD2/TFRT--3245--SE. Department of Automatic Control, Lund University, Sweden.

Garpinger, O. and T. Hägglund (2008). "A software tool for robust PID design". In: *Proceedings 17th IFAC World Congress*.

Grant, M. and S. Boyd (2008). "Graph implementations for nonsmooth convex programs". In: Blondel, V. et al. (Eds.). *Recent Advances in Learning and Control*. Lecture Notes in Control and Information Sciences. Springer-Verlag Limited, pp. 95–110.

Grimholt, C. and S. Skogestad (2015). "Improved optimization-based design of PID controllers using exact gradients". In: *12th International Symposium on Process Systems Engineering and 25th European Symposium on Computer Aided Process Engineering.*

Grimholt, C. and S. Skogestad (2012). "Optimal PI-control and verification of the SIMC tuning rule". In: *Proceedings of the IFAC Conference on Advances in PID Control, PID'12*.

Guzmán, J. L. and T. Hägglund (2011). "Simple tuning rules for feedforward compensators". *Journal of Process Control* **21**:1, pp. 92–102.

Hang, C. C., K. J. Åström, and W. K. Ho (1991). "Refinements of the Ziegler–Nichols tuning formula". In: *IEE Proceedings D (Control Theory and Applications)*. Vol. 138. 2. IET, pp. 111–118.

Khalil, H. K. and J. Grizzle (1996). *Nonlinear systems*. Prentice hall New Jersey.

Kristiansson, B. and B. Lennartson (2002). "Robust and optimal tuning of PI and PID controllers". *IEE Proceedings-Control Theory and Applications* **149**:1, pp. 17–25.

Larsson, P. O. and T. Hägglund (2011). "Control signal constraints and filter order selection for PI and PID controllers". In: *2011 American Control Conference*. San Francisco, California, USA.

Leva, A. and L. Bascetta (2006). "On the design of the feedforward compensator in two-degree-of-freedom controllers". *Mechatronics* **16**:9, pp. 533–546.

Lim, J. S. and Y. I. Lee (2008). "Design of discrete-time multivariable PID controllers via LMI approach". In: *Control, Automation and Systems, 2008. ICCAS 2008. International Conference on*. IEEE, pp. 1867–1871.

Lin, C., Q. G. Wang, and T. H. Lee (2004). "An improvement on multivariable PID controller design via iterative LMI approach". *Automatica* **40**:3, pp. 519–525.

Lipp, T. and S. Boyd (2014). *Extensions and variations on the convex-concave procedure*. To be published.

Lofberg, J. (2004). "YALMIP: a toolbox for modeling and optimization in MATLAB". In: *Computer Aided Control Systems Design, 2004 IEEE International Symposium on*. IEEE, pp. 284–289.

MATLAB (2012). *7.14.0.739 (R2012a)*. The MathWorks Inc., Natick, Massachusetts.

Monje, C. A., B. M. Vinagre, V. Feliu, and Y. Chen (2008). "Tuning and auto-tuning of fractional order controllers for industry applications". *Control Engineering Practice* **16**:7, pp. 798–812.

O'Dwyer, A. (2009). *Handbook of PI and PID Controller Tuning Rules, 3rd edn.* Imperial College Press London. ISBN: 978-1-85233-7092-5.

Padula, F. and A. Visioli (2011). "Tuning rules for optimal PID and fractional-order PID controllers". *Journal of Process Control* **21**:1, pp. 69–81.

Panagopoulos, H., K. J. Åström, and T. Hägglund (2002). "Design of PID controllers based on constrained optimisation". In: *IEE Proceedings on Control Theory and Applications*. Vol. 149. 1, pp. 32–40.

Research, C. (2012). *CVX: MATLAB software for disciplined convex programming, version 2.0*. `http://cvxr.com/cvx`.

Rivera, D. E., M. Morari, and S. Skogestad (1986). "Internal model control: PID controller design". *Industrial & engineering chemistry process design and development* **25**:1, pp. 252–265.

Rockafellar, R. T. (1970). *Convex analysis*. Princeton university press.

Rodríguez, C., J. L. Guzmán, M. Berenguel, and T. Hägglund (2013). "Generalized feedforward tuning rules for non-realizable delay inversion". *Journal of Process Control* **23**:9, pp. 1241–1250.

Rodríguez, C., J. L. Guzmán, M. Berenguel, and T. Hägglund (2014). "Optimal feedforward compensators for systems with right-half plane zeros". *Journal of Process Control* **24**:4, pp. 368–374.

Saeki, M., M. Ogawa, and N. Wada (2010). "Low-order H_∞ controller design on the frequency domain by partial optimization". *International Journal of Robust and Nonlinear Control* **20**, pp. 323–333.

Segovia, V. R., T. Hägglund, and K. J. Åström (2014). "Measurement noise filtering for common PID tuning rules". *Control Engineering Practice* **32**, pp. 43–63.

Sell, N. J. (1995). *Process control fundamentals for the pulp and paper industry*. Technical Association of the Pulp & Paper Industry. Chap. 6, pp. 221–226. ISBN: 0-899852-294-3.

Skogestad, S. (2003). "Simple analytic rules for model reduction and PID controller tuning". *Journal of Process Control* **13**:4, pp. 291–309.

Skogestad, S. and C. Grimholt (2012). "The SIMC method for smooth PID controller tuning". In: *PID Control in the Third Millennium*. Springer, pp. 147–175.

Taguchi, H. and M. Araki (2000). "Two-degree-of-freedom PID controllers–their functions and optimal tuning". *IFAC digital control: past, present and future of PID control*, pp. 5–7.

Tan, W., T. Chen, and H. Marquez (2002). "Robust controller design and PID tuning for multivariable processes". *Asian Journal of control* **4**:4, pp. 439–451.

Toh, K.-C., M. J. Todd, and R. H. Tütüncü (1999). "SDPT3—a MATLAB software package for semidefinite programming, version 1.3". *Optimization methods and software* **11**:1-4, pp. 545–581.

Tütüncü, R. H., K. C. Toh, and M. J. Todd (2003). "Solving semidefinite-quadratic-linear programs using SDPT3". *Mathematical programming* **95**:2, pp. 189–217.

Vilanova, R., V. Alfaro, and O. Arrieta (2012). "Simple robust autotuning rules for 2-DoF PI controllers". *ISA transactions* **51**:1, pp. 30–41.

Vilanova, R. (2008). "IMC based robust PID design: tuning guidelines and automatic tuning". *Journal of Process Control* **18**:1, pp. 61–70.

Vilanova, R. and A. Visioli (2012). *PID Control in the Third Millennium: Lessons Learned and New Approaches*. Springer.

Wang, Q., B. Zou, T. Lee, and Q. Bi (1997). "Auto-tuning of multivariable PID controllers from decentralized relay feedback". *Automatica* **33**:3, pp. 319–330.

Wood, R. and M. Berry (1973). "Terminal composition control of a binary distillation column". *Chemical Engineering Science* **28**:9, pp. 1707–1717.

Yuille, A. and A. Rangarajan (2003). "The concave-convex procedure". *Neural Computation* **15**:4, pp. 915–936.

Zheng, F., Q.-G. Wang, and T. H. Lee (2002). "On the design of multivariable PID controllers via LMI approach". *Automatica* **38**:3, pp. 517–526.

Zhou, K. and J. C. Doyle (1998). *Essentials of robust control*. Prentice-Hall.

Ziegler, J. and N. Nichols (1942). "Optimum settings for automatic controllers". *Transactions of the A.S.M.E* **5**:11, pp. 759–768.

Paper I

PID Design by Convex-Concave Optimization

Martin Hast Karl Johan Åström Bo Bernhardsson Stephen Boyd

Abstract

This paper describes how PID controllers can be designed by optimizing performance subject to robustness constraints. The optimization problem is solved using convex-concave programming. The method admits general process descriptions in terms of frequency response data and it can cope with many different constraints. Examples are presented and some pitfalls in optimization are discussed.

1. Introduction

Controller design is a rich problem because it requires that many factors related to performance and robustness are taken into account [Vilanova and Visioli, 2012]. Many features can be captured by formulating the design problem as a constrained optimization problem [K. J. Åström and Hägglund, 2006; Garpinger, 2009; Karimi and Galdos, 2010; Karimi, Kunze, et al., 2007; Panagopoulos et al., 2002; Sadeghpour et al., 2012].

Convex programming [Boyd and Vandenberghe, 2004] is a powerful optimization technique, which has guaranteed convergence and efficient algorithms that have been packaged in easy-to-use tools [Research, 2012; Grant and Boyd, 2008]. There is a modification called convex-concave optimization which admits nonconvex criteria and constraints [Boyd, 2013; Yuille and Rangarajan, 2003]. There is in general no guarantee of convergence to a global minimum but the algorithms converge to a saddle point or local minimum.

In this paper we will consider convex-concave programming for design of PID controllers. Following the ideas in [K. J. Åström, Panagopoulos, et al., 1998; Panagopoulos et al., 2002] we consider maximization of integral gain subject to robustness constraints on the sensitivities and other constraints. Both disturbance attenuation and response time are inversely proportional to integral gain. Unconstrained maximization of integral gain does not necessarily lead to good controllers because the responses may be highly oscillatory.

PID controllers have been designed using optimization earlier with similar problem formulations [K. J. Åström and Hägglund, 2006; Garpinger, 2009; Panagopoulos et al., 2002]. The proposed method is similar to M-constrained Integral Gain Optimization, MIGO [Hägglund and K. Åström, 2002; Hägglund and K. Åström, 2004], but it admits more flexible constraints and the computations are simpler. Similar approaches using linear programming can be found in [Karimi and Galdos, 2010; Karimi, Kunze, et al., 2007; Sadeghpour et al., 2012], and [Galdos et al., 2010] for MIMO systems.

The advantages of convex-concave optimization are that the software package CVX [Research, 2012; Grant and Boyd, 2008] allows for very compact programs, and many different criteria and constraints can be accommodated. The technique can also be extended to more complicated systems.

2. PID Design

Consider a closed loop system with PI or PID control. The process transfer function is $P(s)$, and controller transfer functions are

$$C_{\text{PI}}(s) = k_p + \frac{k_i}{s}, \quad C_{\text{PID}}(s) = k_p + \frac{k_i}{s} + k_d s,$$

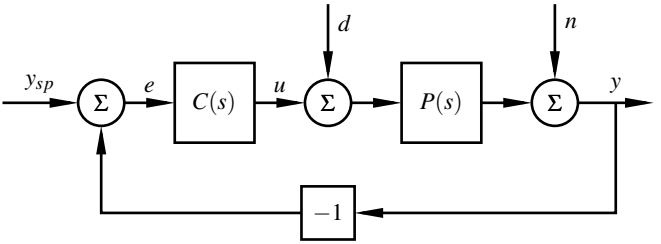

Figure 1. Block diagram.

where k_p, k_i, and k_d are the controller parameters. A block diagram of the system is shown in Fig. 1. Measurement noise can be reduced by a second order filter with the transfer function

$$G_f(s) = \frac{1}{1 + sT_f + s^2T_f^2/2}, \tag{1}$$

where T_f is the filter time constant. A first order filter may suffice for PI control but a second order filter is required to ensure roll-off when derivative action is used; see [K. J. Åström and Hägglund, 2006].

The combinations of the controllers and the filter transfer functions are denoted by

$$C(s) = C_{\text{PI}}(s)G_f(s), \qquad C(s) = C_{\text{PID}}(s)G_f(s).$$

Using this representation ideal controllers can be designed for the augmented plant $P(s)G_f(s)$.

A good controller should give a closed-loop system with a fast response to command signals y_{sp}, load disturbances d should be well attenuated and measurement noise should not generate too large control signals. In addition the closed-loop system should be insensitive to variations in the dynamics of the process $P(s)$.

Common criteria for control performance are the integrated error and the integrated absolute error

$$\text{IE} = \int_0^\infty e(t)dt, \qquad \text{IAE} = \int_0^\infty |e(t)|\,dt,$$

where e is the control error due to a unit step load disturbance applied at the process input or the process output or a unit step change in the command signal. The quantities IE and IAE are good measures of load disturbance attenuation for controllers with integral action. For systems that are well damped, the two criteria are approximately the same. It can be shown [K. J. Åström and Hägglund, 2006] that

$$\text{IE} = \frac{1}{k_i} \tag{2}$$

for a unit step disturbance and $1/(P(0)k_i)$ for a set-point step. Minimizing IE does not guarantee that the responses are satisfactory because the responses may be highly oscillatory. Combined with robustness constraints, minimization of IE may, however, give controllers with good properties [K. J. Åström and Hägglund, 2006]. That this is not always the case will be investigated in Example 4.

The sensitivity function and the complementary sensitivity functions are defined as

$$S(s) = \frac{1}{1+L(s)}, \quad T(s) = \frac{L(s)}{1+L(s)} \tag{3}$$

where $L(s) = P(s)C(s)$ is the loop transfer function. Robustness to process uncertainty can be captured by constraints on the maximum sensitivities M_s and M_t

$$M_s = \max_{\omega} |S(i\omega)|, \quad M_t = \max_{\omega} |T(i\omega)|. \tag{4}$$

Such constraints have nice geometric interpretations in the Nyquist plot of the loop transfer function. Requirements on the sensitivities mean that the Nyquist plot is outside circles; see Figure 4.15 in [K. J. Åström and Hägglund, 2006]. Process uncertainty can be represented by circles around the nominal loop transfer function. These constraints are well captured by convex-concave programming as will be shown in Sec. 4.1 and 4.2.

It is important that the control actions generated by measurement noise are not too large. The fluctuations in the control signal can be computed from the transfer functions of the process and the controller and a characterization of the measurement noise, like its spectral density. Such detailed information is rarely available for PI or PID control and we will therefore use simpler measures.

The transfer function from measurement noise n to controller output for the closed loop system is

$$G_{un}(s) = C(s)S(s). \tag{5}$$

The transfer function G_{un} can be characterized by its largest value

$$M_{un} = \max_{\omega} |G_{un}(i\omega)|. \tag{6}$$

To ensure that measurement noise does not generate too large control actions we can introduce constraints on the transfer function G_{un}. For processes with $P(0) \neq 0$ and controllers with integral action we have $G_{un}(0) = 1$, and hence $M_{un} \geq 1$.

An approximate expression for M_{un} is the high-frequency controller gain k_d/T_f where T_f is the parameter of the noise filter (1). The parameter T_f can be determined as a compromise between noise injection and load disturbance attenuation. For a given T_f the condition is then a constraint on the derivative gain k_d which fits well into convex-concave optimization.

The constraints on noise injection can also be dealt with in this framework, design of the filter time constant can be dealt with iteratively as described in [Romero Segovia et al., 2013].

3. Convex-concave Optimization

Convex-concave optimization [Boyd, 2013; Yuille and Rangarajan, 2003] is a procedure for problems where the optimization criterion and constraints are written as a difference between two convex functions,

$$\begin{array}{ll} \text{minimize} & f_0(x) - g_0(x) \\ \text{subject to} & f_i(x) - g_i(x) \leq 0 \quad i = 1, \ldots, m \end{array}$$

where f_i and g_i are convex functions. This is not a convex problem since $-g_i$ is concave. The convex functions are left unchanged and all concave functions are replaced by linearizations around the current solution point x_k, i.e., replace $f(x) - g(x)$ by

$$\hat{f}(x) = f(x) - g(x_k) - \nabla g(x_k)^T (x - x_k). \tag{7}$$

This convex approximation is an upper bound on the function being approximated. It follows that the resulting convex constraints are more conservative than the original: the feasible set will be a convex subset of the original feasible set. The new problem can be solved efficiently to produce a new feasible point x_{k+1}, and the procedure is repeated. Since the approximation is conservative, the new iterate is guaranteed to be feasible, and not to have larger objective value.

The iterative procedure converges to a saddle point or a local minimum [Yuille and Rangarajan, 2003]. Even though there is no guarantee of convergence to a global minimum, experience has shown the method to often be effective in producing good solutions.

4. Optimization

The constraints presented in the following sections will be defined frequency-wise and the considered optimization problems will have an infinite number of constraints. In order to obtain a tractable optimization problem, the problems are solved over a grid of frequency points. Since the problem is convex, a large number of constraints can efficiently be handled and a fine grid can therefore be used.

Since the convex-concave procedure is iterative there is a need for an initial controller. The initial controller needs to stabilize the system. For a stable plant it will suffice to choose the initial controller parameters to be zero but care must be taken if the process is open loop unstable; see Example 3.

The objective of the optimization is to minimize IE under robustness constraints. From (2) it can easily be seen that minimizing IE is equivalent to

$$\text{maximize } k_i. \tag{8}$$

4.1 Circle constraints

Consider a circle with center c and radius r. The constraint that the Nyquist plot should lie outside the circle is equivalent to

$$r - |L - c| = r - g(\alpha) \leq 0, \tag{9}$$

where $\alpha = \begin{pmatrix} k_p & k_i & k_d \end{pmatrix}^T$. The inequality constraint (9) is a convex-concave constraint since $g(\alpha)$ is a convex function. Using (7) on (9) we obtain

$$\hat{f}(\alpha) = r - \Re\left(\frac{(L_k - c)^*}{|L_k - c|}(L - c)\right) \leq 0$$

where $\Re$ and $*$, denotes the real part and complex conjugate respectively, and L_k, is the open-loop transfer function with the controller parameters from the last iteration.

The classical robustness constraints given by the sensitivity functions imply that $L(i\omega)$ should be outside two circles with centres in $c_s = -1$ and $c_t = -M_t^2/(M_t^2 - 1)$ and radii $r_s = 1/M_s$ and $r_t = M_t/(M_t^2 - 1)$ respectively; see [K. J. Åström and Hägglund, 2006].

4.2 Process uncertainty

The robustness constraints can be generalized to settings with process uncertainties. Consider a process $\tilde{P}$ with uncertainties such that for each frequency point the Nyquist plot is known to lie within a circle with a frequency dependent, radius ρ, i.e.,

$$\tilde{P}(i\omega) = P(i\omega) + \Delta(i\omega), \quad |\Delta(i\omega)| \leq \rho(i\omega).$$

A constraint specifying that the Nyquist plot should lie outside a circle with centre c and radius r is then

$$r - |PC + \Delta C - c| \leq 0. \tag{10}$$

Furthermore,

$$\inf_{|\Delta| \leq \rho} \{|1 + PC + \Delta C|\} = \max(|1 + PC| - \rho|C|, 0)$$

where the equality follows from the triangle inequality and by choosing the magnitude as $|\Delta| = \min\left(\rho, \frac{|1+PC|}{|C|}\right)$ and the phase as $\arg(\Delta) = \arg(1 + PC) - \arg(C)$. Hence, (10) can be formulated as

$$r - |L - c| + \rho|C| \leq 0 \tag{11}$$

for which the concave part can be approximated in the same way as the circle constraints.

4.3 Curvature constraints

Minimization of IE may sometimes give Nyquist curves with very small curvature; see Example 4. A constraint on the curvature can be expressed as a convex-concave constraint. To obtain this constraint the loop transfer function is decomposed as $L(i\omega) = x(\omega) + iy(\omega)$. Furthermore, let dots denote derivatives of the corresponding variables with respect to ω. The curvature of L at a frequency point ω is then given by

$$\kappa = \frac{\dot{x}\ddot{y} - \dot{y}\ddot{x}}{(\dot{x}^2 + \dot{y}^2)^{3/2}}.$$

To avoid kinks in the Nyquist plot we introduce a constraint limiting its curvature by γ. The constraint can then be formulated as $\Im(\dot{L}^*\ddot{L}) - \gamma|\dot{L}|^3 \leq 0$, equivalent to

$$\alpha^T Q \alpha - \gamma |Z\alpha|^3 \leq 0 \tag{12}$$

where Q is an indefinite, rank one matrix. Since Q is rank one, it has only one eigenvalue different from zero, whose sign determines whether the matrix is positive semi-definite or negative semi-definite. Hence, the constraint given by (12) fits nicely into the convex-concave framework and can be written as

$$\alpha^T Q_p \alpha - \alpha^T Q_n \alpha - \gamma |Z\alpha|^3 \leq 0. \tag{13}$$

where either Q_p or Q_n is zero and where the nonzero matrix is positive semi-definite. Utilizing (7), a convex approximation of (13) is given by

$$\begin{aligned} \hat{f}(\alpha) &= \alpha^T Q_p \alpha + A_k \alpha + b_k \leq 0 \\ A_k &= -\alpha_k^T Q_n - 3\gamma |Z\alpha_k| \alpha_k^T Z^H Z \\ b_k &= \alpha_k Q_n \alpha_k + 2\gamma |Z\alpha_k|^3 \end{aligned} \tag{14}$$

where H denotes the Hermitian transpose.

5. Examples

The proposed design method will be illustrated by four examples. The examples share some common features. In all examples the integral gain is maximized. The robustness constraints are $M_s = M_t = 1.4$ unless otherwise stated. A grid of 1000 logarithmically spaced frequencies between $\omega_{min} = 10^{-2}$ [rad/s] and $\omega_{max} = 10^2$ [rad/s] is used in all examples. The initial controller parameters are zero unless otherwise stated. For the examples provided, the optimization algorithm converges to a solution within seven iterations excluding the optimization problem with curvature constraints in Example 4, for which 11 iterations were needed.

Example 1. Heat conduction

An advantage of the method is that it can be applied to processes where the transfer function is only given as a frequency response or processes described by partial differential equations. We illustrate with a simple example of a process representing heat conduction where the process has the transfer function

$$P(s) = \mathrm{e}^{-\sqrt{s}}. \tag{15}$$

Maximizing integral gain with constraints on the maximum values of the sensitivity functions using the convex-concave procedure gives the following PI and PID controllers

$$C_{\mathrm{PI}} = 2.94 + \frac{11.54}{s}, \quad C_{\mathrm{PID}}(s) = 7.40 + \frac{48.25}{s} + 0.46s.$$

Some performance measures are given in Table 1. The Nyquist plots of the loop transfer functions are shown in Fig. 2, red dashed lines for PI control and blue solid lines for PID control. The unit step load disturbance responses are shown in Fig. 3. Notice that IE is less than IAE because the responses are oscillatory. The criterion IE does not penalize oscillatory responses, on the contrary, an oscillatory response may give lower IE. The PID controller has significantly better performance than the PI controller, which is not surprising because the process (15) has lag-dominated dynamics.

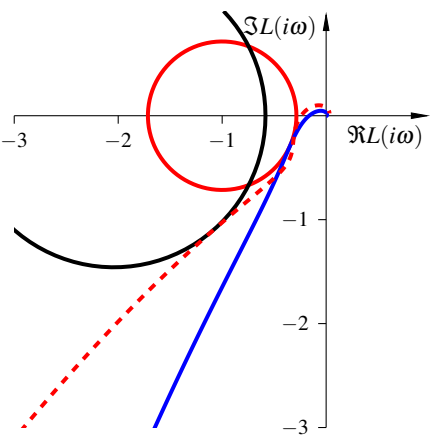

Figure 2. Nyquist plots of the loop transfer functions for PI control (red dashed lines) and PID control (blue solid lines) of the process $P(s) = \mathrm{e}^{-\sqrt{s}}$ in Example 1. The robustness constraints $M_s = 1.4$ and $M_t = 1.4$ are shown in red and black circles.

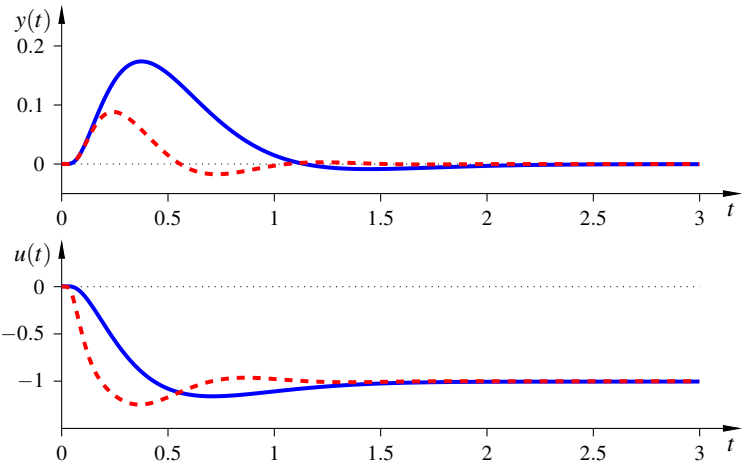

Figure 3. Responses to a unit step load disturbances for PI (red dashed) and PID control (blue solid) of the process $P(s) = \mathrm{e}^{-\sqrt{s}}$ in Example 1.

Table 1. Controller parameters and performance measures for the system in Example 1 and Example 2

Optimization	k_i	IE	IAE	$y_{\max}$
Ex. 1 PI	11.54	8.67e-2	9.98e-2	17.83e-2
Ex. 1 PID	48.25	2.07e-2	3.14e-2	8.84e-2
Ex. 2 PI	7.43	13.46e-2	14.92e-2	19.45e-2
Ex. 2 PID	26.81	3.73e-2	4.63e-2	10.57e-2

Example 2. Explicit Process Uncertainty

Process uncertainty can be accounted for explicitly as discussed in Sec. 4.2. To illustrate this we will consider the same system as in Example 1 but we will now assume that the process transfer function has a relative uncertainty of 20%. Maximizing integral gain with constraints on the maximum values of the sensitivity functions using the convex-concave procedure gives the following PI and PID controllers

$$C_{\mathrm{PI}} = 2.37 + \frac{7.43}{s}, \quad C_{\mathrm{PID}} = 5.74 + \frac{26.81}{s} + 0.36s.$$

Some performance measures are given in Table 1. The Nyquist plots of the loop transfer functions are shown in Fig. 4, red dashed lines for PI control and blue solid lines for PID control. The unit step load disturbance responses are shown in Fig. 5.

Comparing the Nyquist plots in Fig. 2 and Fig. 4 we can see that adding un-

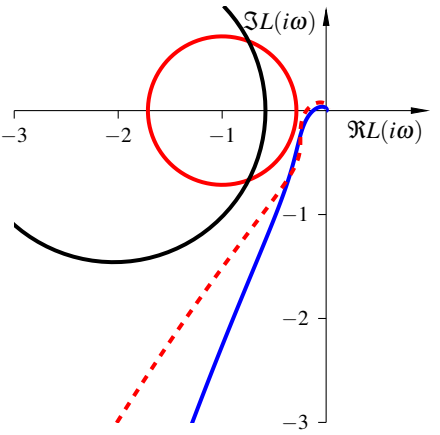

Figure 4. Nyquist plots of the loop transfer functions for PI control (red dashed lines) and PID control (blue solid lines) of the process $P(s) = \mathrm{e}^{-\sqrt{s}}$ with 20% relative uncertainty in Example 2. The robustness constraints $M_s = 1.4$ and $M_t = 1.4$ are shown in red and black circles.

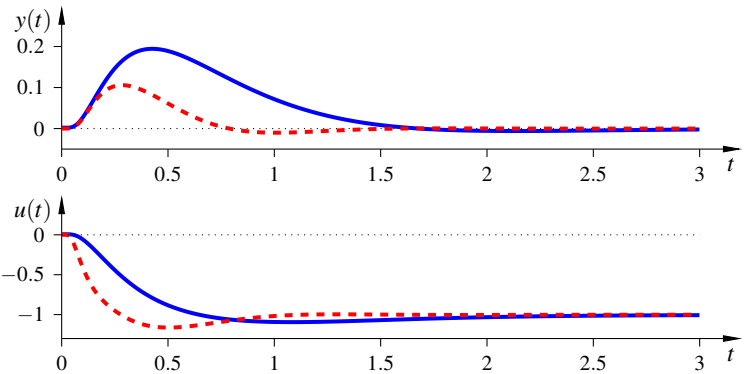

Figure 5. Responses to a unit step load disturbances for PI (red dashed) and PID control (blue solid) of the process $P(s) = \mathrm{e}^{-\sqrt{s}}$ with 20% relative uncertainty in Example 2.

certainties in the process model moves the Nyquist curve further away from the robustness circles in such a way that the robustness constraints are satisfied for all uncertainties in the uncertainty set. The resulting controllers obtained for the uncertain processes are less aggressive, the time responses are more sluggish and hence, the performance measures deteriorate compared with the controllers obtained in Example 1.

Example 3. Unstable Process

For unstable processes it is necessary to choose the initial controller parameters so that the loop transfer functions has the correct winding number. If the initial controller stabilizes the plant and satisfies the constraints, the winding number is preserved in the iterations if the frequency points are sufficiently dense.

To illustrate this consider PI control of an unstable process with the transfer function

$$P(s) = \frac{1}{(s-1)(1+0.1s)}.$$

The initial stabilizing controller is chosen as $C_0(s) = 6 + \frac{1}{s}$. This controller gives a loop transfer function that satisfies the encirclement condition and the constraints on the sensitivity functions as is shown in Fig. 6. The integrated error with the initial controller is $\mathrm{IE}_0 = 1$. Maximizing integral gain with constraints on the maximum values of the sensitivity functions using the convex-concave procedure gives the PI controller

$$C_{\mathrm{PI}}(s) = 4.67 + \frac{1.76}{s}$$

with $\mathrm{IE} = 0.57$. Responses of the closed-loop system to a unit load disturbance at the process input are shown in Fig. 7. The responses are well damped and, hence, IE is equal to IAE. The Nyquist plots for the loop transfer functions can be seen in Fig. 6.

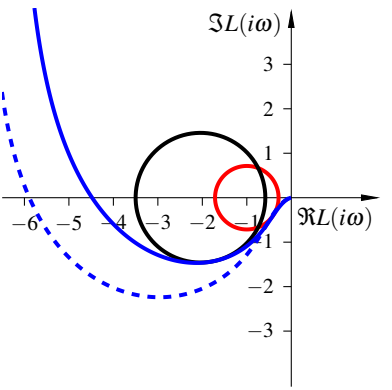

Figure 6. Nyquist plots of the loop transfer functions for PI control of the unstable process $P(s) = \dfrac{1}{(s-1)(0.1s+1)}$ in Example 3. The robustness constraints $M_s = 1.4$ and $M_t = 1.4$ are shown in red and black circle respectively. The loop transfer function corresponding to the initial controller parameters $k_p = 6$ and $k_i = 1$ are shown in dashed curve. The loop transfer function corresponding to the optimal controller parameters $k_p = 4.67$ and $k_i = 1.76$ are shown in solid curves.

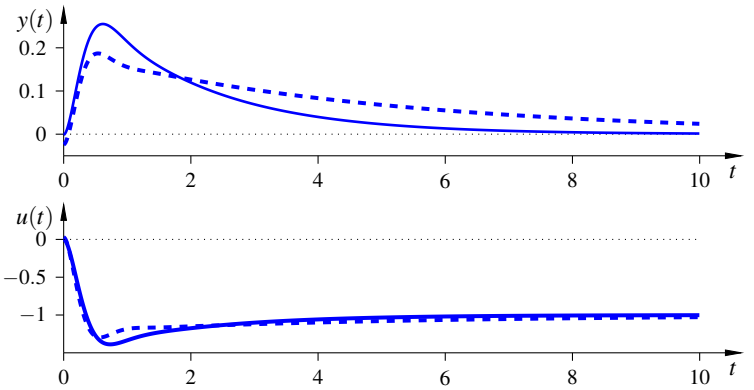

Figure 7. Responses to a unit step load disturbances for PI control of the unstable system $P(s) = \dfrac{1}{(s-1)(1+0.1s)}$ in Example 3. The responses corresponding to the initial controller parameters $k_p = 6$ and $k_i = 1$ are shown in dashed curves. The responses corresponding to the optimal controller parameters $k_p = 4.67$ and $k_i = 1.76$ are shown in solid curve

Example 4. Nyquist plots with kinks

The previous examples show that minimization of IE subject to robustness constraints give controllers with good properties. However, minimizing IE does not, in general, guarantee well-behaved closed loop system; see [K. J. Åström and Hägglund, 2006]. The reason is that minimization of IE may result in closed loop systems with poorly damped oscillations. Intuitively we may expect that the tendency for oscillatory responses is counteracted by the robustness constraint. The following example shows that difficulties may indeed occur. Consider a process with the transfer function

$$P(s) = \frac{1}{(s+1)^3}. \tag{16}$$

Assuming that we only impose a constraint on the sensitivity function, maximizing integral gain using the convex-concave procedure gives the PID controller

$$C_{\text{IE}}(s) = 3.31 + \frac{6.62}{s} + 6.26s. \tag{17}$$

The poles for the closed-loop system are located in $s = -1.25 \pm 1.73i, -0.25 \pm 0.87i$ which suggests that the step responses will be poorly damped. The blue curve in Fig. 8 shows that the Nyquist plot of the loop transfer function has a kink and the corresponding time responses in Fig. 9 are highly oscillatory. This behaviour is counter-intuitive because we may expect that strong robustness constraints may induce closed loop systems with good damping. This is indeed the case for PI control but not for PID control [K. J. Åström, Panagopoulos, et al., 1998; Panagopoulos

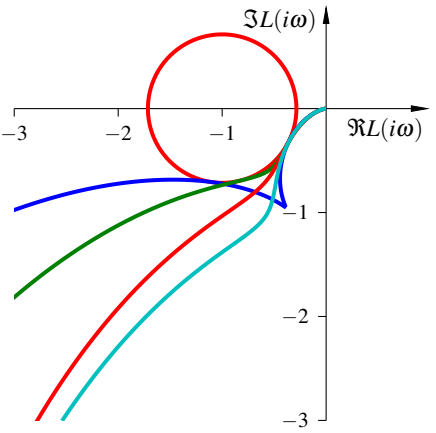

Figure 8. Nyquist plots of the loop transfer functions for PID control of the process $P(s) = (s+1)^{-3}$ in Example 4. All curves are obtained from optimization problems where the sensitivity constraint was $M_s = 1.4$. The blue curve is obtained with no additional constraints. The green curve is obtained with the additional constraint $k_d \leq 3.82$. The red curve is obtained with the additional constraint that the curvature of the Nyquist plot is less than M_s. The cyan curve is obtained by minimizing IAE using the algorithm in [Garpinger and Hägglund, 2008].

et al., 2002]. The kink will be a little smaller with tighter robustness constraints but it remains even if we require that sensitivities are less than 1.1. The problem is discussed in [K. J. Åström and Hägglund, 2006] where it is labelled the *derivative cliff*; see Figure 6.24 in [K. J. Åström and Hägglund, 2006]. The problem can be avoided in several different ways, one way is to change the criterion to IAE; see [Garpinger and Hägglund, 2008]. Another is to constrain the optimization so that the edges are avoided. In the MIGO design the edges are avoided by restricting the derivative gain. Other attempts to constrain the controller have also been proposed, the derivative gain has been restricted to the largest gain of a PD controller that maximizes proportional gain [K. J. Åström and Hägglund, 2006] and the constraint $T_i = 4T_d$ is proposed in [Wallén et al., 2002]. When using convex-concave optimization the problem can be avoided by introducing a curvature constraint on the loop transfer function.

The Nyquist plots in Fig. 8 and the time responses in Fig. 9 also show results for two modified controllers. The green curve shows a PID controller where the derivative gain is restricted to the derivative gain of a PD controller that maximizes proportional gain with the robustness constraint. The red curve shows results where the curvature of $L(i\omega)$ is restricted to M_s.

Controller parameters and performance criteria are summarized in Table 2. It is clear that the controller obtained by minimizing IE subject to robustness constraints

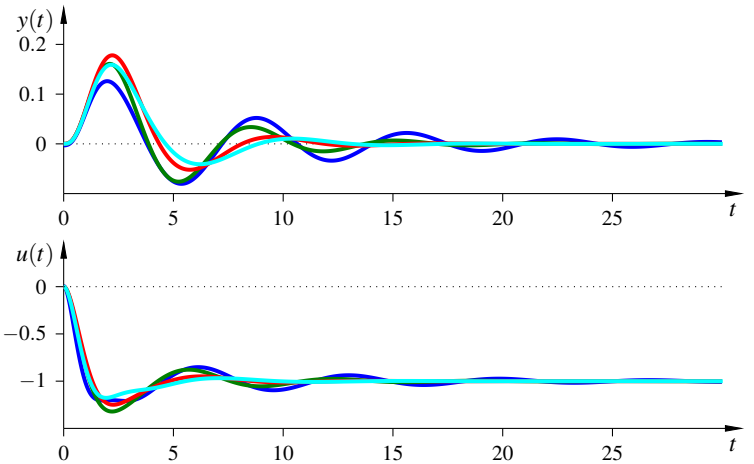

Figure 9. Responses to unit step load disturbances for the process $P(s) = (s+1)^{-3}$ in Example 4. All curves are obtained from optimization problems where the sensitivity constraint was $M_s = 1.4$. The blue curve is obtained with no additional constraints. The green curve is obtained with the additional constraint $k_d \leq 3.82$. The red curve is obtained with the additional constraint that the curvature of the Nyquist plot is less than M_s. The cyan curve is obtained by minimizing IAE using the algorithm in [Garpinger and Hägglund, 2008].

Table 2. Controller parameters and performance measures for the systems Example 4.

Optimization	k_p	k_i	k_d	IE	IAE	$y_{\max}$
IE-min.	3.31	6.62	6.26	0.15	0.74	0.126
k_d-limit	3.71	4.49	3.82	0.22	0.61	0.161
κ-limit	3.61	3.20	3.34	0.31	0.57	0.178
IAE-min.	3.81	3.33	4.25	0.30	0.53	0.159

gives oscillatory and unsatisfactory behaviour. By limiting the derivative part the oscillations are damped and IAE is reduced. The drawback with this approach is that two optimizations need to be performed. However, the performance, in terms of IAE, is better for the controller obtained by limiting the curvature of the Nyquist plot. The approach in which the curvature is limited, renders a controller whose performance is close to the controller obtained using the algorithms from [Garpinger and Hägglund, 2008] which minimizes IAE subject to constraints on the sensitivity functions.

6. Conclusions

We have shown that design of PID controllers can be captured in a format that is well suited for convex-concave optimization. The criterion is to minimize IE or equivalently to maximize integral gain subject to robustness constraints. To avoid oscillatory responses we have also introduced a constraint on the curvature of the Nyquist curve of the loop transfer function. The optimization problems are conveniently solved using CVX, the code is very compact and additional convex-concave constraints can be included.

Since the PID controller only has three parameters the design problem can be solved by gridding. This approach [Nordfeldt and Hägglund, 2005] has the advantage that any criteria and constraints can be used. However, the complexity of gridding increases dramatically with the number of controller parameters. Convex optimization does not suffer from this difficulty and it can therefore also be applied to other controller structures such as multi-variable PID or fixed higher-order controllers.

Acknowledgment

This work was supported by the Swedish Research Council through the LCCC Linnaeus Center, the ELLIIT Excellence Center and the Swedish Foundation for Strategic Research through PICLU.

References

Åström, K. J., H. Panagopoulos, and T. Hägglund (1998). "Design of PI controllers based on non-convex optimization". *Automatica* **34**:5, pp. 585–601.

Åström, K. J. and T. Hägglund (2006). *Advanced PID Control*. Instrumentation, Systems, and Automation Society.

Boyd, S. and L. Vandenberghe (2004). *Convex Optimization*. Cambridge University Press.

Boyd, S. (2013). *Sequential Convex Programming*. *Lecture note* `http://www.stanford.edu/class/ee364b/lectures/seq_slides.pdf`.

Galdos, G., A. Karimi, and R. Longchamp (2010). "H_∞ controller design for spectral MIMO models by convex optimization". *Journal of Process Control* **20**:10, pp. 1175–1182.

Garpinger, O. (2009). *Design of Robust PID Controllers with Constrained Control Signal Activity*. Licentiate Thesis ISRN LUTFD2/TFRT--3245--SE. Department of Automatic Control, Lund University, Sweden.

Garpinger, O. and T. Hägglund (2008). "A software tool for robust PID design". In: *Proceedings 17th IFAC World Congress*.

Grant, M. and S. Boyd (2008). "Graph implementations for nonsmooth convex programs". In: Blondel, V. et al. (Eds.). *Recent Advances in Learning and Control*. Lecture Notes in Control and Information Sciences. Springer-Verlag Limited, pp. 95–110.

Hägglund, T. and K. Åström (2002). "Revisiting the Ziegler-Nichols tuning rules for PI control". *Asian Journal of Control* **4**:4, pp. 364–380.

Hägglund, T. and K. Åström (2004). "Revisiting the Ziegler-Nichols step response method for PID control". *Journal of Process Control* **14**:6, pp. 635–650.

Karimi, A. and G. Galdos (2010). "Fixed-order H_∞ controller design for nonparametric models by convex optimization". *Automatica* **46**:8, pp. 1388–1394.

Karimi, A., M. Kunze, and R. Longchamp (2007). "Robust controller design by linear programming with application to a double-axis positioning system". *Control Engineering Practice* **15**:2, pp. 197–208.

Nordfeldt, P. and T. Hägglund (2005). "Design of PID controllers for decoupled multi-variable systems". In: *Proc. 16th IFAC World Congress*. Prague, Czech Republic.

Panagopoulos, H., K. J. Åström, and T. Hägglund (2002). "Design of PID controllers based on constrained optimisation". In: *IEE Proceedings on Control Theory and Applications*. Vol. 149. 1, pp. 32–40.

Research, C. (2012). *CVX: MATLAB software for disciplined convex programming, version 2.0*. `http://cvxr.com/cvx`.

Romero Segovia, V., T. Hägglund, and K. Åström (2013). "Noise filtering in PI and PID control". In: *American Control Conference*.

Sadeghpour, M., V. de Oliveira, and A. Karimi (2012). "A toolbox for robust PID controller tuning using convex optimization". In: *IFAC Conference in Advances in PID Control*.

Vilanova, R. and A. Visioli (2012). *PID Control in the Third Millennium: Lessons Learned and New Approaches*. Springer.

Wallén, A., K. Åström, and T. Hägglund (2002). "Loop-shaping design of PID controllers with constant Ti/Td ratio". *Asian Journal of Control* **4**:4, pp. 403–409.

Yuille, A. and A. Rangarajan (2003). "The concave-convex procedure". *Neural Computation* **15**:4, pp. 915–936.

Paper II

MIMO PID Tuning via Iterated LMI Restriction

Stephen Boyd Martin Hast Karl Johan Åström

Abstract

We formulate multi-input multi-output (MIMO) proportional-integral-derivative (PID) controller design as an optimization problem that involves nonconvex quadratic matrix inequalities. We propose a simple method that replaces the nonconvex matrix inequalities with a linear matrix inequality (LMI) restriction, and iterates to convergence. This method can be interpreted as a matrix extension of the convex-concave procedure, or as a particular majorization-minimization (MM) method. Convergence to a local minimum can be guaranteed. While we do not know that the resulting controller is globally optimal, the method works well in practice, and provides a simple automated method for tuning MIMO PID controllers. The method is readily extended in many ways, for example to the design of more complex, structured controllers.

Submitted to the *International Journal of Robust and Nonlinear Control*.

1. Introduction

Single-input single-output (SISO) proportional-integral-derivative (PID) control is the automatic control scheme most widely used in practice, with a long history going back at least 250 years; see [Åström and Hägglund, 2006, §1.4]. It has only three parameters to tune, and achieves reasonable or good performance on a wide variety of plants. The effect of the tuning parameters (or gains) on the closed-loop performance are well understood, and there are well known simple rules for tuning these parameters; see, e.g., [Åström and Hägglund, 2006, Chap. 6] or [Luyben, 1986]. Systems for automatically tuning SISO PID controllers have been developed, and are available in commercial controllers [Åström and Hägglund, 2006; Garpinger and Hägglund, 2008; Garpinger, Hägglund, and Åström, 2012; Vilanova and Visioli, 2012]. The authors of this paper recently developed yet another SISO PID tuning method in [Hast et al., 2013], which is a precursor for the method described in this paper.

SISO PID controllers have been used for multiple-input multiple-output (MIMO) plants for many years. This is generally done by pairing inputs (actuators) and outputs (sensors), and connecting them with SISO PID controllers. These SISO PID controllers can be tuned one at a time (in 'successive loop closure') using standard SISO PID tuning rules. For MIMO plants that are already reasonably well decoupled, multi-loop SISO PID design can work well. Unlike SISO PID design, however, MIMO PID design is more complex; the SISO loops have to be chosen carefully, and then tuned the correct way in the correct order.

An alternative to multi-loop SISO PID control is to design one MIMO PID controller, which uses matrix coefficients, all at once. Such a controller potentially uses all sensors to drive all actuators, but it is possible to specify a simpler structure by imposing a sparsity constraint on the controller gain matrices. Like SISO PID controllers for SISO plants, MIMO PID controllers can achieve very good performance on a wide variety of MIMO plants, even when the plant dynamics are quite coupled. The challenge is in tuning MIMO PID controllers, which require the specification of three matrices, each with a number of entries equal to the number of plant inputs times the number of plant outputs. For example, a MIMO PID controller for a plant with 4 inputs and 4 outputs requires the specification of up to 48 parameters. This would be very difficult, if not impossible, to tune by hand one parameter at a time. Hand tuning a MIMO PID controller with 10 inputs and 10 outputs would be impossible in practice.

In this paper we describe a method for designing MIMO PID controllers. The method is based on solving a small number of convex optimization problems, specifically semidefinite programs (SDPs), which can be done efficiently. Our method is a local optimization method, and we cannot guarantee that it finds the globally optimal controller parameter values. On many examples, however, the method seems to work very well.

We first describe a basic form for the method. We impose constraints on the sen-

sitivity and complementary sensitivity transfer functions, which guarantees closed-loop stability and a MIMO stability margin, and also a limit on actuator effort. The objective is to minimize the low-frequency sensitivity of the closed-loop system, which is a MIMO analog of maximizing the integral gain in the SISO case. In a later section we describe a number of generalizations of the method.

There is an enormous literature on automated SISO PID tuning, and a very large literature on MIMO PID tuning (see, e.g., [Åström, Panagopoulos, et al., 1998] and its references, or [Panagopoulos et al., 2002; Saeki, Kashiwagi, et al., 2007]) including some methods that are very close to the one we describe, and others that are close in spirit. We give a more detailed technical analysis of other methods in §6.4, after describing the details of our method. But we mention here some earlier work that is very closely related to ours. In [C. Lin et al., 2004], the authors also form a linear matrix inequality (LMI) based restriction of a problem with nonconvex quadratic matrix inequalities, and iterates to convergence. Another previous work that is close in spirit to ours is [Bianchi et al., 2008], which formulates the design problem as one involving bilinear matrix inequalities (BMIs), which are in turn solved (approximately) by iteratively solving a set of SDPs. (The connection between our method and BMI formulations will be discussed in §6.4.) The closest prior work appears in [Saeki, Ogawa, et al., 2010], which takes a very similar approach to ours. The authors consider MIMO PID design, using frequency-domain specifications, and their algorithm involves LMI restrictions of nonconvex matrix inequalities, as our method does. We will discuss some of the differences in §6.4.

2. Model and Assumptions

2.1 Plant

The linear time-invariant plant has m inputs (actuators) and p outputs (sensors), and is given by its transfer function $P(s) \in \mathbf{C}^{p\times m}$, or more specifically by its frequency response $P(i\omega)$, for $\omega \in \mathbf{R}_+$. We do not assume that P is rational; it can, for example, include transport delay. We assume that the entries of the plant input $u(t) \in \mathbf{R}^m$ are measured in appropriate units (or scaled), so their sizes are (roughly) the same order. We make the same assumption about the plant output $y(t) \in \mathbf{R}^p$. These assumptions justify the use of the (unweighted) ℓ_2-norm to measure the actuator effort and deviation of the plant outputs from the reference signal, and more generally, it justifies the use of (unweighted) matrix norms to measure closed-loop gains.

We make several assumptions about the plant. We assume that $p \leq m$, i.e., there are at least as many actuators as plant outputs, and that $P(0)$ is full rank. This makes it possible to achieve perfect reference tracking at DC ($s = 0$). We will also assume that the plant is stable and strictly proper, i.e., $P(s) \to 0$ as $s \to \infty$. Most of our assumptions can be relaxed or extended to more general settings, but our goal is to keep the ideas simple for now. We will discuss various ways these assumptions can be relaxed in §7.

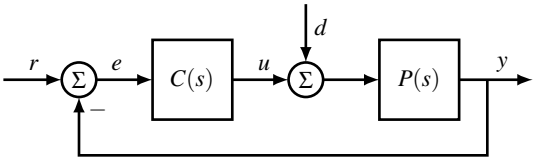

Figure 1. Classical feedback interconnection.

2.2 PID controller

The controller is a proportional-integral-derivative (PID) controller, given by

$$C(s) = K_{\mathrm{P}} + \frac{1}{s}K_{\mathrm{I}} + \frac{s}{1+\tau s}K_{\mathrm{D}},$$

where $K_{\mathrm{P}}, K_{\mathrm{I}}, K_{\mathrm{D}} \in \mathbf{R}^{m\times p}$ are the *proportional gain matrix*, *integral gain matrix*, and *derivative gain matrix*, respectively. The $3mp$ entries in these matrices are the design parameters we are to choose. The constant $\tau > 0$ is the derivative action time constant, and is assumed to be fixed and responsibly chosen, for example, a modest fraction of the desired closed-loop response time.

The plant and controller are connected in the classical loop shown in figure 1, described by the equations

$$e = r - y, \quad u = Ce, \quad y = P(u+d),$$

where r is the reference input, e is the error, and d is an input-referred plant disturbance. The signals u and y are the plant input and output, respectively.

2.3 Closed-loop transfer functions

We will be interested in several closed-loop transfer functions that we describe here.

Sensitivity. The transfer function from reference input r to error e is the *sensitivity* $S = (I+PC)^{-1}$. The size of S gives a measure of the tracking error; for low frequencies S should be small. When $S(0) = 0$ (a constraint we will impose), we have perfect static tracking. The maximum size of S, which occurs near the crossover frequency, is closely related to closed-loop damping and system stability.

Q-parameter. The transfer function from r to u is denoted Q, defined as $Q = C(I+PC)^{-1}$. Its size is a measure of the actuator effort. (We use Q to match the notation often used in Youla's parametrization of closed-loop transfer functions; see [Boyd and Barratt, 1991].)

Complementary sensitivity function. The *complementary sensitivity function* is $T = PC(I+PC)^{-1}$. It is the closed-loop transfer function from r to y. It is near the identity for low frequencies, and will be small for high frequencies; its maximum size is also related to closed-loop damping.

These three closed-loop transfer functions are sufficient to guarantee a sensible controller design (since P is assumed stable). They are related in various ways; for example, we have

$$S+T=I, \quad T=PQ, \quad Q=CS.$$

Other closed-loop transfer functions. Several other closed-loop transfer functions can also be considered. For example, the closed-loop transfer function from the plant disturbance d to the tracking error e is $R=-(I+PC)^{-1}P$. It will be clear how our approach extends to other transfer functions.

2.4 Notation

For a complex matrix $Z \in \mathbf{C}^{p\times q}$, Z^* is its (Hermitian) conjugate transpose, $\|Z\|$ denotes the spectral norm, i.e., the maximum singular value. For full rank Z, we let $\sigma_{\min}(Z)$ denote its minimum singular value. A square matrix is Hermitian if $Z=Z^*$. Between Hermitian matrices the symbol $\succeq 0$ is used to denote matrix inequality, so $Z \succeq 0$ means that Z is Hermitian and positive semidefinite. We use the notation $Z^{-*}=(Z^*)^{-1}$.

For a $p\times q$ transfer function H, $\|H\|_\infty$ is its $\mathbf{H}_\infty$-norm, $\|H\|_\infty = \sup_{\Re s\geq 0}\|H(s)\|$, which can be expressed as

$$\|H\|_\infty = \sup_{\omega\geq 0}\|H(i\omega)\|$$

when H is stable (i.e., $H \in \mathbf{H}_\infty^{p\times q}$).

3. Design Problem

3.1 Objective and constraints

Sensitivity and complementary sensitivity peaking. We require $\|S\|_\infty \leq S_{\max}$, where $S_{\max} > 1$. Reasonable values of $S_{\max}$ are in the range 1.1 to 1.6; lower values give a more damped closed-loop system. This constraint ensures closed-loop stability. We also require $\|T\|_\infty \leq T_{\max}$, with $T_{\max} > 1$. Reasonable values of $T_{\max}$ are similar to those for $S_{\max}$.

Static and low frequency sensitivity. Assuming that $P(0)K_{\mathrm{I}}$ is nonsingular, we have $S(0)=0$, which means that we have zero error for constant reference signals. The next term in the expansion of the sensitivity near $s=0$ is $S(s)\approx s(P(0)K_{\mathrm{I}})^{-1}$ for $|s|$ small. Our objective will be to attain the best possible low-frequency sensitivity,

which means we will minimize $\|(P(0)K_{\mathrm{I}})^{-1}\|$. This objective indirectly imposes the condition that we achieve perfect tracking since when it is finite we have $P(0)K_{\mathrm{I}}$ nonsingular.

Actuator authority limit. We require that $\|Q\|_\infty \leq Q_{\max}$. This sets a maximum value for the size of the closed-loop actuator signal in response to the reference signal. Since the plant is stable, this constraint ensures that the closed-loop system is stable; see [Boyd and Barratt, 1991].

We can determine a reasonable value for $Q_{\max}$ as follows. Any controller that has a finite objective value (i.e., achieves perfect reference tracking at $s = 0$) satisfies $T(0) = I = P(0)Q(0)$; this has the simple interpretation that at $s = 0$, the controller inverts the plant. It follows that $\|Q(0)\| \geq 1/\sigma_{\min}(P(0))$, from which we conclude $\|Q\|_\infty \geq 1/\sigma_{\min}(P(0))$. Thus we must have $Q_{\max} \geq 1/\sigma_{\min}(P(0))$. The righthand side can be interpreted as the minimum actuator effort (measured by $\|Q\|_\infty$) required to achieve static tracking. A reasonable value for $Q_{\max}$ is therefore a modest multiple of $1/\sigma_{\min}(P(0))$, say, three to ten.

Design problem. Putting it all together we obtain the problem

$$\begin{array}{ll} \text{minimize} & \|(P(0)K_{\mathrm{I}})^{-1}\| \\ \text{subject to} & \|S\|_\infty \leq S_{\max}, \\ & \|T\|_\infty \leq T_{\max}, \\ & \|Q\|_\infty \leq Q_{\max}. \end{array} \tag{1}$$

The variables to be chosen are the coefficient matrices $K_{\mathrm{P}}, K_{\mathrm{I}}, K_{\mathrm{D}}$; the problem data are the plant transfer function P, the controller derivative time constant τ, and the design parameters $S_{\max}$, $T_{\max}$, and $Q_{\max}$. This problem is not convex. Note that the objective contains the implied constraint that $P(0)K_{\mathrm{I}}$ is invertible, which implies that perfect static tracking is achieved for constant reference inputs.

3.2 Sampling semi-infinite constraints

The constraints on the closed-loop transfer functions can be expressed as, for example,

$$\|S(i\omega)\| \leq S_{\max}, \quad \forall \omega \geq 0.$$

This is a so-called semi-infinite constraint, since it consists of an infinite number of constraints, one for each $\omega \geq 0$. Semi-infinite constraints such as these (with one parameter, ω) are readily handled by choosing a reasonable finite (but large) set of frequency samples $0 < \omega_1 < \cdots < \omega_K$, and replacing the semi-infinite constraints with the finite set of constraints at each of the given frequencies. For example, we replace the constraint $\|S\|_\infty \leq S_{\max}$ with $\|S(i\omega_k)\| \leq S_{\max}$, $k = 1, \ldots, N$. Our optimization method has a computational complexity that grows linearly with N, so we can choose a large enough value of N (say, several hundred or more) that this sampling has no practical effect. By 'reasonable' we mean that the frequency sampling

is fine enough to catch any rapid changes in the closed-loop transfer function with frequency, and also cover an appropriate range; in particular, we assume that at ω_1 and ω_N the transfer functions are near their asymptotic values,

$$S(0) = 0, \quad T(0) = I, \quad Q(0) = K_{\mathrm{I}}(P(0)K_{\mathrm{I}})^{-1},$$

and

$$S(\infty) = I, \quad T(\infty) = 0, \quad Q(\infty) = K_{\mathrm{P}} + (1/\tau)K_{\mathrm{D}},$$

respectively.

We will use subscripts to denote a transfer function evaluated at the frequency $s = i\omega_k$. For example, $P_k = P(i\omega_k)$, which is a (given) complex matrix. Note that the quantity $C_k = C(i\omega_k)$ is a complex matrix, and is an affine function of the design variables $K_{\mathrm{P}}, K_{\mathrm{P}}, K_{\mathrm{D}}$.

The sampled problem is then

$$\begin{array}{ll} \text{minimize} & \|(P(0)K_{\mathrm{I}})^{-1}\| \\ \text{subject to} & \|S_k\| \le S_{\max}, \\ & \|T_k\| \le T_{\max}, \\ & \|Q_k\| \le Q_{\max}, \\ & \quad k = 1, \ldots, N. \end{array} \tag{2}$$

This problem has $3N$ constraints, each of which has the form of a matrix norm inequality. The arguments of the matrix norm inequalities, however, are complex functions of the design variables $K_{\mathrm{P}}, K_{\mathrm{I}}, K_{\mathrm{D}}$, given by the various formulas for the closed-loop transfer functions.

4. Quadratic Matrix Inequality Form

In this section we show how the (frequency sampled) design problem (2) can be cast in a simple form in which every constraint has the same *quadratic matrix inequality* (QMI) form

$$Z^*Z \succeq Y^*Y, \tag{3}$$

where both Z and Y are affine functions of the variables.

We start with the objective. First we note that we can just as well maximize $\sigma_{\min}(P(0)K_{\mathrm{I}}) = 1/\|(P(0)K_{\mathrm{I}})^{-1}\|$. We introduce a new scalar variable t, which we maximize subject to $\sigma_{\min}(P(0)K_{\mathrm{I}}) \ge t$. (This is the standard epigraph transformation; see [Boyd and Vandenberghe, 2004, §4.2.4].) We then observe that

$$\sigma_{\min}(P(0)K_{\mathrm{I}}) \ge t \Leftrightarrow (P(0)K_{\mathrm{I}})^*(P(0)K_{\mathrm{I}}) \succeq t^2 I,$$

which has the form (3) with $Z = P(0)K_{\mathrm{I}}$ and $Y = tI$, both of which are affine functions of the variables.

Now consider the sensitivity peaking limit $\|S_k\| \leq S_{\max}$. As above, we have

$$\begin{aligned} \|S_k\| \leq S_{\max} &\Leftrightarrow (I+P_kC_k)^{-*}(I+P_kC_k)^{-1} \preceq S_{\max}^2 I \\ &\Leftrightarrow (I+P_kC_k)^{*}(I+P_kC_k) \succeq (1/S_{\max}^2) I, \end{aligned}$$

where in the second line we have multiplied the left and right sides by $(I+P_kC_k)^*$ and by $(I+P_kC_k)$, respectively. This has the QMI form (3) with $Z = I+P_kC_k$ and $Y = (1/S_{\max})I$.

For the complementary sensitivity constraint $\|T_k\| \leq T_{\max}$, we have

$$\begin{aligned} \|T_k\| \leq T_{\max} &\Leftrightarrow (I+P_k^*C_k^*)^{-1}C_k^*P_k^*P_kC(I+P_kC_k)^{-1} \preceq T_{\max}^2 I \\ &\Leftrightarrow (I+P_kC_k)^{*}(I+P_kC_k) \succeq (1/T_{\max}^2)(P_kC_k)^*(P_kC_k). \end{aligned}$$

This has the QMI form (3) with $Z = I+P_kC_k$ and $Y = (1/T_{\max})P_kC_k$.

In a similar way we have

$$\|Q_k\| \leq Q_{\max} \Leftrightarrow (I+P_kC_k)^{*}(I+P_kC_k) \succeq (1/Q_{\max}^2)C_k^*C_k,$$

which has QMI form with $Z = I+P_kC_k$ and $Y = (1/Q_{\max})C_k$.

We arrive at a problem with the form

$$\begin{array}{ll} \text{maximize} & t \\ \text{subject to} & Z_k^*Z_k \succeq Y_k^*Y_k, \quad k = 1,\ldots,M, \end{array} \tag{4}$$

with variables $t, K_{\mathrm{P}}, K_{\mathrm{I}}, K_{\mathrm{D}}$. (Here we have $M = 3N+1$.) This is the PID controller design problem in QMI form.

5. Linear Matrix Inequality Restriction

We first show how to form a (convex) *linear matrix inequality* (LMI) restriction for the QMI $Z^*Z \succeq Y^*Y$. (See [Boyd, El Ghaoui, et al., 1994] for background on matrix inequalities.) The QMI is already convex in Y, so we can focus on Z. We start with the simple matrix inequality

$$0 \preceq (Z-\tilde{Z})^*(Z-\tilde{Z}) = Z^*Z - Z^*\tilde{Z} - \tilde{Z}^*Z + \tilde{Z}^*\tilde{Z},$$

valid for any matrices Z and $\tilde{Z}$. Re-arranging we get

$$Z^*Z \succeq Z^*\tilde{Z} + \tilde{Z}^*Z - \tilde{Z}^*\tilde{Z}.$$

The lefthand side is a quadratic function of Z; the righthand side is an affine function of Z. It follows that the matrix inequality

$$Z^*\tilde{Z} + \tilde{Z}^*Z - \tilde{Z}^*\tilde{Z} \succeq Y^*Y,$$

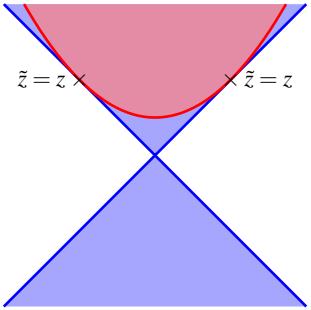

Figure 2. LMI restriction (shown in red) of QMI (shown in blue) for simple case when Z and Y are scalar and real.

which is convex in (Z,Y), implies $Z^*Z \succeq Y^*Y$; that is, it is a convex restriction of the QMI. We can write this convex quadratic matrix inequality as an LMI

$$\left[\begin{array}{cc} Z^*\tilde{Z}+\tilde{Z}^*Z-\tilde{Z}^*\tilde{Z} & Y^* \\ Y & I \end{array}\right] \succeq 0. \tag{5}$$

(Here Z and Y are the variables; $\tilde{Z}$ is an arbitrary matrix.) For any matrix $\tilde{Z}$, the LMI (5) implies the QMI $Z^*Z \succeq Y^*Y$. We call it the *LMI restriction* of the QMI, obtained at the point $\tilde{Z}$.

The LMI restriction of the QMI is illustrated in figure 2, for the simple case of real scalar Z and Y. In this case the QMI is $z^2 \geq y^2$, which gives the two lightly shaded cones in the figure, with blue boundary. The LMI restriction at $\tilde{z}$ is given by $2\tilde{z}z - \tilde{z}^2 \geq y^2$, shown as the shaded region bounded by the parabola, with red boundary. The two boundaries touch at the point where $z = \tilde{z}$.

Now consider the QMI form PID controller design problem (4). Given any matrices $\tilde{Z}_1, \ldots, \tilde{Z}_M$, we can form the *LMI restricted problem*

$$\begin{array}{ll} \text{maximize} & t \\ \text{subject to} & \left[\begin{array}{cc} Z_k^*\tilde{Z}_k+\tilde{Z}_k^*Z_k-\tilde{Z}_k^*\tilde{Z}_k & Y_k^* \\ Y_k & I \end{array}\right] \succeq 0, \quad k=1,\ldots,M. \end{array} \tag{6}$$

This problem has linear objective and LMI constraints, and so is a semidefinite program (SDP) [Vandenberghe and Boyd, 1996]. It is readily solved (globally). Note that any solution of the LMI restriction is feasible for the QMI problem. The LMI restriction, however, need not be feasible; this depends on the choice of $\tilde{Z}_k$.

Simplifications. In the inequalities associated with S_k, the matrix Y_k does not depend on the controller parameters (i.e., is constant). In this case we can work directly with the smaller (equivalent) LMIs

$$Z_k^*\tilde{Z}_k + \tilde{Z}_k^*Z_k - \tilde{Z}_k^*\tilde{Z}_k \succeq Y_k^*Y_k,$$

which can give a modest computational advantage.

Another simplification is to maximize the variable $w = t^2$; in this case the single inequality associated with the objective can be handled as the smaller LMI

$$Z_1^* \tilde{Z}_1 + \tilde{Z}_1^* Z_1 - \tilde{Z}_1^* \tilde{Z}_1 \succeq wI,$$

where $Z_1 = P(0)K_I$. This simplification leads to slightly faster convergence, since t is only an upper bound on $\|(P(0)K_I)^{-1}\|$, whereas $w^{-1/2}$ is actually equal to the objective, after optimization.

6. The Method

6.1 Controller initialization

We first initialize the controller with

$$K_P = 0, \quad K_I = \varepsilon P(0)^\dagger, \quad K_D = 0,$$

where ε is small and positive, and $P(0)^\dagger = P(0)^T(P(0)P(0)^T)^{-1}$ is the pseudo-inverse of the DC gain. For small enough ε, this controller is feasible. Indeed, as $\varepsilon \to 0$, we have

$$S(s) \to \frac{s}{\varepsilon + s} I, \quad T(s) \to \frac{\varepsilon}{\varepsilon + s} I, \quad Q(s) \to \frac{\varepsilon}{\varepsilon + s} P(0)^\dagger,$$

from which it follows that the constraints $\|S\|_\infty \leq S_{\max}$, $\|T\|_\infty \leq T_{\max}$, and $\|Q\|_\infty \leq Q_{\max}$ are feasible for small enough ε (assuming $S_{\max} > 1$, $T_{\max} > 1$, and $Q_{\max} > 1/\sigma_{\min}(P(0))$). Note that $\|Q\|_\infty$ finite implies closed-loop stability of this initial controller.

6.2 Iteration

We then repeat the following steps. We form the LMI restriction (6), using $\tilde{Z}_k = Z_k^{\text{curr}}$, where Z_k^{curr} is the current value of Z_k. This choice guarantees the LMI restriction is feasible. We solve this SDP to get the updated values of the design variables. These are feasible, since they were constrained by the restrictions, and the objective t (which is an upper bound on the original objective $\|(P(0)K_I)^{-1}\|$) cannot decrease.

6.3 Convergence

The iterates are all feasible (and we have closed-loop stability since $\|Q\|_\infty$ is finite), and the objective is nonincreasing. Since it is nonnegative, the objective converges. We can stop when not much progress is being made, which is typically after ten or fewer iterations. At convergence, the optimal value of t, which is in general an *upper bound* on the original objective $\|(P(0)K_I)^{-1}\|$, is actually *equal* to this value. (When we optimize with the variable w, it is directly equal to the objective.)

6.4 Connections, interpretations, and prior work

Our method is similar to, but not the same as, the convex-concave method for the SISO case described in [Hast et al., 2013]. In that paper we linearize a scalar inequality of the form $|Z| \geq |Y|$, which is not the same as linearizing (as we do here) the quadratic inequality $|Z|^2 \geq |Y|^2$. The idea of linearizing concave terms in an otherwise convex optimization problem, which gives a convex restriction, is an old one that has been (re-)invented many times, for many applications; see, e.g., the references in [Lipp and Boyd, 2014] or [Yuille and Rangarajan, 2003]. The idea of linearizing a matrix inequality is given in [Lipp and Boyd, 2014].

The convex-concave procedure is in turn a special case of a very general method for finding a local minimum of a nonconvex optimization problem. In each iteration we replace the objective function and each constraint function by a convex majorization that is tight at the given point, and solve the resulting convex problem. This idea traces back at least to 1970 [Ortega and Rheinboldt, 1970], and has been widely used since then; see [Lipp and Boyd, 2014].

There is also a very close connection of our method to BMIs and methods for them, such as alternating minimization over the two groups of variables. The connection is easiest to see and state when Z and Y have the same dimensions (in general the inequality $Z^*Z \succeq Y^*Y$ implies they have the same number of columns). Define

$$U = (1/2)(Z+Y), \qquad V = (1/2)(Z-Y),$$

so $Z = U+V$ and $Y = U-V$. Then we have

$$\begin{array}{lll} Z^*Z \succeq Y^*Y & \Leftrightarrow & (U+V)^*(U+V) \succeq (U-V)^*(U-V) \\ & \Leftrightarrow & U^*U+V^*U+U^*V+V^*V \succeq U^*U-V^*U-U^*V+V^*V \\ & \Leftrightarrow & V^*U+U^*V \succeq 0, \end{array}$$

which we recognize as a BMI in U and V. Thus our quadratic matrix inequality can be expressed as an equivalent BMI. MIMO PID design via BMIs is discussed in [Bianchi et al., 2008].

The closest prior work is [Saeki, Ogawa, et al., 2010], which contains many of the ideas we use in the present paper. The authors develop quadratic matrix inequalities similar to the ones we use here, and derive a method for MIMO PID design that uses LMI restrictions, as we do. While the two methods are clearly closely related, we are unable to derive our exact algorithm from theirs. We can identify several differences in the approach. First, we consider separate closed-loop transfer functions (e.g., S, T, Q), where they lump them together into one block closed-loop transfer function. One advantage of considering these closed-loop transfer functions separately is that we can give simple and universal choices for the upper bound (such as 1.4, for example, for S). Second, we consider stable plants, which allows us to give a simple low gain PID initialization. Finally, we consider a generic quadratic matrix inequality, and develop a simple universal LMI restriction.

7. Extensions and Variations

In this section we list various extensions and variations on the MIMO PID controller design problem, starting with simple ones and moving to more complex ones. While the basic iteration will work in all of these variations, the design must start from a feasible initial controller, which may be a challenge to find, depending on the variation. (We make more specific comments about this below.)

Exchanging objectives and constraints. As always, we can exchange constraints and objective; for example, we could impose a constraint on $\|(P(0)K_{\mathrm{I}})^{-1}\|$, and instead minimize another objective, such as $\|Q\|_\infty$. In this example, we would be minimizing actuator effort for a given fixed limit on the low-frequency sensitivity.

Frequency-dependent bounds. The bounds $S_{\max}, T_{\max}, Q_{\max}$ could be functions of frequency. This could be used to shape the various closed-loop transfer functions in more sophisticated ways than described here.

Other closed-loop transfer functions. The same approach works for other closed-loop transfer functions. For example, consider $R = -(I+PC)^{-1}P$, which is the closed-loop transfer function from disturbance to error. A limit on R, say, $\|R\|_\infty \leq R_{\max}$, can be expressed as a QMI as follows:

$$\|R_k\| \leq R_{\max} \Leftrightarrow (I+P_kC_k)(I+P_kC_k)^* \succeq (1/R_{\max}^2)P_kP_k^*,$$

which has QMI form with $Z = (I+P_kC_k)^*$ and $Y = (1/R_{\max})P_k^*$. (Note the Hermitian conjugates in this case.)

Low frequency disturbance optimization. We can optimize low-frequency values of R instead of S. At low frequencies we have

$$R(s) = -S(s)P(s) \approx -s(P(0)K_{\mathrm{I}})^{-1}P(0),$$

and we arrive at a very similar problem, which is also easily expressed in our QMI form.

High frequency roll-off. Our method relies only on the fact that $C(s)$ is a linear function of the design variables $K_{\mathrm{P}}, K_{\mathrm{I}}, K_{\mathrm{D}}$. This allows us to use many other variations on the PID controller. As an example of simple variation that is very useful in practice, we can use the controller

$$C(s) = \left(\frac{1}{1+s\tau+(s\tau)^2/2}\right)\left(K_{\mathrm{P}} + \frac{1}{s}K_{\mathrm{I}} + sK_{\mathrm{D}}\right),$$

where $\tau > 0$ is a (fixed) time constant. Here we have an ideal PID controller, with a second-order high frequency roll-off.

Unstable plants. The method can be extended to handle unstable plants, but in this case the initial controller must be stabilizing and also satisfy the constraints; see, e.g., [Hast et al., 2013] for an example. (To satisfy the constraints, they can initially be relaxed.) When initialized this way, all iterates (and the final controller design) will be stabilizing.

More outputs than actuators. We can handle the case when $p > m$ (more outputs than actuators), so perfect static tracking cannot be achieved. In this case we cannot have zero sensitivity at $s = 0$, but we can optimize over the value of $S(0)$, for example, minimize or limit its norm.

Convex constraints on controller parameters. Any convex constraints on the controller parameters can be imposed. For example, we can limit the values of any of the coefficients. A very interesting option here is to limit the sparsity pattern of C by requiring some entries to be zero. This gives structured MIMO PID controller design [Saeki, 2006].

The simple initialization method described in §6.1 will generally not work when the controller parameters are constrained, for example, when a specific sparsity pattern is imposed.

Convex cost terms. We can add any convex function of the controller parameters to the objective. For example we can add regularization to the objective, i.e., a function that encourages the controller parameters to be small. The classical example is the sum of squares term

$$\lambda \sum_{ij} \left((K_{\mathrm{P}})_{ij}^2 + (K_{\mathrm{I}})_{ij}^2 + (K_{\mathrm{D}})_{ij}^2 \right),$$

where $\lambda > 0$ is a parameter used to trade off low-frequency rejection and the size of the controller parameters (measured by the sum of squares).

A very interesting regularization is one that encourages sparsity in the controller parameters, such as

$$\lambda \sum_{ij} \max\{(K_{\mathrm{P}})_{ij}, (K_{\mathrm{I}})_{ij}, (K_{\mathrm{D}})_{ij}\}.$$

This regularization will encourage sparsity in $C(s)$; for similar work, see, e.g., [F. Lin et al., 2012] (for sparse controller design) and [Zou and Hastie, 2005] (for sparsity of blocks of regressors in statistics).

Closed-loop convex constraints. We can also add any contraint or objective term that is convex in the closed-loop transfer functions; see the book [Boyd and Barratt, 1991]. For example, we could include time-domain constraints such as a maximum step response settling time. The very same method (with some added terms to handle the added constraints) will work.

Robustness to plant variations. We can wrap robustness to plant variations into the method. A particularly simple (but very effective) method that gives robustness is to require that the constraints hold not just for one plant, but for several or many plausible values of the plant transfer function. This leads to a bigger problem to solve, but the same method works.

More general controllers. Finally, it should be clear that the method works for any linearly parametrized controller, and not just the simple PID structure that we have focussed on here. For more general structures the design initialization can become a challenge, however.

8. Examples

In this section we describe numerical results for a classic MIMO plant, the 2-input 2-output Wood-Berry binary distillation column described in [Wood and Berry, 1973]. The computations were carried out using CVX [Research, 2012; Grant and Boyd, 2008].

The plant transfer function is

$$P(s) = \begin{bmatrix} \dfrac{12.8e^{-s}}{16.7s+1} & \dfrac{-18.9e^{-3s}}{21.0s+1} \\ \dfrac{6.6e^{-7s}}{10.9s+1} & \dfrac{-19.4e^{-3s}}{14.2+1} \end{bmatrix}.$$

Each entry is a first order system with a time delay. The dynamics are quite coupled, so finding a good MIMO PID controller is not simple. Several design methods and actual designs for this plant have been proposed in the literature, including [Dong and Brosilow, 1997; Wang et al., 1997; Tan et al., 2002]. Our method produced quite similar results, with the same or better metrics judged by our objectives (naturally).

We used design parameters

$$S_{\max} = 1.4, \quad T_{\max} = 1.4, \quad Q_{\max} = 3/\sigma_{\min}(P(0)) = 0.738.$$

The derivative action time constant is chosen to be $\tau = 0.3$. The semi-infinite constraints are sampled using $N = 300$ logarithmically spaced frequency samples in the interval $\left[10^{-3}, 10^{3}\right]$. The initial design uses the method described in §6.1 with $\varepsilon = 0.01$.

The algorithm converges in 7 iterations (which takes a few minutes to run in our simple implementation) to the values

$$K_{\mathrm{P}} = \begin{bmatrix} 0.1750 & -0.0470 \\ -0.0751 & -0.0709 \end{bmatrix}, \quad K_{\mathrm{I}} = \begin{bmatrix} 0.0913 & -0.0345 \\ 0.0402 & -0.0328 \end{bmatrix},$$
$$K_{\mathrm{D}} = \begin{bmatrix} 0.1601 & -0.0051 \\ 0.0201 & -0.1768 \end{bmatrix},$$

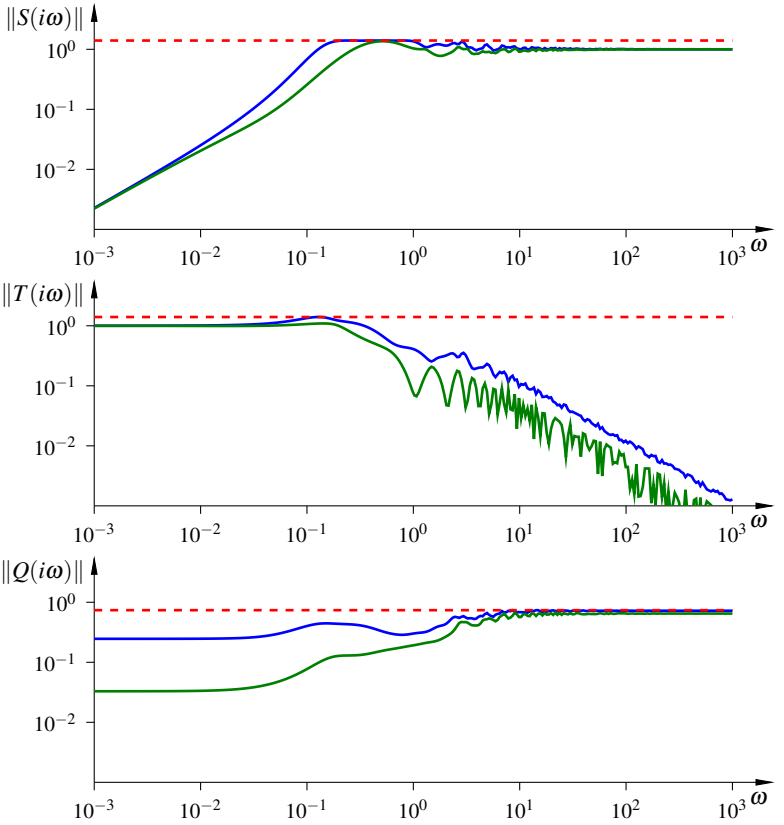

Figure 3. Closed-loop transfer function singular values versus frequency, with constraints shown in red.

which achieves objective value $\|(P(0)K_{\mathrm{I}})^{-1}\| = 2.25$. The resulting closed-loop transfer function singular values are plotted versus frequency in figure 3, along with the imposed limits.

To demonstrate one simple extension, we also carry out MIMO PID design with the additional constraint that the controller is diagonal, i.e., consists of two SISO PID loops. We initialize the algorithm with low gain PI control from y_1 to u_1 and from y_2 to u_2, using the (diagonal) controller

$$K_{\mathrm{P}} = 10^{-3}\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}, \quad K_{\mathrm{I}} = 10^{-3}\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}, \quad K_{\mathrm{D}} = 0.$$

(The minus sign in the 2,2 entry is due to the negative 2,2 value of the 2,2 entry of $P(0)$.) The algorithm converges in 8 iterations (taking a few minutes in our simple

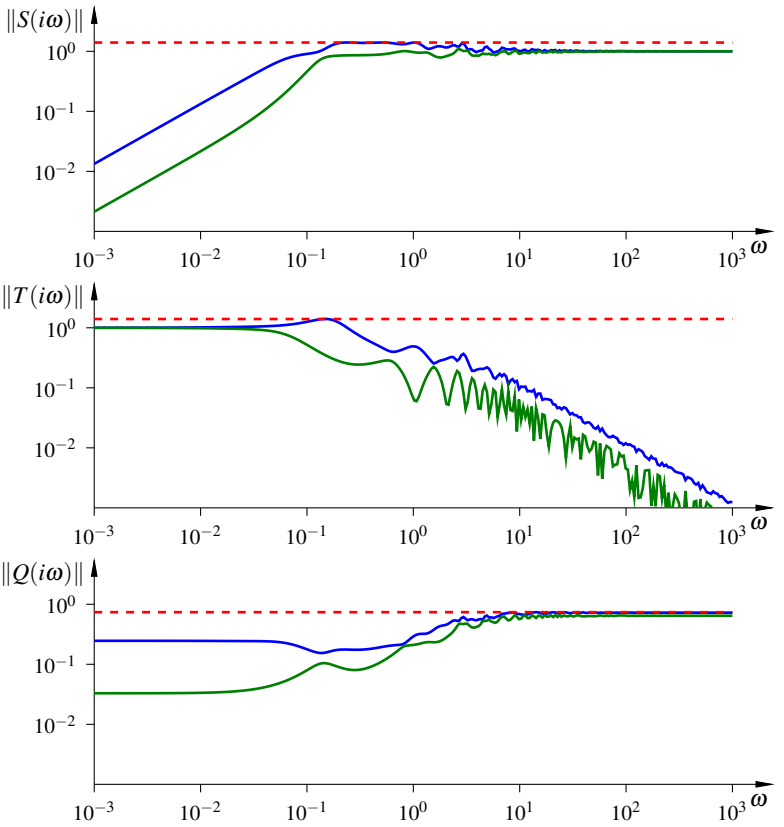

Figure 4. Closed-loop transfer function singular values versus frequency, with constraints shown in red, for diagonal PID design.

implementation) to the controller

$$K_{\mathrm{P}} = \begin{bmatrix} 0.1535 & 0 \\ 0 & -0.0692 \end{bmatrix}, \quad K_{\mathrm{I}} = \begin{bmatrix} 0.0210 & 0 \\ 0 & -0.0136 \end{bmatrix},$$

$$K_{\mathrm{D}} = \begin{bmatrix} 0.1714 & 0 \\ 0 & -0.1725 \end{bmatrix},$$

which achieves objective value $\|(P(0)K_{\mathrm{I}})^{-1}\| = 13.36$, considerably worse than the objective value obtained with a general MIMO PID controller. The resulting closed-loop transfer function singular values are plotted in figure 4. We can see that low frequency sensitivity is considerably worse than that achieved by the full MIMO controller, for example by noting the value of $\|S(\omega)\|_2$ for $\omega = 10^{-2}$.

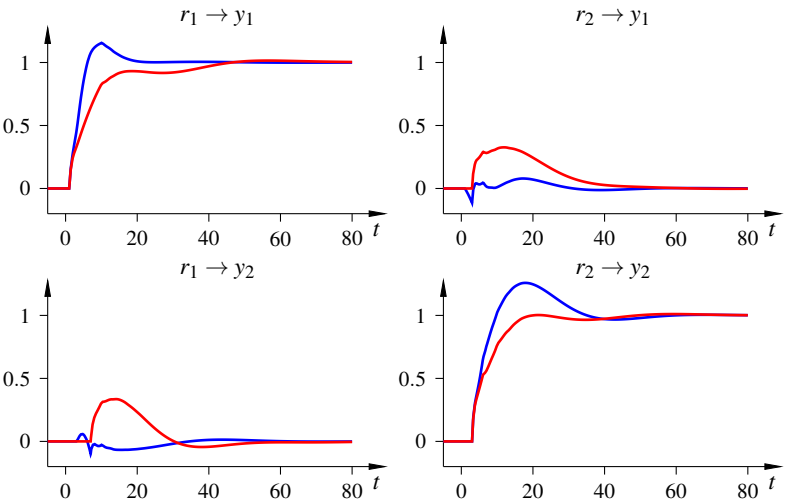

Figure 5. Closed-loop step response from r to y for the MIMO PID controller (blue) and the diagonal PID controller (red).

The step responses of T, the transfer function from r to y, are plotted in figure 5, for both the full MIMO PID controller and the diagonal PID controller. Here too we can observe the worse low frequency rejection for the diagonal PID design, for example in the larger off-diagonal entries of the step response. The step responses of Q, the transfer function from r to u, are plotted in figure 6, for both the full MIMO PID controller and the diagonal PID controller.

9. Conclusions

In this paper we have described a simple method for effectively designing MIMO PID controllers for stable plants given by transfer function (at an appropriate set of frequencies). The method relies on solving a short sequence of SDPs (typically 10 or fewer), and although it cannot guarantee finding the globally optimal design, it appears to find very good designs in practical problems. The method is related to several other methods for MIMO PID design, and relies on ideas that have been used in several other contexts in optimization, such as the convex-concave procedure, and iterative convex restriction.

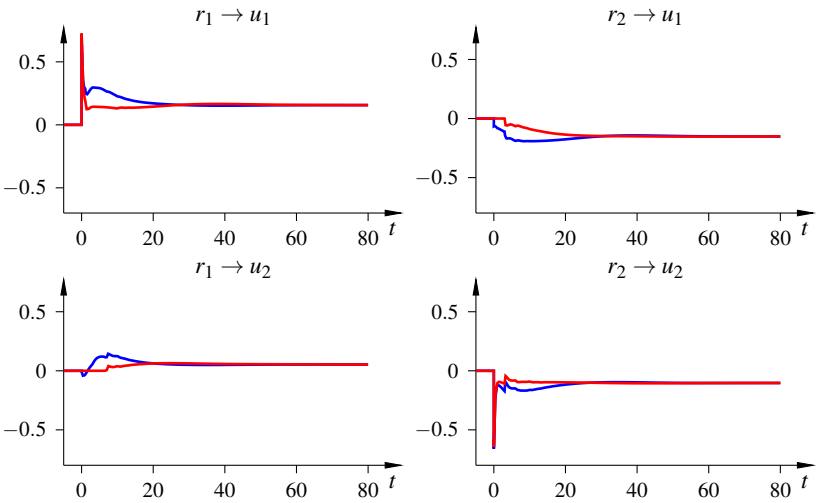

Figure 6. Closed-loop step response from r to u for the MIMO PID controller (blue) and the diagonal PID controller (red).

References

Åström, K. J., H. Panagopoulos, and T. Hägglund (1998). "Design of PI controllers based on non-convex optimization". *Automatica* **34**:5, pp. 585–601.

Åström, K. J. and T. Hägglund (2006). *Advanced PID Control*. Instrumentation, Systems, and Automation Society.

Bianchi, F. D., R. D. Mantz, and C. F. Christiansen (2008). "Multivariable PID control with set-point weighting via BMI optimisation". *Automatica* **44**:2, pp. 472–478.

Boyd, S. and C. Barratt (1991). *Linear Controller Design — Limits of Performance*. Prentice-Hall.

Boyd, S., L. El Ghaoui, E. Feron, and V. Balakrishnan (1994). *Linear Matrix Inequalities in System and Control Theory*. Vol. 15. Studies in Applied Mathematics. SIAM.

Boyd, S. and L. Vandenberghe (2004). *Convex Optimization*. Cambridge University Press.

Dong, J. and C. Brosilow (1997). "Design of robust multivariable PID controllers via IMC". In: *Proceedings of the American Control Conference*. Vol. 5, pp. 3380–3384.

Garpinger, O. and T. Hägglund (2008). "A software tool for robust PID design". In: *Proceedings 17th IFAC World Congress*.

Garpinger, O., T. Hägglund, and K. J. Åström (2012). "Criteria and trade-offs in PID design". In: *IFAC Conference on Advances in PID Control*. Brescia, Italy.

Grant, M. and S. Boyd (2008). "Graph implementations for nonsmooth convex programs". In: Blondel, V. et al. (Eds.). *Recent Advances in Learning and Control*. Lecture Notes in Control and Information Sciences. Springer-Verlag Limited, pp. 95–110.

Hast, M., K. J. Åström, B. Bernhardsson, and S. Boyd (2013). "PID design by convex-concave optimization". In: *Proceedings European Control Conference*, pp. 4460–4465.

Lin, C., Q. G. Wang, and T. H. Lee (2004). "An improvement on multivariable PID controller design via iterative LMI approach". *Automatica* **40**:3, pp. 519–525.

Lin, F., M. Fardad, and M. Jovanovic (2012). "Sparse feedback synthesis via the alternating direction method of multipliers". In: *Proceedings of the American Control Conference*, pp. 4765–4770.

Lipp, T. and S. Boyd (2014). *Extensions and variations on the convex-concave procedure*. To be published.

Luyben, W. L. (1986). "Simple method for tuning SISO controllers in multivariable systems". *Industrial & Engineering Chemistry Process Design and Development* **25**:3, pp. 654–660.

Ortega, R. and W. Rheinboldt (1970). *Iterative Solutions of Nonlinear Equations in Several Variables*. Academic Press.

Panagopoulos, H., K. J. Åström, and T. Hägglund (2002). "Design of PID controllers based on constrained optimisation". In: *IEE Proceedings on Control Theory and Applications*. Vol. 149. 1, pp. 32–40.

Penttinen, J. and H. N. Koivo (1980). "Multivariable tuning regulators for unknown systems". *Automatica* **16**:4, pp. 393–398.

Research, C. (2012). *CVX: MATLAB software for disciplined convex programming, version 2.0*. `http://cvxr.com/cvx`.

Saeki, M. (2006). "Fixed structure PID controller design for standard H_∞ control problem". *Automatica* **42**:1, pp. 93–100.

Saeki, M., K. Kashiwagi, and N. Wada (2007). "Design of multivariable H_∞ PID controller using frequency response". In: *Proc. of the 2007 IEEE Int. Conf. on Control Applications*, pp. 1565–1570.

Saeki, M., M. Ogawa, and N. Wada (2010). "Low-order H_∞ controller design on the frequency domain by partial optimization". *International Journal of Robust and Nonlinear Control* **20**, pp. 323–333.

Tan, W., T. Chen, and H. Marquez (2002). "Robust controller design and PID tuning for multivariable processes". *Asian Journal of control* **4**:4, pp. 439–451.

Vandenberghe, L. and S. Boyd (1996). "Semidefinite programming". *SIAM Review* **38**:1, pp. 49–95.

Vilanova, R. and A. Visioli (2012). *PID Control in the Third Millennium: Lessons Learned and New Approaches*. Springer.

Wang, Q., B. Zou, T. Lee, and Q. Bi (1997). "Auto-tuning of multivariable PID controllers from decentralized relay feedback". *Automatica* **33**:3, pp. 319–330.

Wood, R. and M. Berry (1973). "Terminal composition control of a binary distillation column". *Chemical Engineering Science* **28**:9, pp. 1707–1717.

Yuille, A. and A. Rangarajan (2003). "The concave-convex procedure". *Neural Computation* **15**:4, pp. 915–936.

Zou, H. and T. Hastie (2005). "Regularization and variable selection via the elastic net". **67**:2, pp. 301–320.

Paper III

Design of Optimal Low-Order Feedforward Controllers

Martin Hast Tore Hägglund

Abstract

Design rules for optimal feedforward controllers with lead-lag structure in the presence of measurable disturbances are presented. The design rules are based on stable first-order models with time delays, FOTD, and are optimal in the sense of minimizing the integrated-squared error. The rules are derived for an open-loop setting, considering a step disturbance. This paper also discusses a general feedforward structure, which enables decoupling in the design of feedback and feedforward controllers, and justifies the open-loop setting.

1. Introduction

Feedforward is an efficient way to reduce control errors both for reference tracking and disturbance rejection, given that the disturbances acting on the system are measurable. This paper treats the subject of disturbance rejection. Due to model uncertainties, feedforward cannot eliminate the disturbance and it is therefore often used along with feedback control.

For the design of feedback controllers a large number of design methods exists. For design of PID-controllers there exists a large number of analytical methods for choosing the control parameters, see e.g., [Åström and Hägglund, 2004], [Skogestad, 2003] or [Ziegler and Nichols, 1942]. However, there seems to be a lack of simple methods for tuning feedforward controllers.

The design of low-order feedforward controllers has previously been addressed by e.g., [Isaksson et al., 2008] and [Guzmán and Hägglund, 2011]. [Isaksson et al., 2008] proposes an iterative design procedure, to minimize a system norm in the frequency domain, that takes the feedback controller into account. [Guzmán and Hägglund, 2011] provides simple tuning rules for feedforward controllers, taking the feedback controller into account, in order to reduce the integrated absolute error, IAE.

This paper presents an analytic solution to the problem of designing a feedforward lead-lag filter which minimizes the integrated square error when the system is subjected to a measurable step disturbance. The design rules are derived for FOTDs. The resulting feedforward controller is optimal in an open-loop setting. In general, feedforward controllers should be designed taking the feedback controller into account since they interact.

In [Brosilow and Joseph, 2002] a feedforward structure that separates the feedback and feedforward control design, was presented. This idea has been adopted in this paper and justifies that the designed controller, while optimal in the open-loop case, gives good performance when used in conjunction with feedback control. This structure makes use of the same process models that is used for the design of the feedforward controller. The structure have similarities with Internal Model Control, IMC, see [Garcia and Morari, 1982]. Robust feedforward design within the IMC framework has addressed by [Vilanova et al., 2009].

2. Feedforward Structure

This section describes different structures for feedforwarding from measurable disturbances. Firstly, the most common open-loop and closed structures are discussed. Secondly, a feedforward structure that separates the design of feedforward and feedback controllers, as presented in [Brosilow and Joseph, 2002] is discussed.

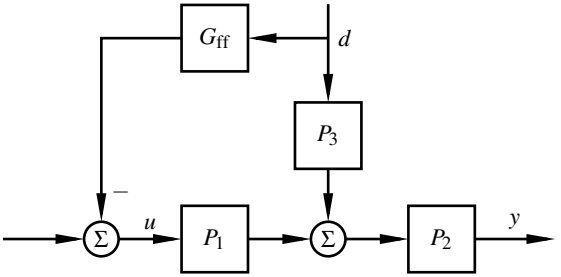

Figure 1. Open-loop structure.

2.1 Open-Loop Behavior

Consider the open-loop structure in Fig. 1 where d is the measurable disturbance, y is the system output and u is the system input. The transfer function from d to y is given by

$$G_{\text{o}}(s) = P_2(s)\big(P_3(s) - P_1(s)G_{\text{ff}}(s)\big). \tag{1}$$

In order to eliminate the effect of the disturbance d the feedforward controller should be chosen as $G_{ff}(s) = P_3(s)P_1^{-1}(s)$. This controller is not always possible or desirable to realize, as e.g., the order of $P_1(s)$ is greater than the order of $P_3(s)$, the time delay of $P_1(s)$ is greater than the time delay of $P_3(s)$ or if $P_1(s)$ has zeros in the right-half plane.

2.2 Feedforward - Feedback Interaction

Compensating for a measurable disturbance using only an open-loop feedforward structure is seldom desirable. Due to model errors and unmeasurable disturbances a feedback controller is needed. Connecting a feedback controller $C(s)$, see Fig. 2, renders the following transfer function from d to y,

$$G_{\text{cl}}(s) = \frac{P_2(s)\left(P_3(s) - P_1(s)G_{\text{ff}}(s)\right)}{1 + P_2(s)P_1(s)C(s)}. \tag{2}$$

When it is possible to realize perfect feedforward $G_{\text{ff}} = P_3(s)P_1^{-1}(s)$ no problems will arise since (2) will be zero. However, when the perfect feedforward is not realizable the closed-loop behavior will differ from the open-loop behavior given by (1). Ways of modifying the feedforward controller in order to get a satisfying system response from the closed-loop system has been presented in [Isaksson et al., 2008] and [Guzmán and Hägglund, 2011].

2.3 Non-Interacting Feedforward Structure.

In [Brosilow and Joseph, 2002] a feedforward structure, equivalent to the one in Fig. 3, was presented. Dropping the argument s, the transfer function from d to y is

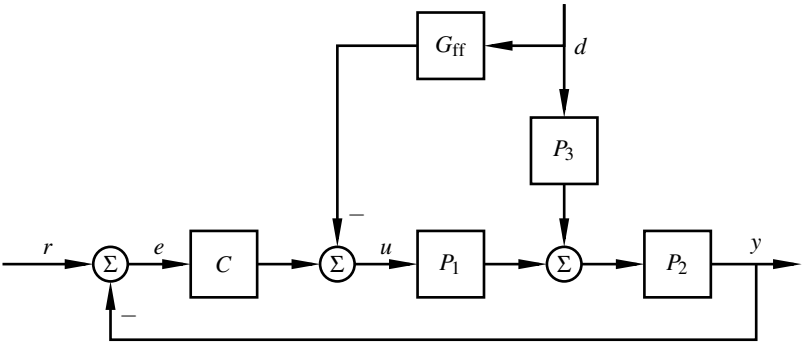

Figure 2. Closed-loop structure.

given by

$$G_{\text{cl}} = \frac{P_2P_3 + P_2P_1(CH - G_{\text{ff}})}{1 + P_2P_1C}. \tag{3}$$

Choosing H as

$$H = P_2P_3 - P_2P_1G_{\text{ff}}, \tag{4}$$

the closed loop transfer function (3) then equals

$$G_{\text{cl}} = P_2(P_3 - P_1G_{\text{ff}}) = G_{\text{o}}.$$

The closed-loop response from a disturbance d will thus be the same as the response in the open-loop case in (1) and the feedback controller, C, will not interact with the feedforward controller, G_{ff}. By using the structure in Fig. 3 with H chosen as (4) it is possible to design the feedforward controller by just considering the open-loop response from d. If the feedback controller has integral action the steady-state response will be $y = r + H(0)d$. Therefore it is desirable to choose $H(0) = 0$.

The method of subtracting the feedforward response from the controller input is common when improving system response from reference signals, cf. [Åström and Hägglund, 2006].

3. Optimal Feedforward Control

In this section optimal feedforward controller parameters, based on stable FOTDs, in the case of a step disturbance d, will be derived. Using the structure in Fig. 3 with H chosen in accordance with (4) we consider optimization over the structure in Fig. 1. The rules are derived for the case $P_2 = 1$. In applications where this is not the case, P_2 can be incorporated into P_1 and P_3 followed by first-order approximations, cf., [Åström and Hägglund, 2006]. The optimality measure is the integrated square error,

$$\text{ISE} = \|e\|_2^2 = \int_0^\infty e^2(t)\,\mathrm{d}t. \tag{5}$$

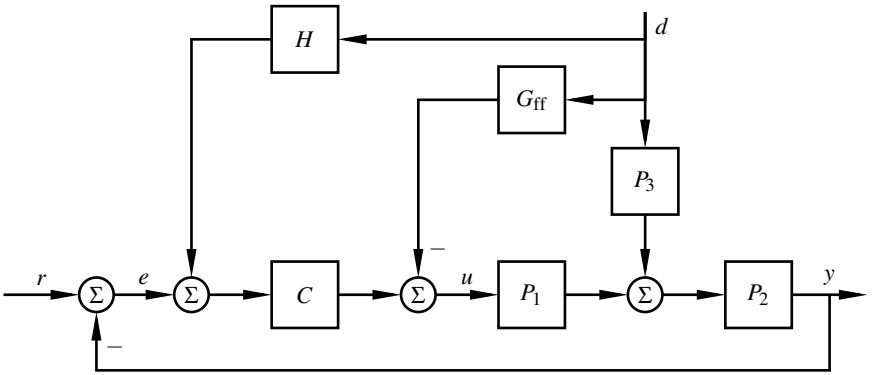

Figure 3. Modified closed-loop feedforward structure.

A vast number of other optimality criteria could be considered, cf., [Åström and Hägglund, 2006]. The ISE measure is an established performance measure and was chosen since it enables analytical solutions for finding the minimal cost for the setting considered in this paper. The drawbacks with the ISE is that it may yield large control signals and prolonged time for steady state.

The processes $P_i(s)$ are assumed to be FOTDs, i.e.,

$$\begin{aligned} P_i(s) &= \frac{K_i}{1+sT_i}e^{-L_i s}, \qquad i = 1,3 \\ L_i &\geq 0, \quad T_i > 0 \\ P_2(s) &= 1. \end{aligned} \tag{7}$$

The feedforward controller has the following structure

$$G_{\text{ff}}(s) = K_{\text{ff}}\frac{1+sT_z}{1+sT_p}e^{-sL_{\text{ff}}}. \tag{8}$$

There are in total four parameters to be determined in order to minimize (5). We require that T_p should be non-negative since negative values of T_p would give an unstable system response. For the case of $L_1 \leq L_3$ perfect feedforward, i.e., no control error, is obtained with the following choice of parameters:

$$G_{\text{ff}}(s) = \frac{K_3}{K_1}\frac{1+sT_1}{1+sT_3}e^{-(L_3-L_1)s}.$$

The following will therefore focus on the case when $L_1 > L_3$ and hence, perfect disturbance rejection is not possible. The time delays in the process models can, without loss of generality, be shifted so that $L = \hat{L}_1 = L_1 - L_3 > 0$ and $\hat{L}_3 = 0$, Furthermore the reference signal r can, without loss of generality be regarded to be zero.

Given a unit step disturbance d the output of the system is given by

$$Y(s) = (P_3(s) - P_1(s)G_{\text{ff}}(s))D(s) \tag{9}$$

where $D(s)$ is the Laplace transform of a unit step. Denote the output response by inverse Laplace transform of (9), $y(t) = \mathscr{L}^{-1}(Y(s))$. The optimization problem can be formulated as

$$\text{minimize } J = \int_L^\infty y^2(t)\,\mathrm{d}t \tag{10a}$$

$$\text{s.t.} \quad T_p \geq 0 \tag{10b}$$

$$L_{\text{ff}} \geq 0. \tag{10c}$$

(10b) and (10c) are included in the optimization formulation to ensure a stable and causal feedforward controller.

3.1 Optimal Feedforward Time Delay

Assume that the time delays are such that perfect disturbance rejection is not possible. Adding time delay in the feedforward controller would increase the time in which there is no control action and thus increase the ISE. The time delay should therefore be chosen as

$$L_{\text{ff}} = \max(0, L_3 - L_1). \tag{11}$$

3.2 Optimal Stationary Gain

In order to ensure that $H(0) = 0$ and for the integral (10a) to converge the gain in the feedforward controller has to be chosen as

$$K_{\text{ff}} = \frac{K_3}{K_1}. \tag{12}$$

3.3 Optimal T_z

Evaluating (10a) yields an expression with the following structure

$$J(T_p, T_z) = q_1 T_z^2 + q_2 T_z + q_3. \tag{13}$$

Introducing

$$a = \frac{T_1}{T_3} \tag{14a}$$

$$b = a\,(a+1)\,e^{\frac{L}{T_3}}, \tag{14b}$$

the expressions for q_1, q_2 and q_3 can be seen in Appendix, (39). Since (39a) is positive, by the assumptions in (7), (13) has a unique minimum with respect to T_z which can be determined by completion of squares:

$$J(T_p, T_z) = q_1(T_z + \frac{q_2}{2q_1})^2 - \frac{q_2^2}{4q_1} + q_3$$

for which the minimum occurs at

$$T_z(T_p) = -\frac{q_2}{2q_1} = \frac{(b-2a)T_3 + bT_p}{b(T_3+T_p)}(T_p + aT_3). \tag{15}$$

The optimal T_z can also be expressed as

$$T_z(T_p) = (T_p + T_1)\left(1 - \frac{2T_3^2}{(T_1+T_3)(T_3+T_p)e^{\frac{L}{T_3}}}\right).$$

By using the optimal T_z, (13) reduces to

$$J(T_p, T_z(T_p)) = \hat{J}(T_p) = q_3 - \frac{q_2^2}{4q_1}, \tag{16}$$

see (40) for complete expression, from which the last controller parameter, T_p, is to be determined. Since T_z is dependent of T_p it is not clear at this moment that $T_z > 0$, i.e., that the controller will be minimum-phase. This will be shown in Sec. 3.8.

3.4 Optimal T_p

Differentiating (16) yields

$$\begin{aligned}\frac{\mathrm{d}\hat{J}}{\mathrm{d}T_p} = \frac{K_3^2 a^2 T_3^2}{2b^2(T_3+T_p)^3(aT_3+T_p)^2} \\ \times \left((4a^2-2a-b)T_3^2 + 2T_pT_3(3a-1-b) - (b-2)T_p^2\right) \\ \times \left((2a+b)T_3 - (b-2)T_p\right).\end{aligned} \tag{17}$$

Equating (17) to zero to find the stationary points yields the following three:

$$T_{p_1}^* = \frac{3a-1-b+\sqrt{(a-1)^2(1+4b)}}{b-2}T_3 \tag{18a}$$

$$T_{p_2}^* = \frac{3a-1-b-\sqrt{(a-1)^2(1+4b)}}{b-2}T_3 \tag{18b}$$

$$T_{p_3}^* = \frac{2a-b}{b-2}T_3. \tag{18c}$$

The optimal choice for T_p will either be one of the three stationary points or the boundary, $T_p = 0$.

The boundary point as $T_p \to \infty$ is in practice the same as no feedforward and will therefore be discarded as a possible solution since

$$\lim_{T_p \to +\infty} J = +\infty.$$

The following subsections are devoted to finding which of the solutions that is optimal. A summary of the resulting, optimal, algorithm can be found in Sec. 4.

3.5 Conditions for Positive Stationary Points

To fulfill (10b) we only consider a stationary point (18) as a candidate for optimality if it is positive.

Case I: $T_{p_1}^* > 0$**.** From (18a), we can conclude that the denominator is positive if $b > 2$. Note that $e^{\frac{L}{T_3}} > 1 \Leftrightarrow b > a(a+1)$. Denote the numerator of (18a) by n_1 i.e.,

$$n_1 = 3a - 1 - b + \sqrt{(a-1)^2(1+4b)}.$$

In order to determine the sign of $T_{p_1}^*$ we first examine when n_1 changes its sign.

$$\begin{aligned}
&n_1 = 0 \Leftrightarrow \\
&b + 1 - 3a = \sqrt{(a-1)^2(1+4b)} \Leftrightarrow \\
&b^2 - 2\left(2a^2 - a + 1\right)b + 4a\left(2a - 1\right) = 0 \Rightarrow \\
&b_1 = 2 \\
&b_2 = a\left(4a - 2\right).
\end{aligned}$$

Assume $a < 1$. Then $a(4a-2) < a(a+1)$. Hence, both numerator and denominator of $T_{p_1}^*$ can only change signs at $b = 2$. By evaluating n_1 for $a < 1$ and for arbitrary $b \neq 2$ we can conclude that $T_{p_1}^*$ is negative for $a < 1$.

Assume instead $a > 1$. Then $a(a+1) > b_1$ and n_1 can only change its sign for $b = b_2$. Furthermore, the denominator is positive for $a > 1$ since $b > 2$. By evaluation of n_1 for arbitrary $a > 1$ and $b < a\left(4a - 2\right)$ we can conclude that $n_1 > 0$. Since the denominator is positive for $a > 1$, $T_{p_1}^*$ is positive if

$$b < a\left(4a - 2\right) \Leftrightarrow e^{\frac{L}{T_3}} < \frac{4a-2}{a+1}.$$

This means that $T_{p_1}^*$ is positive when

$$T_1 > T_3 \text{ and } L < T_3 \ln\left(\frac{4a-2}{a+1}\right). \tag{19}$$

Case II: $T_{p_2}^* > 0$. Denote the numerator and denominator in (18b) by n_2 and d_2 respectively. The numerator is negative since

$$n_2 = 3a - 1 - b - \sqrt{(a-1)^2(1+4b)} < 3a - 1 - a(a+1) - \sqrt{(a-1)^2(1+4b)}$$
$$= -(a-1)^2 - \sqrt{(a-1)^2(1+4b)} < 0.$$

The sign of $T_{p_2}^*$ is thus only dependent on d_2. Since $b > a(a+1)$, d_2 will be positive for $a > 1$. For $a < 1$,

$$T_{p_2}^* > 0 \Leftrightarrow d < 0 \Leftrightarrow b < 2 \Leftrightarrow e^{\frac{L}{T_3}} < \frac{2}{a(a+1)}.$$

To summarize, $T_{p_2}^*$ is positive when

$$T_1 < T_3 \text{ and } L < T_3 \ln\left(\frac{2}{a(a+1)}\right). \tag{20}$$

Case III: $T_{p_3}^* > 0$. By inspection of (18c) we can conclude that $T_{p_3}^*$ is positive if and only if $a < 1$ and

$$\frac{2}{a+1} < e^{\frac{L}{T_3}} < \frac{2}{a(a+1)}.$$

$T_{p_3}^*$ is positive when

$$T_1 < T_3 \text{ and } T_3 \ln\left(\frac{2}{a+1}\right) < L < T_3 \ln\left(\frac{2}{a\,(a+1)}\right). \tag{22}$$

3.6 Conditions for Optimal T_p

A stationary point, $T_{p_i}^*$, is a local minimizer if and only if the second derivative of (16) with respect to T_p is positive, i.e., $\frac{\mathrm{d}^2\hat{J}}{\mathrm{d}T_p^2} > 0$. Since the cost function (16) has three stationary points and approaches infinity when T_p approaches infinity, the cost function can have no more than two local minima.

Solution 1. From the inequalities (19), (20) and (22) we can conclude that if $a > 1$, $T_{p_1}^*$ is the only positive stationary point. Since (19) is the only stationary point for $a > 1$, this stationary point cannot be a maximum since (10a) approaches infinity when T_p approaches infinity. Furthermore,

$$\frac{\mathrm{d}\hat{J}}{\mathrm{d}T_p}(0) = K_3^2 \frac{(b-2a)(b+2a-4a^2)}{2b^2}. \tag{23}$$

If $a > 1$, then $b > 2a$ and subsequently

$$\frac{\mathrm{d}\hat{J}}{\mathrm{d}T_p}(0) < 0 \Leftrightarrow b + 2a - 4a^2 < 0 \Leftrightarrow e^{\frac{L}{T_3}} < \frac{4a-2}{a+1}. \tag{25}$$

From (19) and (25) we therefore conclude that $T_{p_1}^*$, given by (18a), is optimal when it is positive.

Solution 2 From (19) and (20) we can conclude that $T_{p_1}^*$ and $T_{p_2}^*$ cannot simultaneously be positive. Furthermore, when $T_{p_2}^*$ is positive, $T_{p_3}^*$ is either negative or corresponds to a maximum, see the next section.

In order to determine when $T_p = T_{p_2}^*$ is a better solution than $T_p = 0$, take the difference between the corresponding costs as

$$\hat{J}(T_{p_2}^*) - \hat{J}(0) = 2aK_3^2\frac{n}{d}$$
$$\hat{J}(T_{p_2}^*) < \hat{J}(0) \Leftrightarrow 2aK_3^2\frac{n}{d} < 0$$

where expressions for n and d can be found in the Appendix, (41). From these expressions we conclude that $d > 0$. Since $T_{p_2}^*$ is negative for $a > 1$, consider only the case $a < 1$. Whether $T_{p_2}^*$ is better than $T_p = 0$ or not is determined by the sign of n. Solving the equation $n = 0$ gives the following solutions for b

$$b^* = a + \sqrt{a}. \tag{26}$$

Hence, n can only change its sign for $b = b^*$. By evaluation of n with $a < 1$ and both $b < b^*$ and $b > b^*$ we can conclude that $J(T_{p_2}^*) < J(0)$ if $a < 1$ and $b < a + \sqrt{a}$. Hence, $T_p = T_{p_2}^*$ is the optimal solution when

$$a < 1 \text{ and } e^{\frac{L}{T_3}} < \frac{\sqrt{a}+a}{a(a+1)} \Leftrightarrow L < T_3 \ln\frac{\sqrt{a}+a}{a(a+1)}.$$

Solution 3 Inserting $T_{p_3}^*$ given by (18c) into (15) yields $T_p = T_z$ i.e., the static feedforward controller

$$G_{\text{ff}}(s) = \frac{K_3}{K_1}. \tag{27}$$

The second derivative of (16) with respect to T_p evaluated in $T_p = T_{p_3}^*$ is

$$\frac{\mathrm{d}^2\hat{J}}{\mathrm{d}T_p^2}(T_{p_3}^*) = -\frac{K_3^2(b-2)^5a^2}{4(a-1)^3b^3}. \tag{28}$$

$T_{p_3}^*$ is a minimum point if (28) is greater than zero. For $a > 1$ this is equivalent to

$$a(a+1)e^{\frac{L}{T_3}} - 2 < 0 \Leftrightarrow$$
$$e^{\frac{L}{T_3}} < \frac{2}{a(a+1)} < 1.$$

Since both L and T_3 are positive this condition is never fulfilled.

For $a < 1$ we can conclude that in order for $T_{p_3}^*$ to be a minimum point the following condition must hold

$$L > T_3 \ln\left(\frac{2}{a(a+1)}\right).$$

The feedforward strategy given by (27) does not give a lower cost than the strategy given by the controller with $T_p = 0$ since

$$\hat{J}(T_{p_3}^*) - \hat{J}(0) = \frac{K_3^2(b-2a)^2}{2b^2} aT_3 \geq 0.$$

3.7 Special Cases

Two cases have been disregarded in the analysis above. Firstly, the case when $a = 1$, i.e., the process time constants are equal, and secondly, the case where $b = 2$, i.e., when the denominators of (18) are zero.

Case I: Equal time constants, $T_1 = T_3$**.** In the case of equal time constants in the processes, $a = 1$ and (16) simplifies to

$$\hat{J}(T_p) = K_3^2 \frac{T_p T_3 \left(e^{\frac{L}{T_3}} - 1\right)^2}{2\left(T_p + T_3\right) e^{\frac{2L}{T_3}}}$$

from which we conclude that $T_p = 0$ is the optimal solution since $L > 0$ by assumption.

Case II: $b = 2$**.** If $b = 2$, (17) reduces to

$$\frac{\partial \hat{J}}{\partial T_p} = K_3^2 T_3^4 \frac{\left((2a+1)T_3 + 3T_p\right)(a-1)^2}{2(T_3+T_p)^3(aT_3+T_p)^2} a^2$$

for which there is only one stationary point,

$$T_p^* = -\frac{(2a+1)}{3} T_3,$$

which is less than zero. Hence, if $b = 2$, $T_p = 0$ is the optimal solution.

3.8 Optimal T_z Revisited

Since the optimal T_z, given by (15), depends on T_p it is unclear whether T_z for some set of parameters can be negative or not. We here set out to prove that it for all process parameters will be positive. Introducing (14) in (15) yields

$$T_z = \frac{(b-2a)T_3 + bT_p}{b(T_3+T_p)}(aT_3 + T_p). \tag{29}$$

Since $T_p \geq 0$ and $b > a(a+1)$ we can conclude that $T_z > 0$ if $a > 1$.

If $a < 1$, T_z will be positive when $e^{\frac{L}{T_3}} > \frac{2}{a+1}$. When $e^{\frac{L}{T_3}} < \frac{2}{a+1}$, $T_{p_2}^*$ is the optimal solution since

$$\frac{2}{a+1} < \frac{\sqrt{a}+a}{a(a+1)}.$$

The sign of T_z is determined by the sign of

$$(b-2a)T_3 + bT_p \tag{30}$$

Inserting $T_p = T_{p_2}^*$ in (30) yields

$$\frac{T_3}{b-2}\left(-(3-a)b - 4a - b\sqrt{(a-1)^2(1+4b)}\right).$$

Recalling that

$$e^{\frac{L}{T_3}} < \frac{a+\sqrt{a}}{a(a+1)} \Rightarrow b-2 < a+\sqrt{a}-2 < 0$$

we can conclude that when $T_{p_2}^*$ is the optimal solution T_z will be positive and thus T_z will be positive for all values on the process parameter.

4. Design Summary

Below follows a summary of how to choose the parameters in the feedforward controller in order to minimize the integrated square error (10a).

1. $K_{\text{ff}} = \dfrac{K_3}{K_1}$.
2. $L_{\text{ff}} = \max(0, -L)$, $L = L_1 - L_3$.
3. - Introduce $a = \dfrac{T_1}{T_3}$ and $b = a(a+1)e^{\frac{L}{T_3}}$
   - If $a > 1$ and $b < 4a^2 - 2a$

     $$T_p = \frac{3a-1-b+\sqrt{(a-1)^2(1+4b)}}{b-2}T_3.$$
   - If $a < 1$ and $b < \sqrt{a}+a$

     $$T_p = \frac{3a-1-b-\sqrt{(a-1)^2(1+4b)}}{b-2}T_3.$$
   - Else, $T_p = 0$.
4. $T_z(T_p) = (T_p + T_1)\left(1 - \dfrac{2T_3^2}{(T_1+T_3)(T_3+T_p)e^{\frac{L}{T_3}}}\right)$.

Note that even though a small T_p can be optimal, it is not necessarily practical or possible to realize such a controller. The high-frequency gain is given by

$$K_{\text{ff}}\frac{T_z}{T_p}.$$

If the high-frequency gain is too large, choose a larger T_p and recalculate T_z until the high-frequency gain is satisfying.

5. Design Examples

EXAMPLE 1—OPEN-LOOP FEEDFORWARD CONTROL
Consider the open-loop in Fig. 1 with

$$P_1(s) = \frac{1}{1+s}e^{-0.5s}, \qquad P_2(s) = 1, \qquad P_3(s) = \frac{1}{1+2s} \tag{31}$$

and unit step d disturbing the system at $t = 1$. Using the design rule from Sec. 4 gives the following optimal feedforward controller

$$G_{\text{ff}}^{\text{ISE}}(s) = \frac{1+2.35s}{1+3.02s}. \tag{32}$$

For comparison, two other feedforward controllers are simulated. The second controller is tuned in accordance with the rule presented in [Guzmán and Hägglund, 2011]. This rule sets $T_z = T_1$ and tunes T_p in order to reduce the IAE. The IAE-reducing feedforward controller is given by

$$G_{\text{ff}}^{\text{IAE}}(s) = \frac{1+s}{1+1.71s}. \tag{33}$$

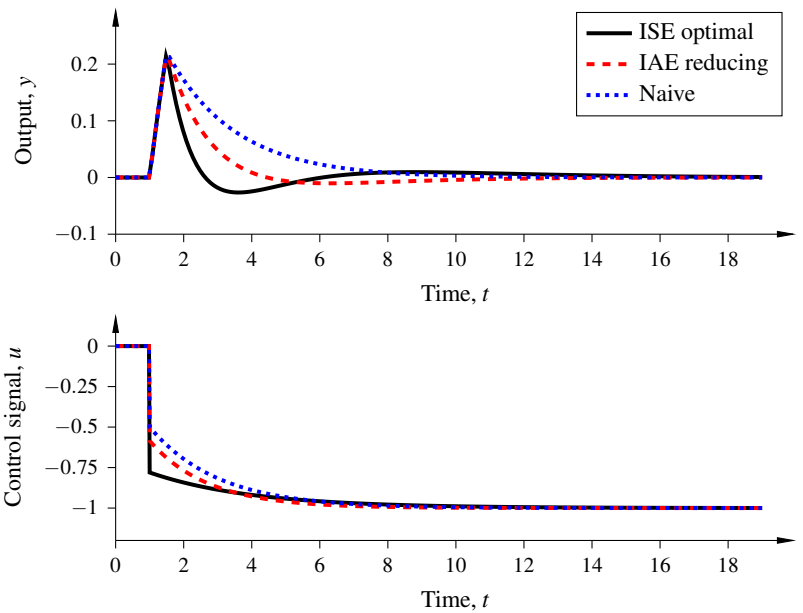

Figure 4. Output and control signals for Example 1.

Table 1. Performance measures. Ex. 1.

Strategy	ISE	IAE
$G_{\text{ff}}^{\text{ISE}}$	0.022	0.267
$G_{\text{ff}}^{\text{IAE}}$	0.034	0.313
$G_{\text{ff}}^{\text{naive}}$	0.058	0.502

Table 2. Performance measures. Ex. 2.

Strategy	ISE	IAE
$G_{\text{ff}}^{\text{ISE}}$	0.013	0.260
$G_{\text{ff}}^{\text{IAE}}$	0.021	0.346
$G_{\text{ff}}^{\text{naive}}$	0.037	0.452
No ff	1.13	3.158

The third controller is given by

$$G_{\text{ff}}^{\text{naive}}(s) = \frac{1+T_1 s}{1+T_3 s} = \frac{1+s}{1+2s}, \tag{34}$$

which is the optimal controller if the time delay is disregarded. The output signals along with the control signals can be seen in Fig. 4. The performance measures from the simulation can be seen in Table 1. The ISE-minimizing feedforward controller out-performs the two other controllers, not only in terms of ISE but also in IAE. □

EXAMPLE 2—CLOSED-LOOP WITH FOTD APPROXIMATIONS.
To examine how the design-rules handle high-order dynamics, consider the same P_1 and P_3 as in the previous example but with

$$P_2(s) = \frac{1}{0.5s+1}.$$

Incorporating P_2 into P_1 and P_3 with subsequently FOTD approximations, [Åström and Hägglund, 2006], renders the following approximations

$$\hat{P}_1 = \frac{1}{1+1.31s}e^{-0.69s}, \qquad \hat{P}_2 = 1, \qquad \hat{P}_3 = \frac{1}{1+2.25s}e^{-0.25s} \tag{35}$$

Based on these approximations a feedback controller, $C(s)$, has been tuned using the AMIGO method. The resulting PI controller is

$$C(s) = 0.38\left(1+\frac{1}{1.21s}\right).$$

The optimal feedforward controller for the process approximations (35) is given by

$$G_{\text{ff}}^{\text{ISE}}(s) = \frac{1+2.82s}{1+3.46s}. \tag{36}$$

H was based on the first-order approximations, i.e.,

$$H = \hat{P}_2\hat{P}_3 - \hat{P}_2\hat{P}_1 G_{\text{ff}}^{\text{ISE}}$$

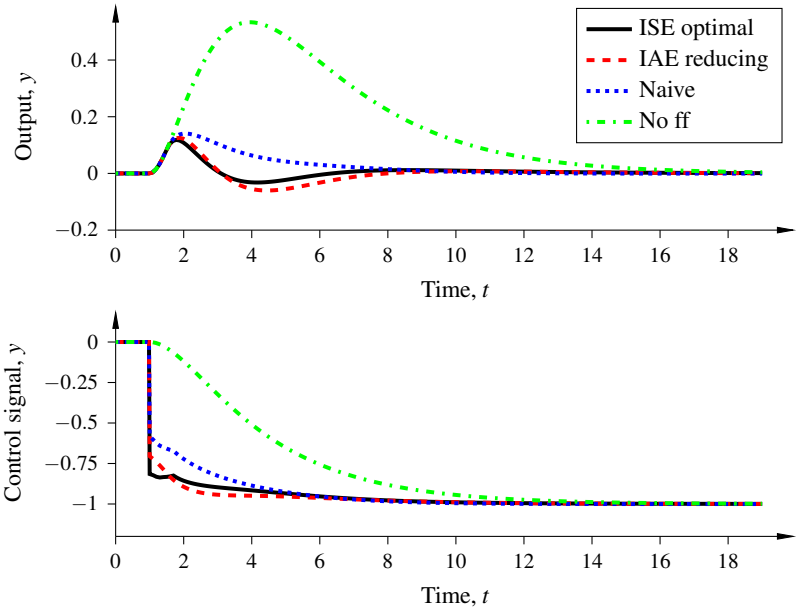

Figure 5. Output and control signals for Example 2.

As before, for comparison, the two other feedforward controllers given by

$$G_{\mathrm{ff}}^{\mathrm{IAE}}(s) = 0.99\frac{1+1.31s}{1+1.84s} \tag{37}$$

and

$$G_{\mathrm{ff}}^{\mathrm{naive}}(s) = \frac{1+1.31s}{1+2.25s}, \tag{38}$$

where (38) was used with the same structure as (36) with H as

$$H = \hat{P}_2\hat{P}_3 - \hat{P}_2\hat{P}_1 G_{\mathrm{ff}}^{\mathrm{naive}}.$$

For simulation of (37) the structure given in Fig. 2 was used. The result from simulations can be seen in Fig. 5 and the performance measures in Table 2. □

6. Conclusions

In this paper we present design rules for a lead-lag feedforward controller that minimizes the integrated squared error in the case of stable first-order process models with time delay, affected by a measurable step disturbance in an open-loop setting. A control structure that separates feedback and feedforward design has been discussed.

References

Åström, K. J. and T. Hägglund (2004). "Revisiting the Ziegler-Nichols step response method for PID control". *Journal of Process Control* **14**:6, pp. 635–650.

Åström, K. J. and T. Hägglund (2006). *Advanced PID Control*. Instrumentation, Systems, and Automation Society.

Brosilow, C. and B. Joseph (2002). *Techniques of Model-Based Control*. Prentice Hall PTR, pp. 221–239.

Garcia, C. and M. Morari (1982). "Internal model control. A unifying review and some new results". *Industrial & Engineering Chemistry Process Design and Development* **21**:2, pp. 308–323.

Guzmán, J. L. and T. Hägglund (2011). "Simple tuning rules for feedforward compensators". *Journal of Process Control* **21**:1, pp. 92–102.

Isaksson, A., M. Molander, P. Moden, T. Matsko, and K. Starr (2008). "Low-order feedforward design optimizing the closed-loop reponse". In: *Preprints, Control Systems*. Vanccouver, Canada.

Skogestad, S. (2003). "Simple analytic rules for model reduction and PID controller tuning". *Journal of Process Control* **13**:4, pp. 291–309.

Vilanova, R., O. Arrieta, and P. Ponsa (2009). "IMC based feedforward controller framework for disturbance attenuation on uncertain systems". *ISA Transactions* **48**:4, pp. 439–448.

Ziegler, J. and N. Nichols (1942). "Optimum settings for automatic controllers". *Transactions of the A.S.M.E* **5**:11, pp. 759–768.

A. Miscellaneous equations

$$q_1 = \frac{1}{2}\frac{K_3^2}{T_p + aT_3} \tag{39a}$$

$$q_2 = K_3^2 \frac{(2a-b)T_3 - bT_p}{b(T_3+T_p)} \tag{39b}$$

$$\begin{aligned} q_3 = \frac{K_3^2}{2b^2(T_3+T_p)(T_p+aT_3)} \cdot \Big(& \big(a(a+1)^2 + b(b-4a)\big)a^2T_3^3 \\ & + \big(a(a+1)^3 + b^2(a+3) - 4ab(a+2)\big)aT_pT_3^2 \\ & + \big((a(a+1)-2b)^2 + 3b^2(a-1)\big)T_p^2T_3 + b^2T_p^3\Big) \end{aligned} \tag{39c}$$

$$\hat{J}(T_p) = \frac{K_3^2 T_3 a}{2b^2(T_3+T_p)^2(T_p+aT_3)}\Big(\big(a(a+1)^2+b(b-4a)\big)T_p^3 + a^2(a-1)^2T_3^3 + \big(a(a+2)(a+1)^2 - 4ba^2 - 4a(b+1) + 2b^2\big)T_3T_p^2 + \big((2a^2-b)^2 - a(a-1)^2(2a-1)\big)T_3^2T_p\Big) \quad (40)$$

$$\hat{J}(T_{p_2}^*) - \hat{J}(0) = \frac{2K_3^2 a T_3}{(1-a)(b+1+\sqrt{1+4b})(3+\sqrt{1+4b})^2b^2} \times \Big[\big(-(16+10b)a^3 + (16+26b+10b^2)a^2 - (4+17b^2+10b+2b^3)a + 4b^2 + 5/2b^3\big)\sqrt{1+4b} - (4b^2+38b+16)a^3 + (42b^2+54b+16+4b^3)a^2 - (33b^2+4+18b+19b^3)a + (b^2+19/2b+4)b^2\Big] \quad (41)$$

Paper IV

Low-Order Feedforward Controllers: Optimal Performance and Practical Considerations

Martin Hast Tore Hägglund

Abstract

Feedforward control from measurable disturbances can significantly improve the performance in control loops. However, tuning rules for such controllers are scarce. In this paper design rules for how to choose optimal low-order feedforward controller parameter are presented. The parameters are chosen so that the integrated squared error, when the system is subject to a step disturbance, is minimized. The approach utilizes a controller structure that decouples the feedforward and the feedback controller. The optimal controller can suffer from undesirable high-frequency noise characteristics and tuning methods for how to filter the control signal are also provided. For scenarios where perfect disturbance attenuation in theory is achievable but where noise-filtering is needed, the concept of precompensation is introduced as a way to shift the controller time-delay to compensate for the low-pass filtering.

1. Introduction

Feedforward is an efficient way to reduce control errors both for reference tracking and disturbance rejection, given that the disturbances acting on the system are measurable. This paper addresses tuning of feedforward controllers for rejection of measurable disturbances. The use of feedforward alone often cannot eliminate the disturbance completely and it is therefore often used along with feedback control.

For design of PID-controllers there exists a large number of tuning methods for choosing the control parameters, see e.g., [Åström and Hägglund, 2004; O'Dwyer, 2009; Skogestad, 2003; Ziegler and Nichols, 1942]. However, there is a lack of methods for how to tune feedforward controllers in order to efficiently attenuate disturbances.

The design of low-order feedforward controllers has previously been addressed by e.g., [Guzmán and Hägglund, 2011; Hast and Hägglund, 2012; Isaksson et al., 2008; Rodríguez et al., 2013]. In [Isaksson et al., 2008] an iterative design procedure is proposed that minimizes a system norm in the frequency domain, taking the feedback controller into account. In [Guzmán and Hägglund, 2011] simple tuning rules for feedforward controllers, that reduces the integrated absolute error, are provided. An overview of low-order feedforward from both references and load disturbances are discussed in [Vilanova and Visioli, 2012].

In [Brosilow and Joseph, 2002] a feedforward structure that separates the feedback and feedforward control design, was presented. This idea has been adopted in this paper and justifies that the designed controller, while optimal in the open-loop case, gives good performance when used in conjunction with feedback control. This structure makes use of the same process models that are used for the design of the feedforward controller. The structure has similarities with Internal Model Control, IMC, see [Garcia and Morari, 1982]. Robust feedforward design within the IMC framework was addressed by [Vilanova, Arrieta, et al., 2009]. The method of subtracting the feedforward response from the controller input is common when improving system response from reference signals, cf. [Åström and Hägglund, 2006].

This paper presents an analytic solution to the problem of designing a feedforward lead-lag filter which minimizes the integrated square error when the system is subjected to a measurable step disturbance. The design rules are derived for stable process with dynamics described by first-order plants with dead time, (FOTD). The paper also discusses how a feedforward controller should be filtered in order to reduce the effect of measurement noise and aggressive controller actions. Lastly, tuning rules for reducing the control signal activity by precompensation is presented.

2. Feedforward Control

Feedforward control from measurable disturbances has usually been treated and solved as an open-loop problem. For a system described by Figure 1 the transfer

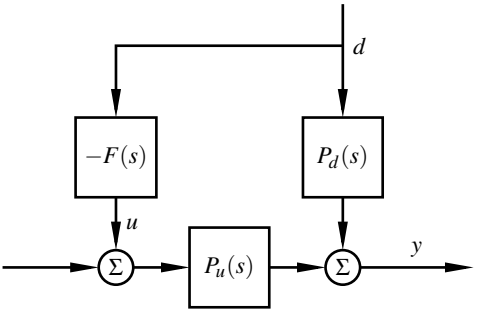

Figure 1. Open-loop structure for disturbance rejection using feedforward control.

function from the measurable disturbance d to system output y is

$$G_o(s) = P_d(s) - P_u(s)F(s). \tag{1}$$

In order to completely eliminate the effect of the disturbance d the feedforward controller should be chosen as

$$F(s) = \frac{P_d(s)}{P_u(s)}. \tag{2}$$

This controller is not realizable for instance if the time-delay in $P_u(s)$ is longer than that of $P_d(s)$ or if $P_u(s)$ has zeros in the right-half plane. If the controller is realizable it might give rise to larger control signals than what is desirable. Common remedies for this is to use a low-order approximation of (2) or even just the static gain [Åström and Hägglund, 2006]. Due to model errors, uncertainties and other disturbances than the measurable acting on the system, feedforward controllers are often used together with feedback controllers. By combining a feedforward controller with an output feedback controller with the structure in Figure 2 the transfer function from d to y becomes

$$G_{yd}(s) = \frac{P_d(s) - P_u(s)F(s)}{1 + P_u(s)C(s)}. \tag{3}$$

Using this structure, and the controller given by (2), perfect disturbance rejection is possible although the same remarks as above regarding realizability apply. It can be seen from (3) that the effect of the disturbance is now dependent on both the feedforward and the feedback controller. Thus, if perfect disturbance rejection is not possible the responses from the open- and the closed loop will differ. With the closed-loop structure, the controllers will interact which might lead to a deterioration in performance compared to the open-loop structure. The remedies for this can be divided into two categories. Firstly, the feedforward controllers can be tuned, taking the feedback controller into account. Ways of modifying the feedforward controller in order to get satisfying response from the closed-loop system has been

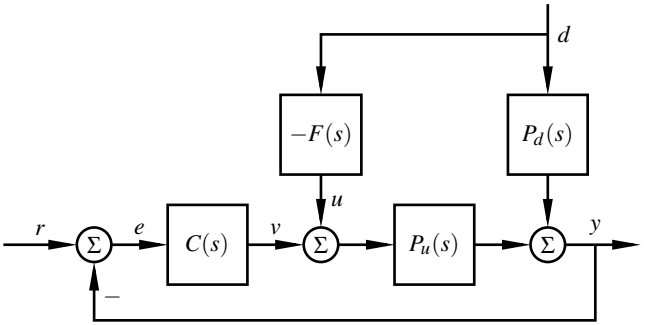

Figure 2. Closed-loop structure commonly used when combining feedforward and feedback control.

presented in [Guzmán and Hägglund, 2011] and [Isaksson et al., 2008]. The drawback with these kind of approaches is that if the feedback controller is retuned, the feedforward controller needs to be retuned as well. Secondly, the effect of the interaction can be decreased or even eliminated by yet another feedforward to the feedback controller input, see Figure 3. The advantage with this method is that the feedforward and the feedback controllers can be tuned individually. A drawback is the increased overall complexity of the controller. A feedforward control structure, equivalent to the one in Figure 3, that achieves the desired decoupling was presented in [Brosilow and Joseph, 2002]. Dropping the argument s, the transfer function from d to y is given by

$$G_{\text{cl}} = \frac{P_d - P_u(F - CH)}{1 + P_u C}. \tag{4}$$

Choosing H as

$$H = P_d - P_u F, \tag{5}$$

the closed loop transfer function (4) then equals

$$G_{\text{cl}} = P_d - P_u F = G_{\text{o}}.$$

The closed-loop response from a disturbance d will thus be the same as the response in the open-loop case in (1) and the feedback controller, C, will not interact with the feedforward controller, F. By using the structure in Figure 3 with H chosen as (5) it is possible to design the feedforward controller by just considering the open-loop response from d.

3. Optimal Feedforward Control

Tuning rules for both feedback and feedforward controllers are often based on process models with low complexity, see e.g., [Åström and Hägglund, 2006; Guzmán

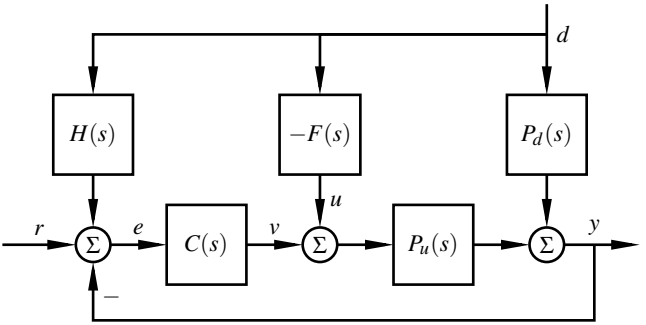

Figure 3. Controller structure that decouples the response, and design, of the feedback and feedforward controllers.

and Hägglund, 2011; Rodríguez et al., 2013; Sell, 1995]. In this section, feedforward controller parameters will be derived in the same spirit. The parameters will be obtained as the solution to an optimization problem that ensures that the system has good disturbance attenuation.

The considered feedforward controller is a lead-lag filter, or equivalently, a low-pass filtered PD-controller, with possibly a time delay i.e.

$$F(s) = K_{\text{ff}} \frac{1 + sT_z}{1 + sT_p} e^{-sL_{\text{ff}}} \tag{6}$$

with in total four parameters to choose. Deriving simple analytic tuning rules that are optimal for more advanced feedforward controllers are not tractable.

By using the controller structure in Figure 3, the feedforward controller can be designed for the open-loop case depicted in Figure 1.

Let the processes P_u and P_d be described by

$$P_u(s) = \frac{K_u}{1 + sT_u} \mathrm{e}^{-sL_u}, \quad P_d(s) = \frac{K_d}{1 + sT_d} \mathrm{e}^{-sL_d}. \tag{7}$$

In the analysis and derivation of optimal controller parameters the disturbance d is assumed to be a unit step. The time response of the system subject to the disturbance can be calculated as $y(t) = \mathscr{L}^{-1}\left(G_{\text{o}}(s)\frac{1}{s}\right)$. The reference signal can without loss of generality be disregarded and for the remainder of this paper r is zero.

The performance is measured by the integrated squared error

$$\text{ISE} = \int_0^\infty e^2(t)\,\mathrm{d}t. \tag{8}$$

A large number of other measures could be considered, cf., [Åström and Hägglund, 2006]. The ISE measure is an established performance measure and is chosen since

it enables the derivation of analytical solutions for finding the minimal cost for the setting considered in this paper. The drawbacks of using the ISE as the performance measure is that it penalizes large deviations from the reference hard which gives optimal controllers that may yield large control signals and overshoot in the measured variable.

In the case of $L_u \leq L_d$ perfect feedforward, i.e., no control error, is obtained with the, realizable, controller given by (2). The remainder of this section will therefore focus on the case when $L_u > L_d$ and hence, perfect disturbance rejection is not possible. The time delays in the process models can then, without loss of generality, be shifted so that $L_d = 0$ and the delay in P_u becomes

$$L = L_u - L_d. \tag{9}$$

Optimal feedforward controller parameters that give good disturbance attenuation can be found by solving the nonconvex optimization problem

$$\begin{aligned} \underset{K_{\text{ff}}, L_{\text{ff}}, T_z, T_p}{\text{minimize}} \quad & J = \int_L^\infty y^2(t)\,\mathrm{d}t \\ \text{subject to} \quad & T_p \geq 0 \\ & L_{\text{ff}} \geq 0 \end{aligned} \tag{10}$$

where the constraints are included to ensure that the controller is stable and causal.

The remainder of this section contains the derivation of the optimal controller parameters. A summary of the result can be found in Sec. 4.

3.1 Optimal Stationary Gain

Using the decoupling structure and provided that the feedback controller has integral action, the steady-state response is $y = r + H(0)d$. It is therefore desirable to ensure that $H(0) = 0$. Furthermore, the integral in (10) converges if and only if the controller's stationary gain is

$$K_{\text{ff}} = \frac{K_d}{K_u}. \tag{11}$$

From (5) it then follows that $H(0) = 0$.

3.2 Feedforward Time Delay

If the time delays are such that perfect disturbance rejection is not possible, the ISE will increase if there is time-delay in the controller. The time delay should therefore be chosen as

$$L_{\text{ff}} = \max(0, -L). \tag{12}$$

3.3 Optimal T_z

Using the optimal static gain (11) and time delay (12) it was shown in [Hast and Hägglund, 2012] that J is a convex quadratic function in the parameter T_z. The expression for J can be found in the Appendix, see (84). The unique global minimizer can be found by completion of squares and it is given by

$$T_z^* = (T_u + T_p)\left(1 - \frac{2T_u}{b(T_d + T_p)}\right) \tag{13}$$

where

$$a = \frac{T_u}{T_d} \tag{14a}$$

$$b = a(a+1)\,\mathrm{e}^{\frac{L}{T_d}}. \tag{14b}$$

Since the optimal choice of T_z is a function of T_p it is not obvious that $T_z^* > 0$, i.e. that the controller will be minimum-phase. That this is in fact the case will be shown in Sec. 5.2.

3.4 Optimal T_p

Denote the cost function evaluated at the optimal K_{ff}, L_{ff} and T_z by

$$\hat{J}(T_p). \tag{15}$$

The expression for $\hat{J}(T_p)$ can be found in the Appendix, see (85). The optimal choice of feedforward time constant T_p will either be one of the stationary points of (15) or the boundary point, $T_p = 0$. From an optimization point of view this will be considered to be a feasible solution. This corresponds to the feedforward controller being an ideal PD-controller. The limit as $T_p \to \infty$ is practically the same as no feedforward and will therefore be discarded as a possible solution.

The stationary points to (15) are given as the solutions to

$$\frac{\mathrm{d}\hat{J}}{\mathrm{d}T_p} = 0 \tag{16}$$

and are

$$T_{p_1}^* = \frac{3a - 1 - b + \sqrt{(a-1)^2(1+4b)}}{b-2} T_d \tag{17a}$$

$$T_{p_2}^* = \frac{3a - 1 - b - \sqrt{(a-1)^2(1+4b)}}{b-2} T_d \tag{17b}$$

$$T_{p_3}^* = \frac{2a - b}{b-2} T_d. \tag{17c}$$

Substituting (17c) in (13) gives $T_z^* = T_p^*$, i.e. static feedforward compensation. Define the cost difference between an arbitrary choice of T_p and the boundary controller as

$$D(T_p) = \hat{J}(T_p) - \hat{J}(0). \tag{18}$$

The expression for the cost difference can be found in the Appendix, see (87). The static controller gives a higher cost than the PD-controller since

$$D(T_{p_3}^*) = \frac{K_d^2(b-2a)^2}{2b^2} T_u \geq 0 \tag{19}$$

unless $T_u = T_d$ and $L = 0$ i.e. the scenario where the process dynamics allows for perfect disturbance rejection. Therefore, the stationary point $T_{p_3}^*$ can be excluded as the optimal solution.

Conditions for Nonnegative Stationary Points. To ensure a stable feedforward controller the time constant T_p must be nonnegative. It was shown in [Hast and Hägglund, 2012] that $T_{p_1}^*$ is nonnegative if and only if

$$a > 1, \quad b < 4a^2 - 2a \tag{20}$$

and that $T_{p_2}^*$ is nonnegative if and only if

$$a < 1, \quad b < 2. \tag{21}$$

Simplification of stationary point T_p. Since $T_{p_1}^*$ and $T_{p_2}^*$ are nonnegative for $a > 1$ and $a < 1$ respectively, the expressions given by (17a) and (17b) can be simplified to the single expression

$$T_p^* = \frac{3a - 1 - b + (a-1)\sqrt{1+4b}}{b-2} T_d. \tag{22}$$

Conditions for Optimal T_p. The reduced cost function, (15), has three stationary points and approaches infinity when T_p approaches infinity. The cost function can therefore have no more than two local minima. According to (19), $T_{p_3}^*$ renders a higher cost than the boundary $T_p = 0$ and is therefore excluded. Only one of $T_{p_1}^*$ and $T_{p_2}^*$ is positive for any set of process parameters. The optimal solution must therefore be T_p^* or $T_p = 0$.

By determining for which positive T_p that (16) changes its sign, conditions for when T_p^* is optimal can be derived. The difference function can be expressed as

$$D(T_p) = \tilde{K}(T_p)\left(T_p^2 + c_1 T_p + c_0\right) T_p \tag{23}$$

where $\tilde{K}(T_p)$ is positive. The expressions for the $\tilde{K}$, c_1 and c_0 can be found in the Appendix, see (88). The positive solution to $D(T_p) = 0$ is

$$T_p = -\frac{c_1}{2} + \sqrt{\frac{c_1^2}{4} - c_0}. \tag{24}$$

Inserting (22) and solving for b gives the following values for which the difference function can change its sign

$$b^* = \begin{cases} 4a^2 - 2a \\ a + \sqrt{a} \\ a - \sqrt{a}. \end{cases} \tag{25}$$

The last of the three solutions can be disregarded since $a - \sqrt{a} < b$ for all values of a and L.

For $a < 1$, the first solution can also be disregarded since $4a^2 - 2a < b$. Furthermore, since

$$\left.\frac{\mathrm{d}D(T_p^*)}{\mathrm{d}b}\right|_{b=a+\sqrt{a}} > 0 \tag{26}$$

the difference function changes sign from negative to positive. The choice of T_p if $a < 1$, is hence,

$$T_p = \begin{cases} T_p^* & \text{if} \quad b < a + \sqrt{a} \\ 0 & \text{if} \quad b \geq a + \sqrt{a}. \end{cases} \tag{27}$$

For $a > 1$, the second solution can be disregarded since $a + \sqrt{a} < b$ and since

$$\left.\frac{\mathrm{d}D(T_p^*)}{\mathrm{d}b}\right|_{b=4a^2-2a} > 0, \tag{28}$$

the sign of the difference function changes from negative to positive for $b = 4a^2 - 2a$. The optimal choice of T_p if $a > 1$ is

$$T_p = \begin{cases} T_p^* & \text{if} \quad b < 4a^2 - 2a \\ 0 & \text{if} \quad b \geq 4a^2 - 2a. \end{cases} \tag{29}$$

The magnitude of the two conditions on b in (27) and (29) are related as

$$4a^2 - 2a < a + \sqrt{a} \Leftrightarrow a < 1. \tag{30}$$

Therefore the optimal choice can be expressed as

$$T_p = \begin{cases} T_p^* & \text{if} \quad b < \begin{cases} 4a^2 - 2a & \text{or} \\ a + \sqrt{a} \end{cases} \\ 0 & \text{otherwise.} \end{cases} \tag{31}$$

3.5 Special cases

There are three parameter combinations that are not treated in the analysis above. The first is the case when $T_d = 0$. In this case, the ISE defined in (8) is equal to

$$J = K_d^2 \frac{T_u^2 + (3T_p - 2T_z)T_u + (T_p - T_z)^2}{2(T_u + T_p)}. \tag{32}$$

It can easily be verified that the ISE is zero if the controller parameter are chosen as $T_p = 0$ and $T_z = T_u$.

The second and third overlooked parameter combinations are when the time constants are equal, that is $T_u = T_d$ and subsequently $a = 1$, and when $b = 2$. It has been shown in [Hast and Hägglund, 2012] that the optimal controller parameters are given by $T_p = 0$ and T_z chosen as (13).

4. Design Summary

Below follows a summary of how to choose the feedforward controller parameters so that they minimize the ISE and are the solution to the optimization problem formulated in Sec. 3. Calculate the delay difference

$$L = L_u - L_d. \tag{33}$$

If it is negative, perfect disturbance rejection is possible with the controller

$$F = \frac{K_d}{K_u}\frac{1+sT_u}{1+sT_d}\mathrm{e}^{-s(L_d-L_u)}. \tag{34}$$

If the delay difference is positive, the optimal ISE controller is obtained by choosing the controller parameters as

1. $K_{\mathrm{ff}} = \dfrac{K_d}{K_u}$.
2. $L_{\mathrm{ff}} = 0$.
3. - Calculate $a = T_u/T_d$ and $b = a(a+1)\mathrm{e}^{\frac{L}{T_d}}$.
   - If $b < 4a^2 - 2a$ or $b < a + \sqrt{a}$
     $$T_p = \frac{3a - 1 - b + (a-1)\sqrt{1+4b}}{b-2}T_d.$$
   - Otherwise, $T_p = 0$.
4. $T_z = (T_p + T_u)\left(1 - \dfrac{2T_u}{b(T_d + T_p)}\right)$.

Note that even though a small T_p can be optimal, it is not necessarily practical or possible to realize such a controller. Considerations related to the controllers noise characteristics and realizability are presented in Sec. 6.

5. Optimal Feedforward Controller Characteristics

This section will provide an illustration of how the optimal controller parameters depend on the process parameters.

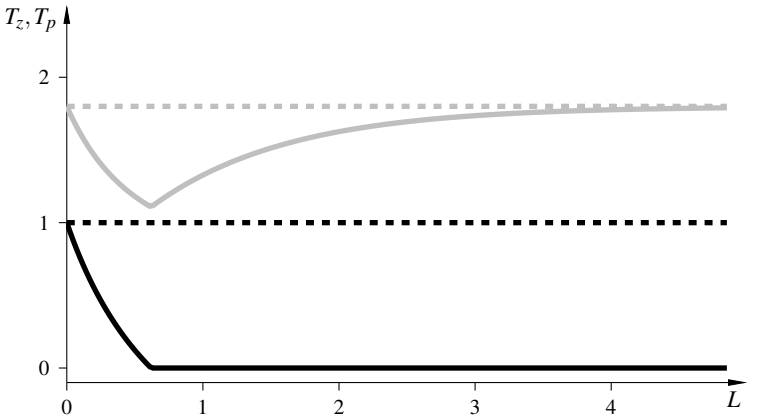

Figure 4. Controller parameters as functions of the delay difference L for $T_u = 1.8$, $T_d = 1$. The solid lines correspond to the optimal controller parameters and the dashed lines correspond to the common controller (35). T_p and T_z in black and gray respectively.

5.1 Lead-lag characteristics

The optimal controller will be compared with the commonly used feedforward controller given by (2), where the non-realizable part is discarded i.e.

$$F^0 = \frac{K_d}{K_u}\frac{1+T_u s}{1+T_d s}. \tag{35}$$

By not taking the delay difference into account, the time constants alone determine if this controller will have lead or lag characteristics. A feedforward controller will have lead characteristics if $T_z > T_p$ and a lag characteristics if $T_z < T_p$. It is apparent from (35) that this controller switches from a lead- to a lag-filter for $a = 1$. For the optimal controller it is straight-forward, using (13) and (31), to show that $T_z < T_p$ if and only if $a < 1$ and $b < a + \sqrt{a}$. Examples of the optimal controller parameters as functions of the process parameters can be seen in Figure 4 and Figure 5. These figures show how the optimal controller parameters depend on the the time-delay difference L.

For $L = 0$ it is easily verified, using (14) and (22) that $T_p^* = T_d$. Furthermore, its derivative with respect to L is

$$\frac{\mathrm{d}T_p^*}{\mathrm{d}L} = (1-a)\frac{\left(2b+5+3\sqrt{1+4b}\right)b}{\sqrt{1+4b}\,(b-2)^2}, \tag{36}$$

from which it, together with (31), can be concluded that $T_p^* > T_d$ if and only if $a < 1$ and $b < \sqrt{a} + a$. This can be seen in Figure 4 and Figure 5.

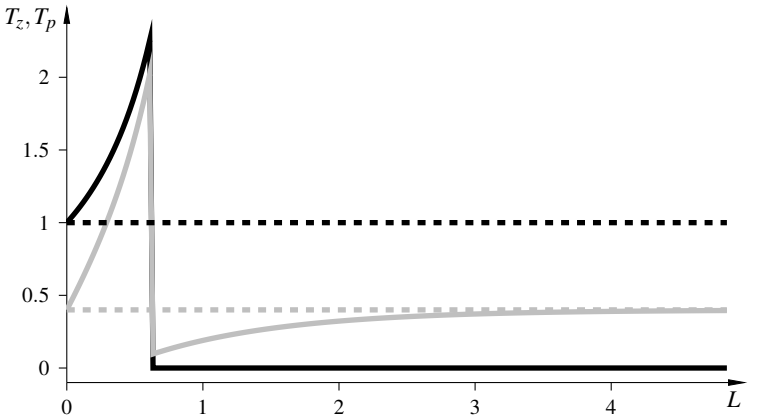

Figure 5. Controller parameters as functions of the delay difference L for $T_u = 0.4$, $T_d = 1$. The full lines correspond to the optimal controller parameters and the dashed lines correspond to the common controller (35). T_p and T_z in black and gray respectively.

5.2 Nonnegativity of the optimal T_z.

It follows from (13) that T_z^*, is nonnegative if and only if

$$(b-2a)T_d + bT_p \geq 0. \tag{37}$$

Furthermore, from the definition of b (14) it also follows that

$$a \geq 1 \Rightarrow b \geq 2a \tag{38}$$

and thus, the optimal T_z is positive for $a \geq 1$.

For time-constant ratios $a < 1$ and long time delay differences L, the optimal value of T_p is zero according to (31). In this case T_z is nonnegative since

$$b > \sqrt{a} + a > 2a \tag{39}$$

and the inequality (37) is satisfied.

For the last case, $a <$ and $b < \sqrt{a} + a$, the optimal T_z is positive since

$$(b-2a)T_d + bT_p > a(a+1)(T_d + T_p) - 2aT_d > 0 \tag{40}$$

where the first inequality follows from the definition of b and that L is positive. The second inequality holds since, $T_p^* > T_d$. Using T_p^* will therefore give an optimal T_z that is positive.

Asymptotic Controller Parameters

If the delay difference L is zero, the expressions for the optimal controller parameters simplify $T_p = T_d$ and $T_z = T_u$ i.e. the controller given by (2) is realizable and the disturbance will not give rise to any control error. The design rules for the optimal controller behaves as expected for $L = 0$. This is also true for the case of very large time-delay difference.

According to the definition of b, (14), and optimal choice of T_p, (31), $T_p = 0$ is optimal if the time-delay difference L is sufficiently large. From (13) it can be shown that

$$\lim_{L\to\infty} T_z = T_u. \tag{41}$$

This means that if the time-delay difference is large no additional lag is introduced by the feedforward controller and the pole in P_u is canceled. This can be seen in the figures 4 and 5.

6. Control Signal Considerations

The optimal controller proposed minimizes the integrated-squared error but can be sensitive to high-frequency noise and give rise to large control signals. The high-frequency gain of the controller is

$$K_{\text{ff}} \frac{T_z}{T_p} \tag{42}$$

and using the optimal controller parameters can give too large, or even infinite, high-frequency gain. The unit step response of the controller is

$$u(t) = -K_{\text{ff}} \left(1 + \frac{(T_z - T_p)\, \mathrm{e}^{-t/T_p}}{T_p} \right) \tag{43}$$

from which it can be concluded that the largest magnitude of the signal is equal to the high-frequency gain given by (42). A large ratio between T_z and T_p can be undesirable for two reasons; it yields large control signals and, can amplify and feed measurement noise into the feedback loop.

To limit the effect of high-frequency noise and to get a smoother control signal, the feedforward controller (6) is augmented with a second-order low-pass filter;

$$F_f(s) = K_{\text{ff}} \frac{1 + sT_z}{(1 + sT_p)} \frac{1}{\left(1 + sT_f\right)^2} \mathrm{e}^{-sL_{\text{ff}}}. \tag{44}$$

We propose that the parameters K_{ff}, L_{ff}, T_z and T_p are chosen in accordance to what is optimal for a controller such as (6). The filter time constant T_f is then chosen so

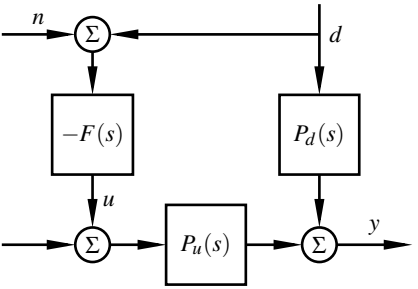

Figure 6. Open-loop structure with measurement noise, n, on the controller input.

that the noise propagation through the controller and control signal activity is satisfactory. The order of the low-pass filter is chosen so that the feedforward controller has roll-off also when T_p is zero.

With the added high-frequency roll-off, control action will be smoother and the wear on the actuator will decrease.

Assuming additive white noise corruption, n, on the controller input, see Figure 6, the variance of controller output is

$$\mathrm{var}\,(u) = \frac{K_{\mathrm{ff}}}{4T_f}\left(1 + \frac{T_z^2 - T_p^2}{\left(T_f + T_p\right)^2}\right). \tag{45}$$

This can be shown, using methods from [Åström, 1970], to be equal to

$$\int_0^\infty \left(\frac{\mathrm{d}u(t)}{\mathrm{d}t}\right)^2 \mathrm{d}t \tag{46}$$

which is a measure of the actuator movement during a unit-step disturbance. By increasing T_f, the control signal will be smoother and less aggressive, possibly at the expense of decreased performance.

According to (31), the optimal choice of T_p is zero if b is large which is implied if the delay difference is large. For $T_p = 0$, two approaches to chose T_f are presented below. Firstly, it will be shown how to choose T_f to limit the peak in the control signal. The tuning rule is derived, as were the ISE-optimal tuning rules, for a unit step-disturbance. Secondly, it will be shown how to choose T_f to limit the peak in the controller's Bode magnitude plot.

High-frequency noise, arising from the measurements of the disturbance, will be attenuated if H is strictly proper. However, if this noise is not sufficiently attenuated by H, the feedback controller should have roll-off to prevent the noise from being amplified and fed into the feedback system. For PID-controllers the necessity of filtering and how the filter should be designed have been treated in [Larsson and Hägglund, 2011].

The introduction of the low-pass filter also makes it meaningful to address the problem of noise when perfect feedforward is possible. This will be treated in Sec. 7.

6.1 Filter choice for $T_p = 0$.

The design rules for an optimal lead-lag feedforward controller given in Sec.4 will for long time delays state that T_p is zero. The feedforward controller is then an ideal PD-controller and a special case of (44) given by

$$F_0(s) = K_{\text{ff}} \frac{1 + sT_z}{\left(1 + sT_f\right)^2} e^{-sL_{\text{ff}}}. \tag{47}$$

For this controller, considerations regarding the control signal characteristics can be derived without approximations. This is also the worst case scenario from a control signal perspective since

$$|F_0(i\omega)| \geq \left|F_f(i\omega)\right|. \tag{48}$$

Limiting the control signal peak The filter time constant can be chosen such that the control signal peak when the system is subject to a step disturbance is smaller than some user-specified value. The control signal, using (47), subject to a unit-step disturbance is

$$u(t) = -K_{\text{ff}} \left(1 - \frac{T_f^2 - \tau(T_z - T_f)}{T_f^2} \mathrm{e}^{-\frac{\tau}{T_f}}\right) \tag{49}$$

where $\tau = t - L_{\text{ff}}$. Differentiating by, and solving for τ, the control signal peak can be be shown to be obtained at

$$t_{\text{peak}} = \frac{T_z T_f}{T_z - T_f} + L_{\text{ff}}. \tag{50}$$

The peak in the control signal is

$$u\left(t_{\text{peak}}\right) = -K_{\text{ff}} \left(1 + \frac{T_z - T_f}{T_f} \mathrm{e}^{\frac{T_z}{T_f - T_z}}\right). \tag{51}$$

By introducing $x = T_f/(T_z - T_f)$, the expression for the peak being Δ times larger than K_{ff}, can be expressed by

$$u\left(t_{\text{peak}}\right) = -K_{\text{ff}} \left(1 + \frac{1}{x} \mathrm{e}^{-(x+1)}\right) = -K_{\text{ff}}\Delta. \tag{52}$$

The equation can be rewritten as

$$x\mathrm{e}^x = \frac{\mathrm{e}^{-1}}{\Delta - 1} \tag{53}$$

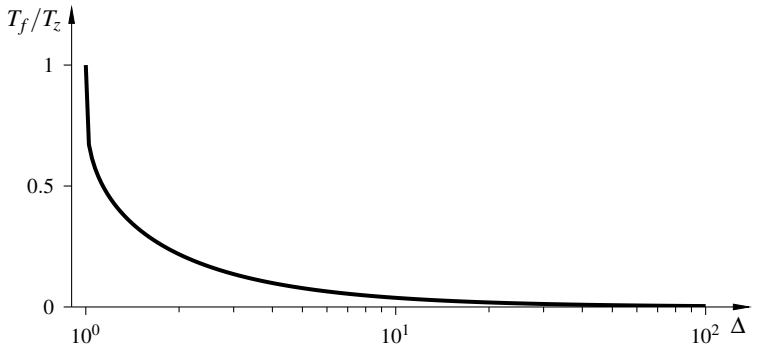

Figure 7. The ratio between T_f and T_z as a function of the desired control signal peak magnitude Δ.

for which the solution is given by the Lambert W-function i.e.

$$x = W\left(\frac{\mathrm{e}^{-1}}{\Delta - 1}\right), \quad \Delta > 1. \tag{54}$$

By assumption, Δ is larger than one and the argument to the Lambert W-function is therefore positive and real, and the solution to (53) is therefore given by the principal branch W_0 of the W function. See [Corless et al., 1996] for an introduction to, a brief history, and computational aspects of the Lambert W-function. To obtain a control signal with a peak Δ the filter time-constant should be chosen as

$$T_f = \frac{T_z}{1 + \frac{1}{W_0\left(\frac{\mathrm{e}^{-1}}{\Delta - 1}\right)}}. \tag{55}$$

The ratio between the filter time-constant and T_z is thus a function of the desired control signal peak magnitude Δ. This function is displayed in Figure 7. It can be seen in the figure that for example choosing the filter time constant as a fifth of T_z yields a control signal peak approximately twice as large as the static gain.

6.2 Bode magnitude peak reduction

The filter time-constant can also be chosen such that the largest value of the Bode magnitude-plot is equal to or lower than a desired value. The maximum of the Bode magnitude is

$$\gamma_0 = \max_{\omega} |F_0(i\omega)| \tag{56}$$

and the maximum occurs at the frequency

$$\omega^* = \frac{\sqrt{T_z^2 - 2T_f^2}}{T_z T_f}, \tag{57}$$

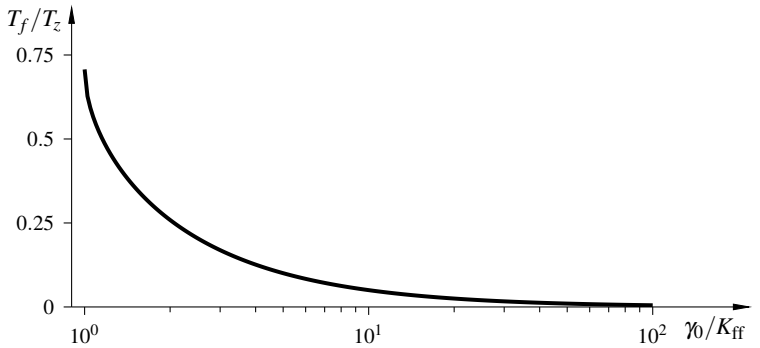

Figure 8. The ratio between T_f and T_z as a function of the desired Bode peak magnitude γ_0/K_{ff}.

which means that ω^* is the only positive real solution to

$$\frac{\mathrm{d}F_0(i\omega)}{\mathrm{d}\omega} = 0. \tag{58}$$

The magnitude of the Bode plot peak at the frequency ω^* is

$$\gamma_0 = K_{\text{ff}} \frac{T_z^2}{2T_f\sqrt{T_z^2 - T_f^2}}. \tag{59}$$

Solving this equation for T_f gives

$$T_f = \frac{T_z}{\sqrt{2}}\sqrt{1 - \sqrt{1 - \frac{K_{\text{ff}}^2}{\gamma_0^2}}}, \qquad \gamma_0 > K_{\text{ff}}, \tag{60}$$

which can be used as a design rule to choose the filter time-constant. The condition that γ_0 should be larger than K_{ff} is necessary since if $T_f > T_z/\sqrt{2}$ the maximum will be K_{ff} and occur at $\omega = 0$. Figure 8 shows how the ratio between T_f and T_z depends on the desired Bode peak magnitude.

Filter choice for $T_p > 0$. For the lead-lag feedforward controller (6), the high-frequency gain (42) is finite if $T_p > 0$ but it can still be too large. If $T_z > T_p$ the high frequency gain will be larger than K_{ff}. Deriving an analytic solution to the problem of limiting the Bode magnitude is not tractable and therefore an approximation is derived and presented in this section.

The magnitude of the controller's Bode plot is

$$\left|F_f(i\omega, T_f)\right| = K_{\text{ff}}\sqrt{\frac{1+\omega^2T_z^2}{1+\omega^2T_p^2}} \cdot \frac{1}{1+\omega^2T_f^2}. \tag{61}$$

A necessary condition for the Bode plot to have a peak larger than K_{ff} is that $T_z > T_p$. The peak will be located at the positive real solution to

$$\frac{\mathrm{d}\left|F_f(i\omega, T_f)\right|}{\mathrm{d}\omega} = 0. \tag{62}$$

Straight-forward but tedious calculations give that a positive real solution exists if and only if

$$0 < T_f < \sqrt{\frac{T_z^2 - T_p^2}{2}} = \hat{T}_f. \tag{63}$$

If noise conditions or constraints on the control signals aggressiveness are such that the filter time constant has to be chosen larger than this upper bound, the benefit of using the optimal parameter values diminishes to a level where a second order low-pass filter should be considered as the feedforward controller.

Denote the peak of the Bode magnitude plot

$$\gamma(T_f) = \max_{\omega} \left|F_f(i\omega, T_f)\right| \tag{64}$$

which has the the following boundary conditions

$$\begin{aligned} &\gamma(0) = K_{\text{ff}}\tfrac{T_z}{T_p}, \quad &&\gamma'(0) = -K_{\text{ff}}\tfrac{2\hat{T}_f}{T_p^2}, \\ &\gamma(\hat{T}_f) = K_{\text{ff}}, \quad &&\gamma'(\hat{T}_f) = 0. \end{aligned} \tag{65}$$

As an approximation of the peak consider the function

$$\tilde{\gamma} = K_{\text{ff}} \frac{b_1 T_f + b_2}{T_f^2 + b_3 T_f + b_4} \tag{66}$$

where the parameters b_i are determined so that the approximation has the same boundary conditions as (65). This results in

$$\tilde{\gamma} = K_{\text{ff}} \frac{\hat{T}_f T_f + \frac{1}{2} T_z (T_z + T_p)}{T_f^2 - \hat{T}_f T_f + \frac{1}{2} T_p (T_z + T_p)}. \tag{67}$$

Denote the peak magnitude relative to the static gain by λ, i.e. $\lambda = \tilde{\gamma}/K_{\text{ff}}$. By solving (67) for T_f, the filter time-constant can be determined by

$$T_f = \frac{\hat{T}_f(1+\lambda)}{2\lambda} \sqrt{1 - \frac{2\lambda(T_z - \lambda T_p)(T_z + T_p)}{(1+\lambda)^2 \hat{T}_f^2}}. \tag{68}$$

An example of the true Bode magnitude peak and its approximation can be seen in Figure 9 where $T_z = 1$, $T_p = 0.5$. The approximation is close to the true value of the Bode magnitude peak.

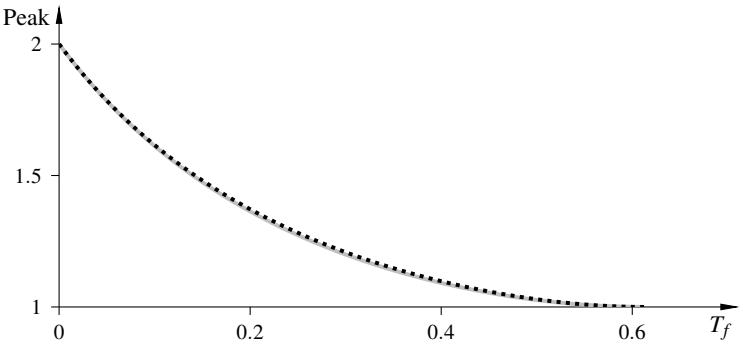

Figure 9. The true Bode peak magnitude (gray) and the approximation (black) given by (68) as functions of the filter time constant T_f. The controller parameters are $K_{\text{ff}} = 1$, $T_z = 1$ and $T_p = 0.5$.

7. Precompensation

Without the noise-reducing filter, perfect disturbance rejection is possible with the controller (2) if $L_u \leq L_d$. However, this controller can suffer from noise sensitivity and aggressive control action. Adding the low-pass filter will reduce these problems but it will also deteriorate the performance.

With the introduction of the low-pass filter to the optimal controller, additional lag is introduced. However, if the optimal controller contains a time delay, this can be adjusted in order to reduce the lag from the filter. Consider the controller (44) with the time delay

$$L_{\text{ff}} = L_d - L_u + \delta \tag{69}$$

where δ denotes the time-delay shift. Denote the integrated-squared error (8) obtained with this controller J_f. This is a convex function of δ since

$$\frac{\mathrm{d}^2 J_f}{\mathrm{d}\delta^2} = \frac{2K_d^2 T_d^2 (T_d + T_z)}{\mathrm{e}^{\frac{\delta}{T_d}} (T_p + T_d) \left(T_f + T_d\right)^2 (T_u + T_d)} \geq 0$$

and hence the unique minimizer is given by the only stationary point J_f i.e.

$$\delta^* = \ln\left(\frac{2T_d^3 (T_d + T_z)}{(T_f + T_d)^2 (T_p + T_d)(T_u + T_d)}\right) T_d. \tag{70}$$

By using the T_p and T_z that are optimal in the case without the low-pass filter, this expression simplifies to

$$\delta^* = 2\ln\left(\frac{T_d}{T_f + T_d}\right) T_d. \tag{71}$$

This indicates that the delay in the controller should be decreased when filtering is introduced. The time delay L_{ff} in the controller must be positive and it follows from (69) that

$$\delta \geq L_u - L_d. \tag{72}$$

Hence, there exists a bound on the filter time-constant T_f for which the controller delay will be zero. Introducing (71) into the inequality (72) and solving for the filter time constant gives

$$T_f \leq T_d \left(\mathrm{e}^{\frac{L_d - L_u}{2T_d}} - 1 \right). \tag{73}$$

For reasonable amounts of filtering, the time-delay shift rules provided in (70) and (71) reclaims some of the performance lost by the introduction of the low-pass filter.

8. Design Examples

In this section two examples are presented to illustrate that the ideas presented in this paper also work well in a closed-loop setting where the process models differ from the actual processes. The first examples compares the different control structures as well as makes use of the design considerations from Sec. 6. The last example shows that the concepts of precompensation is an easy way to increasing performance when the closed-loop decoupling structure is used.

EXAMPLE 1—CLOSED-LOOP PERFORMANCE

This example illustrates how the feedforward structure presented in Figure 3 and the design rules presented in this paper performs in a setting with noise and uncertain process models . It also shows the benefits of feedforward control and especially feedforward control using the decoupling structure.

Consider the controller structure in Figure 3 with the processes given by

$$P_u(s) = \frac{1}{(1+s)^3}, \quad P_d(s) = \frac{1}{(1+0.1s)^2} \tag{74}$$

and their FOTD-approximations

$$P_u^*(s) = \frac{1}{1+2.45s}\mathrm{e}^{-0.81s}, \quad P_d^*(s) = \frac{1}{1+0.19s}\mathrm{e}^{-0.03s}.$$

To simulate a scenario where the process knowledge is limited, the controllers will be designed based on the approximations.

The feedback controller is a PI controller, where the controller parameters are found using the method in [Hast, Åström, et al., 2013], given by

$$C(s) = 0.55 + \frac{0.27}{s}. \tag{75}$$

Using the design rules in Sec. 4 yields the, nonrealizable, feedforward controller

$$F^0 = 1 + 2.44s. \tag{76}$$

To obtain a realizable controller that doesn't suffer from high-frequency noise amplification, a filter is augmented to the feedforward controller. The filter time-constant is chosen so that the control signal peak is $\Delta = 5$ using (55). The feedforward controller with low-pass filter is

$$F = \frac{1 + 2.44s}{(1 + 0.19s)^2}. \tag{77}$$

Assuming that perfect process knowledge is not available, the decoupling filter is based on the approximations i.e.

$$H = P_u^* - P_d^* F. \tag{78}$$

The system is simulated using the processes (74) with band-limited white noise added to the measurements of the disturbance d. The noise power is $5 \cdot 10^{-9}$ and the sample time is 10^{-4} units. The disturbance is a unit-step entering at $t = 1$. In order to compare the performance of the controller and the structures, the results from three simulations can be seen in Figure 10. The gray curves correspond to pure feedback control, i.e. $F = 0$ and $H = 0$ and are provided for reference. The black dash-dotted curves correspond to feedforward control using the controller (77) with $H = 0$, i.e. no control action decoupling. Finally, the black solid curves correspond to a simulation where both the feedforward controller (77) and the decoupler (78) was used. The measurement noise on the feedforward controller input is effectively attenuated to such a degree that it is not visible in the control signal, Figure 10.

As can be seen from Table 1, the introduction of feedforward action increases the ability to reject the disturbance in measures of ISE but not integrated absolute error, IAE. Although the feedforward controller would perform well in an open-loop setting, in closed-loop the interaction with the feedback controller yields large control signals and a significant undershoot in the disturbance response. The decoupling filter increases performance as well as makes it possible to re-tune either the feedforward and feedback controllers without the need to re-tune the other. The output from the feedback controller can be seen in Figure 11. Comparing the black curves, it can be concluded that the introduction of the decoupler reduces the control action from the feedback controller. With the decoupling filter, the feedback controller only acts on the mismatch between the model and the process. □

Table 1. Performance measures for the control strategies in Example 1.

Scenario	ISE	IAE	u_{peak}
FB only	1.75	2.91	1.49
FB and FF	1.30	2.95	5.62
FB, FF and decoupling	0.86	1.91	4.99

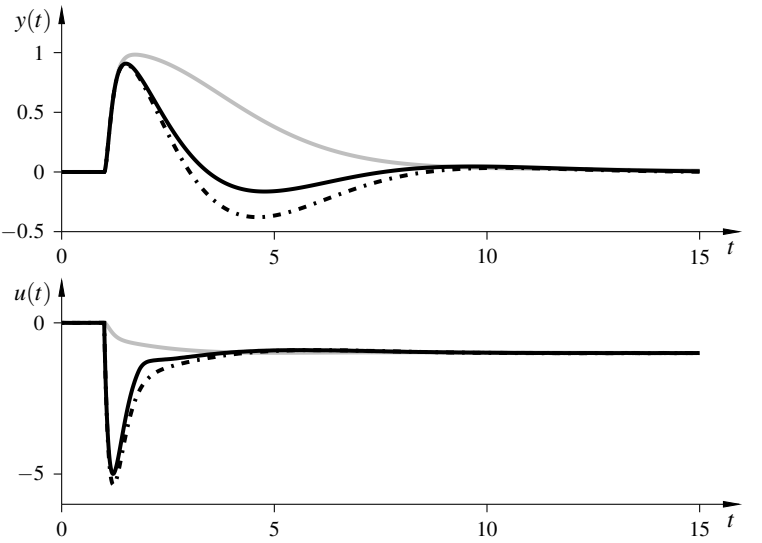

Figure 10. Output (upper) and control signals (lower) for the scenarios in Example 1. The solid gray curves correspond to only feedback control, the dash-dotted black curves to feedback and feedforward control using the conventional controller structure, Figure 2, and the solid black to feedback and feedforward control using the decoupling structure, Figure 3.

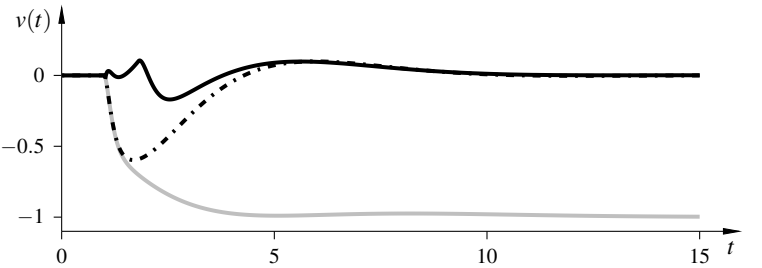

Figure 11. Feedback controller output for the scenarios in Example 1. The solid gray curves correspond to only feedback control, the dash-dotted black curves to feedback and feedforward control using the conventional controller structure, Figure 2, and the solid black to feedback and feedforward control using the decoupling structure, Figure 3.

EXAMPLE 2—CLOSED-LOOP PERFORMANCE USING PRECOMPENSATION
This example will show that the concept of precompensation works well in a closed-loop setting where the process' dynamics are not fully known. Consider a system with the structure as in Figure 3 with the same process dynamics as the Example 1 except for an added time delay of two time units in the disturbance dynamics i.e.

$$P_u(s) = \frac{1}{(1+s)^3}, \quad P_d(s) = \frac{1}{(1+0.1s)^2} e^{-2s} \tag{79}$$

with their FOTD-approximations

$$P_u^*(s) = \frac{1}{1+2.45s} e^{-0.81s}, \quad P_d^*(s) = \frac{1}{1+0.19s} e^{-2.03s}.$$

The same feedback controller, (75), that was used in Example 1 can be used since P_u is unchanged. Using the design rules presented in Sec. 4, the feedforward controller is

$$F_0 = \frac{1+2.45s}{1+0.19s} e^{-1.22s}. \tag{80}$$

The high-frequency gain of this controller is 12.9, which will be the amplification of high-frequency measurement noise. To limit the noise amplification, low-pass filtering is introduced. To limit the largest value of the Bode magnitude to approximately $\lambda = 5$, (68), is used to calculate a suitable filter time-constant as $T_f = 0.22$. The filtered controller is thus given by

$$F_f = \frac{1+2.45s}{(1+0.19s)(1+0.22s)^2} e^{-1.22s}. \tag{81}$$

To counter-act the additional lag introduced by the low-pass filter, the time delay may be shifted according to (70), i.e. $\delta = -0.27$ the delay-shifted controller is then given by

$$F_\delta = \frac{1+2.45s}{(1+0.19s)(1+0.22s)^2} e^{-0.94s}. \tag{82}$$

The three controllers were tested in simulation with the same measurement noise as in the previous example with a unit-step disturbance entering at $t = 0$. In each simulation the decoupling filter was chosen according to (5) as

$$H = P_d^* - P_u^* F. \tag{83}$$

The processes used in the simulations are the ones given by (79). The results from the simulations can be seen in Figure 12 where the gray curves correspond to the nonfiltered controller, the dash-dotted to the filtered and the black solid to the filtered and delay-shifted controller. The performance measures associated with the simulations can be seen in Table 2. Introducing the low-pass filter reduces the noise-amplification significantly as well as the initial peak in the control signal. However, this comes at the expense of lost performance both in terms of ISE and IAE. By

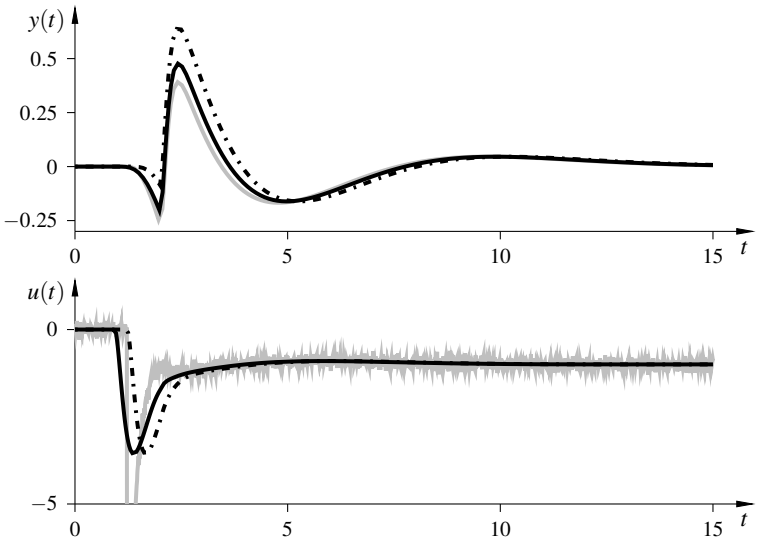

Figure 12. Output (upper) and control signals (lower) using the three controllers presented in Example 2. The gray solid curves correspond to the controller without filtering (80), the dash-dotted black curves to the low-pass filtered controller (81), and the solid black curves to the low-pass filtered controller with precomensation, (82).

Table 2. Performance measures for the control strategies in Example 2.

Controller	ISE	IAE	u_{peak}
$F_0(s)$	0.18	1.13	13.3
$F_f(s)$	0.37	1.38	3.5
$F_\delta(s)$	0.23	1.20	3.5

shifting the time-delay in the controller, the performance loss can be reduced while the good noise-properties of the filtered controller remain. This example shows that the method of precompensation works well also in scenarios where the process knowledge is limited and feedback is used together with feedforward. □

9. Conclusions

This paper describes how to tune low-order feedforward controllers in order to minimize the integrated squared error for measurable step disturbances. The de-

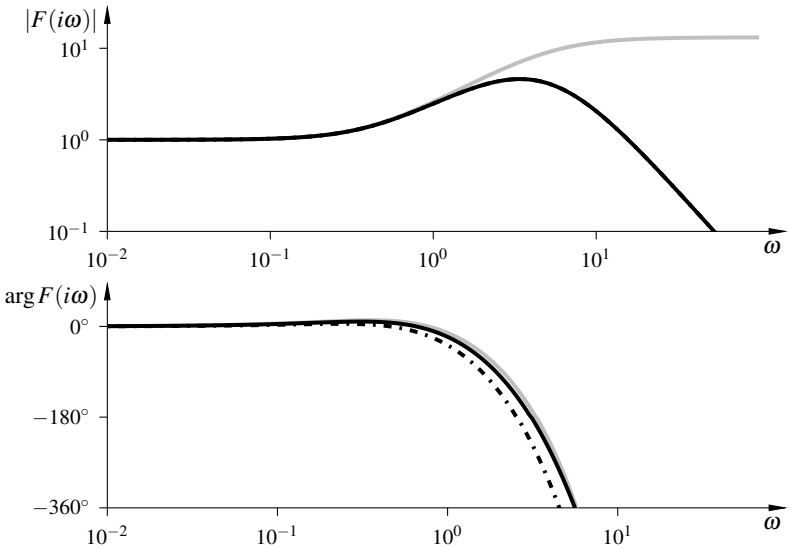

Figure 13. Bode diagram for the three feedforward controllers in Example 2. The gray solid curves correspond to the controller without filtering (80), the dash-dotted black curves to the low-pass filtered controller (81), and the solid black curves to the low-pass filtered controller with precomensation, (82).

sign rules are derived for an open-loop structure with FOTD plant models. The open-loop controller structure is motivated by the use of a decoupling structure that enables the feedforward and the feedback controllers to be tuned separately. The resulting response to measurable disturbances is that of the open-loop. The paper also describes the characteristics of the optimal controller parameters as functions of the plant model parameters. The optimal feedforward controller can suffer from large high-frequency gain and noise sensitivity which can result in large unwanted control action. To reduce the control action and noise amplification it is proposed that the optimal feedforward controller is low-pass filtered. The filter proposed is a second order filter that assures that the controller has roll-off and thus attenuates high-frequency noise that arises for example from measurement noise.

A number of design methods for choosing the filter time-constant is also proposed. Design rules to limit the peak in both the Bode magnitude or in control signal are provided. Examples show that the controller structure used and the design rules provided gives good performance also in settings with process uncertainties. By filtering the ISE-optimal controllers the control signal characteristics can be significantly improved in terms of variance and aggressiveness with a reasonable loss in performance.

For situations were the disturbance can be completely rejected, design rules for

how the time-delay can be shifted in order to compensate low-pass filtering, was provided. This approach of precompensation was shown in examples to give significant performance improvements in open-loop settings in scenarios where filtering of the measurable disturbance was needed. The approach was also tested in simulations in closed-loop with uncertain processes, where it also gave an increase in performance.

10. Acknowledgements

This work was carried out within the framework of the Process Industrial Centre at Lund University (PIC-LU) and supported by the Swedish Foundation for Strategic Research (SSF). The authors are members of the LCCC Linnaeus Center and the eLLIIT Excellence Center at Lund University.

References

Åström, K. J. (1970). *Introduction to Stochastic Control Theory*. Vol. 70. Academic Press, New York.

Åström, K. J. and T. Hägglund (2004). "Revisiting the Ziegler-Nichols step response method for PID control". *Journal of Process Control* **14**:6, pp. 635–650.

Åström, K. J. and T. Hägglund (2006). *Advanced PID Control*. Instrumentation, Systems, and Automation Society.

Brosilow, C. and B. Joseph (2002). *Techniques of Model-Based Control*. Prentice Hall PTR, pp. 221–239.

Corless, R. M., G. H. Gonnet, D. E. Hare, D. J. Jeffrey, and D. E. Knuth (1996). "On the Lambert-W function". *Advances in Computational mathematics* **5**:1, pp. 329–359.

Garcia, C. and M. Morari (1982). "Internal model control. A unifying review and some new results". *Industrial & Engineering Chemistry Process Design and Development* **21**:2, pp. 308–323.

Guzmán, J. L. and T. Hägglund (2011). "Simple tuning rules for feedforward compensators". *Journal of Process Control* **21**:1, pp. 92–102.

Hast, M., K. J. Åström, B. Bernhardsson, and S. Boyd (2013). "PID design by convex-concave optimization". In: *Proceedings European Control Conference*, pp. 4460–4465.

Hast, M. and T. Hägglund (2012). "Design of optimal low-order feedforward controllers". In: *IFAC Conference on Advances in PID Control*. Brescia, Italy.

Isaksson, A., M. Molander, P. Moden, T. Matsko, and K. Starr (2008). "Low-order feedforward design optimizing the closed-loop reponse". In: *Preprints, Control Systems*. Vanccouver, Canada.

Larsson, P. O. and T. Hägglund (2011). "Control signal constraints and filter order selection for PI and PID controllers". In: *2011 American Control Conference*. San Francisco, California, USA.

O'Dwyer, A. (2009). *Handbook of PI and PID Controller Tuning Rules, 3rd edn.* Imperial College Press London. ISBN: 978-1-85233-7092-5.

Rodríguez, C., J. L. Guzmán, M. Berenguel, and T. Hägglund (2013). "Generalized feedforward tuning rules for non-realizable delay inversion". *Journal of Process Control* **23**:9, pp. 1241–1250.

Sell, N. J. (1995). *Process control fundamentals for the pulp and paper industry*. Technical Association of the Pulp & Paper Industry. Chap. 6, pp. 221–226. ISBN: 0-899852-294-3.

Skogestad, S. (2003). "Simple analytic rules for model reduction and PID controller tuning". *Journal of Process Control* **13**:4, pp. 291–309.

Vilanova, R., O. Arrieta, and P. Ponsa (2009). "IMC based feedforward controller framework for disturbance attenuation on uncertain systems". *ISA Transactions* **48**:4, pp. 439–448.

Vilanova, R. and A. Visioli (2012). *PID Control in the Third Millennium: Lessons Learned and New Approaches*. Springer.

Ziegler, J. and N. Nichols (1942). "Optimum settings for automatic controllers". *Transactions of the A.S.M.E* **5**:11, pp. 759–768.

A. Supplementary Equations

Equations that are related to the analysis of the optimal feedforward controller are presented here for completeness.

With $K_{\text{ff}} = K_d/K_u$ and $L_{\text{ff}} = 0$ the cost function (10) can be expressed as

$$J = \int_L^\infty y^2(t)\,\mathrm{d}t = K_d^2\left(\alpha_2 T_z^2 + \alpha_1 T_z + \alpha_0\right) \tag{84}$$

where the coefficients

$$\alpha_0 = \frac{T_d}{2\mathrm{e}^{\frac{2L}{T_d}}} + \frac{T_p^2 + 3T_uT_p + T_u^2}{2\left(T_u + T_p\right)} - 2a\frac{T_pT_u + T_d(T_p + T_u)}{b(T_p + T_d)}$$

$$\alpha_1 = \frac{2T_u}{b\left(T_p + T_d\right)} - 1$$

$$\alpha_2 = \frac{1}{2\left(T_u + T_p\right)}$$

are functions of T_p. The cost function is minimized with respect to T_z by $T_z = -\frac{\alpha_1}{2\alpha_2}$. With this choice of T_z the cost function reduces to

$$\hat{J}(T_p) = K_d^2 \left(\alpha_0 - \frac{\alpha_1^2}{4\alpha_2} \right). \tag{85}$$

The cost for the boundary controller, $T_p = 0$, is

$$\hat{J}(0) = \frac{K_d^2 a^2 (a-1)^2 T_d}{2b^2}. \tag{86}$$

The difference function (18) is given by

$$D(T_p) = \tilde{K}(T_p) \left(T_p^2 + c_1 T_p + c_0 \right) T_p \tag{87}$$

where

$$\begin{aligned} \tilde{K}(T_p) &= \frac{K_d^2 T_u (b-2a)^2}{2b^2 (T_u + T_p)(T_p + T_d)^2} \\ c_1 &= \frac{4T_d \left(a^3 + (2-b)a^2 - (b+1)a + \frac{1}{2}b^2 \right)}{(b-2a)^2} \\ c_0 &= \frac{T_d^2 \left(8a^3 - 4a^2 + b^2 - 4a^2 b \right)}{(b-2a)^2}. \end{aligned} \tag{88}$$

Paper V

Feedforward controller design using convex optimization and tuning rules for proportional set-point weighting

Martin Hast Tore Hägglund

Abstract

In this work we present a method for design of low-order feedforward controllers from both reference signal and measurable disturbance. The feedforward controllers from reference are equivalent to the use of a PID controller with set-point weighting. The design problem is formulated as a convex optimization problem and then solved for a batch of process models. The optimal proportional set-point weights are then used to derive tuning rules that minimize the integrated absolute error. Examples illustrate the usefulness of the proposed method and tuning rules.

Submitted to *IET Control Theory & Applications.*

1. Introduction

Tracking of reference and attenuation of disturbances are the core in the control of a system. Because of uncertainties in the model describing the process and disturbances this is most conveniently handled using a feedback controller. The tracking and disturbance rejection can be improved without sacrificing robustness by introducing filters or controllers that has an open-loop impact on the control system. Ideally those feedforward controllers should invert the process dynamics so that we obtain perfect tracking and disturbance rejection. This is unfortunately not possible in general due to e.g., nonminimum-phase behavior, time delays and saturation in the actuators. This paper describes a method for designing feedforward controllers by solving a suitable convex optimization problem that will handle also those cases. The same optimization problem will be solved for a batch of processes and the results will be used to formulate tuning rules for how to chose the proportional set-point weight for PI and PID controllers.

In the literature concerning design of low-order feedforward controllers intended for use in connection to PID controllers there seem to a bias towards feedforward action from the reference signal. This is most probable due to the fact that the reference signal, in contrast to disturbances in general, is measurable and feedforward therefore is possible.

Convex optimization has evolved into a mature branch in the tree of optimization disciplines, see [Boyd and Vandenberghe, 2004] and its many references. The book [Boyd and Barratt, 1991] provides a thorough analysis of linear controller design and different specifications that could be considered. It also provides a treatment of controller design with connections to convex optimization.

A variety of easy-to-use tools are available for a number of different platforms and in several programming languages, for instance the MATLAB toolboxes CVX, [Research, 2012; Grant and Boyd, 2008] and YALMIP [Lofberg, 2004], and the Python software package CVXOPT [M. Andersen et al., 2011; M. S. Andersen et al., 2013].

The design of feedforward compensators, assuming that the process is described by a first-order model plus dead-time, FOTD, has been addressed in an number of papers. In [Visioli, 2004] a nonlinear feedforward reference control scheme is used to obtain good tracking followed by a PID design that ensures good robustness. A tuning rule for a low-order controller that gives optimal rejection of measurable disturbances is derived in [Hast and Hägglund, 2014]. The method presented there is independent on the feedback controller but relies on first-order plus dead-time (FOTD) process models. That paper also discusses some of the practical issues that noise filtering introduce and how these can be solved. Optimal low-order feedforward from measurable disturbance, taking the feedback controller into account, is presented in [Isaksson et al., 2008]. Low-order feedforward from measurable disturbances, for processes described by FOTD models, has been addressed in [Guzmán and Hägglund, 2011]. The tuning rule presented there cancels the process pole and

the remaining parameters are chosen so that the integrated absolute error is minimized. In [Vilanova and Visioli, 2012] another method for tuning feedforward controllers based on low-order process models is presented. Our method, presented in this paper, does not rely on a specific model structure.

The use of feedforward control in order to increase performance in the scope of model predictive control was examined in [Carrasco and Goodwin, 2011].

Inversion-based feedforward methods, where the reference signal is altered so that the transition between to set-points behaves in an optimal fashion, have been presented, for example, in the case of linear minimum-phase systems [Piazzi and Visioli, 2001b], linear nonminimum-phase systems [Piazzi and Visioli, 2001a] and nonlinear systems [Graichen et al., 2005].

Methods for feedforward controller design for multiple-inputs multiple-outputs (MIMO) plants have been addressed by e.g, [Prempain and Postlethwaite, 2001] where controllers are designed using Youla parameterizations in the robustness H_∞-framework. The method presented there also applies to gain-scheduled feedforward controllers for linear parametric varying plants. In [Piccagli and Visioli, 2009] an optimal feedforward control design, from the reference signal, for MIMO-plants described by FOTD models was presented. The feedforward signals are determined so that the system outputs achieve predefined transition times. Our method aims at providing a simple optimization method that minimizes IAE that also can take constraints on error and control signal into account as well as to provide tuning rules for how to chose the set-point weights.

The problem of designing robust feedforward controllers using convex optimization when the poles of the controllers are fixed has been presented in [Giusto, Neto, et al., 1996] where they minimize the H_∞ and H_2 norms. This result was extended to the case of uncertainties in [Giusto and Paganini, 1999]. Design of robust feedforward controllers where the poles of the controllers are part of the optimization can be found in [Scorletti and Fromion, 2006] and [Kose and Scherer, 2007]. These papers focus on minimization of H_∞ and H_2 norms and do not take the controller signal into account.

Set-point weighting for MIMO plants by formulating and solving an optimization problem with bilinear matrix inequalities has been addressed by [Bianchi et al., 2008]. Their design method relies on transformation of the problem to a problem of static output feedback and minimization of H_∞ and H_2 norms. Our method minimizes the integrated absolute error and admits constraints on both the error and control signal responses.

A similar method as the one presented here, in the case for reference feedforward controller design using a quadratic cost function has been addressed in [Leva and Bascetta, 2007]. Our method admits any convex cost function and provides tuning rules for the proportional set-point weight.

The paper is organized as follows. In Section 2 we define the controller structure, models, controllers and relevant signals. In Section 3 aspects of error minimization is discussed and how signals can be sampled to obtain a tractable optimization prob-

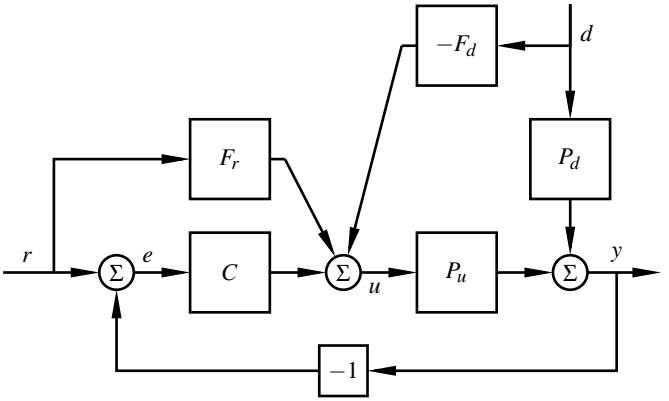

Figure 1. Control structure.

lem. In Section 4 we discuss how design specifications can be turned into a convex optimization problem and give an example of how the problem of finding optimal set-point weights can be formulated. The AMIGO process batch is presented in Section 5. The optimal solution to the problem of minimizing IAE for all 134 processes in the batch are presented and tuning rules are derived. The tuning rules are also verified and discussed. In the last section we provided some conclusions.

2. Plant Models, Controllers and Signals

We denote the system output by $y(t)$, the control signal by $u(t)$, the set-point or reference by $r(t)$ and the measurable disturbance by $d(t)$. The error is given by $e(t) = r(t) - y(t)$. The feedback controller $C(s)$, the feedforward controllers $F_r(s)$ and $F_d(s)$, the process $P_u(s)$ and disturbance dynamics $P_d(s)$ are connected as in Figure 1. We assume that all processes and controllers are described by single-input single-output, linear time-invariant models.

We assume that the process $P_u(s)$ is controlled by a PID controller with set-point weighting i.e., that the control signal is given by the transfer function

$$U(s) = k_p\left(bR(s) - Y(s)\right) + \frac{k_i}{s}\left(R(s) - Y(s)\right) + k_d s D_f(s)\left(cR(s) - Y(s)\right).$$

where s denotes the Laplace operator and k_p, k_i and k_d are the proportional, integral, and derivative gain, respectively. The derivative term is filter by a filter $D_f(s)$. The filter is assumed to be fixed and responsibly chosen to give desired high-frequency gain. The disturbance dynamics $P_d(s)$ is assumed to be asymptotically stable. The set-point weighted PID controller equivalently can be represented by the block dia-

gram depicted in Figure 1 where the feedback controller

$$C(s) = k_p + \frac{k_i}{s} + k_d s D_f(s) \tag{1}$$

is a PID controller without set-point weighting and

$$F_r(s) = k_p(b-1) + k_d(c-1)sD_f(s) \tag{2}$$

is a reference feedforward filter. We define the disturbance feedforward controller as

$$F_d(s) = \tilde{k}_p + \tilde{k}_d s D_f(s). \tag{3}$$

The feedforward controllers are PD-controllers. Their simple structure is motivated by the simplicity of the feedback controller. For a process where PID-control is chosen as the feedback controller it is reasonable to use feedforward controllers of the same complexity. The use of an integrating part in the feedforward controllers would cause the magnitude of the control signal to grow infinitely large for steps in reference or disturbances. PD controllers can be implemented in any modern programmable logic controller (PLC) and distributed control system (DCS) and can be considered the standard feedforward controller in process industry.

The error, $e(t) = r(t) - y(t)$ and the control signal are given by Laplace transformations as

$$E(s) = G_{er}(s)R(s) + G_{ed}(s)D(s) \tag{4}$$

$$U(s) = G_{ur}(s)R(s) + G_{ud}(s)D(s) \tag{5}$$

where

$$G_{er} = S(1 - P_u F_r), \quad G_{ed} = -S(P_d - P_u F_d) \tag{6a}$$

$$G_{ur} = S(C + F_r) \qquad G_{ud} = -S(CP_d + F_d) \tag{6b}$$

where the sensitivity function and the complementary sensitivity function are denoted by $S = 1/(1 + P_u C)$ and $T = P_u CS$, respectively. The signals above are functions of the Laplace operator that has been omitted in order to avoid cluttering.

A disadvantage with this structure is that the feedforward controller from the measurable disturbance F_d and the feedback controller are coupled. If the feedback controller is retuned then the feedforward controller also has to be appropriately retuned. There exist controller structures that decouple the two controllers see e.g., [Brosilow and Joseph, 2002] and the work building upon that idea [Hast and Hägglund, 2014; Hast and Hägglund, 2012].

We assume that the feedback controller, C, has been designed so that the resulting feedback loop is asymptotically stable and has good robustness.

From (6) we see that perfect tracking and disturbance attenuation are obtained if the feedforward controllers are chosen as

$$F_r = P_u^{-1}, \quad F_d = P_u^{-1} P_d \tag{7}$$

These controllers need not be realizable or stable. If for instance P_u has zeros in the right half-plane, the feedforward controllers are unstable or, if P_u has a time-delay larger than that of P_d the F_d controller would not be realizable. In cases when the inversion of the process is possible the controller could give rise to large control signal and the performance could deteriorate if the process dynamics change.

It follows from (2), (3) and (6) that if the measurable disturbance acts on the process output i.e., $P_d = 1$ the transfer functions from r to e and from d to e are equal, apart from the sign. Hence, if the reference tracking problem is solved and optimal b and c are obtained, the optimal solution for $\tilde{k}_p$ and $\tilde{k}_d$ are readily obtained as

$$\begin{aligned} \tilde{k}_p &= k_p(b-1) \\ \tilde{k}_d &= k_d(c-1). \end{aligned} \tag{9}$$

3. Error Minimization

It is common to measure the performance of control systems by some appropriate norm of the error for some reference and disturbance that the system is likely to be subjected to. The aim of this article is to present the problem of choosing the set-point weights and the disturbance feedforward controller that minimizes the error measured in some norm subject to appropriate constraints that can be solved by solving a convex optimization problem [Boyd and Vandenberghe, 2004]. For simplicity we will illustrate the method using the integrated absolute error, which is common in process control, defined by

$$\text{IAE} = \int_0^\infty |e(t)| \mathrm{d}t \tag{10}$$

as the performance measure. It is also common that there are limitations on the control signal that should be used to track the reference or to attenuate the disturbance. Using the controllers (2) and (3) in (4) and (5), with (6), we see that the error and control signals are affine expressions of the feedforward controller parameters.

REMARK 1
We focus on IAE as the performance measure in this article. However, any convex function could be used to measure the performance. Special cases include the integrated-square-error defined as

$$\text{ISE} = \int_0^\infty e(t)^2 dt, \tag{11}$$

which is equivalent to the minimizing $\|e(t)\|_{\mathscr{L}_2}$ and therefore equivalent to $\|e(t)\|_2$. Furthermore, the maximum-absolute-error

$$e_{\max} = \max_t |e(t)| \tag{12}$$

is a special case of the $\mathscr{L}_p$-norm for $p = \infty$. □

The error can be computed as functions of the parameters by inverse Laplace transformation of (4) and be expressed as

$$e(t) = \mathscr{L}^{-1}(E(s)) = e_r(t) + e_d(t) \tag{13}$$

where the influence of the reference feedforward is given by

$$e_r(t) = \mathscr{L}^{-1}\left(S\left(1 + P_u\left(k_p + k_d s D_f\right)\right)R - SPR\begin{bmatrix}k_p & k_d s D_f\end{bmatrix}\mathbf{x_r}\right) = e_r^0(t) + Z_r(t)\mathbf{x_r} \tag{14}$$

where $\mathbf{x_r} = \begin{bmatrix}b & c\end{bmatrix}^T$ and the influence of the disturbance is given by

$$e_d(t) = \mathscr{L}^{-1}\left(-SD - SP\begin{bmatrix}1 & sD_f\end{bmatrix}\mathbf{x}_d\right) = e_d^0(t) - Z_d(t)\mathbf{x_d}. \tag{15}$$

In a similar way the control signal can be expressed as

$$u(t) = \mathscr{L}^{-1}(U(s)) = u_r(t) + u_d(t) \tag{16}$$

where

$$u_r(t) = \mathscr{L}^{-1}\left(SCR + SR\begin{bmatrix}k_p & k_d s D_f\end{bmatrix}\mathbf{x}_r\right) = u_r^0 + W_r(t)\mathbf{x}_r \tag{17}$$

where $\mathbf{x}_d = \begin{bmatrix}\tilde{k}_p & \tilde{k}_d\end{bmatrix}^T$ and

$$u_d(t) = \mathscr{L}^{-1}\left(-SCD + SD\begin{bmatrix}1 & sD_f\end{bmatrix}x_d\right) = u_d^0 + W_d(t)\mathbf{x}_d \tag{18}$$

where $u^0(t)$ is the control signal if feedforward control is not used.

As can be seen in (14), (15), (17) and (18) all four responses are linear in the controller variables. Splitting the error and control signal into two parts, one dependent only on the reference and one on the disturbance, allows us to formulate one optimization problem that simultaneously and independently designs the two feedforward controllers.

Note that also the derivative of the control signal can be obtained in a similar fashion by multiplying the expression (16) by s and performing the partitioning of the signal into the different parts corresponding to the feedforward parameters.

REMARK 2
Since the transfer functions in Equation (6) are affine in $\mathbf{x}_r$ and $\mathbf{x}_d$ the optimization problems can also be defined in the frequency domain where some suitable norm of (6a) is minimized. Mixes of the time responses and the frequency responses are also possible. □

3.1 Sampling

To obtain a tractable optimization problem, that can be solved by a numerical solver, the cost function and constraints need to be sampled. The sampled problem will be convex so a large number of sampling points, say several thousands, can be used.

To sample the responses we first define a set of N time samples at which the response functions should be evaluated. Let the set of time-samples be

$$\mathbf{t} = \{t_1, t_2, \ldots, t_N\} \tag{19}$$

where the time-samples are monotonically increasing i.e., $t_i < t_{i+1}$ for $i = 1, 2, \ldots, N-1$. The samples should be chosen dense enough so that all relevant information in the time-responses are captured. This implies that the last sample, t_N has to be sufficiently large so that all the signals that are sampled has converged.

By evaluating the error functions (14) and (15) at the samples in $\mathbf{t}$ we obtain their sampled counterparts

$$\mathbf{e}_r = \mathbf{e}_r^0 + \mathbf{Z}_r \mathbf{x}_r \tag{20a}$$
$$\mathbf{e}_d = \mathbf{e}_d^0 - \mathbf{Z}_d \mathbf{x}_d \tag{20b}$$

where $\mathbf{e}_r$, $\mathbf{e}_d$, $\mathbf{e}_r^0$ and $\mathbf{e}_d^0$ are vectors of length N and, $\mathbf{Z}_r$ and $\mathbf{Z}_r$ are $N \times m_r$ matrices.

By evaluating the control signals (17) and (18) at the same time-instances we obtain the sampled control signals

$$\mathbf{u}_r = \mathbf{u}_r^0 + \mathbf{W}_r \mathbf{x}_r \tag{21a}$$
$$\mathbf{u}_d = \mathbf{u}_d^0 + \mathbf{W}_d \mathbf{x}_d \tag{21b}$$

Using the sampled versions of the error, minimization of IAE is equivalent to minimizing the 1-norm of $\mathbf{e_r} + \mathbf{e_d}$. Furthermore, the response of set-point changes and disturbances are independent so that the minimization can be performed on each term individually.

REMARK 3
The presented method for designing feedforward controllers is dependent on the fixed denominator in the feedforward controllers. Fortunately, the controller poles are fixed when set-point weighting is used. The method is readily extendable to any feedforward controller that has fixed denominator. By specifying an asymptotically stable prefilter with relative degree n and allowing the numerator polynomial coefficients to be optimization parameters the optimization problem is convex and can easily be solved. □

4. Feedforward Design

Feedforward is usually introduced to improve the system's response to an exogenous input signal. The applicability of the two feedforward controllers in Figure 1 is highly application dependent. If the system operates at a fixed set-point at all times there is no need for F_r and if there are no measurable disturbances, F_d cannot be used. As mentioned in Sec. 2, inverting the system's dynamics is usually not possible in the general case. Utilizing the set-point weighted PID controller and PD feedforward controller from a measurable disturbance where the poles are fixed we can formulate a great variety of convex optimization problems that give sound feedforward controllers and that significantly improve the overall system behavior. We only optimize the numerator, or zeros, of the feedforward controller and there is therefore no guarantees that the feedforward controller is globally optimal.

4.1 Convex constraints

Designing a controller by just minimizing an appropriate norm of the error will not necessarily produce a "good" controller. Quantifying what means by a good controller can be hard and is highly dependent on the type of application. In this section we will present a number of convex constraints that can help the designer to obtain a better controller. A convex constraint is one on the form

$$f(x) \leq 0 \tag{22}$$

where the function $f : \mathbf{R}^n \to \mathbf{R}$ has a convex domain and for all x, y in that domain satisfies

$$f(\theta x + (1-\theta)y) \leq \theta f(x) + (1-\theta) f(y) \tag{23}$$

for $0 \leq \theta \leq 1$.

The functions (20), (21) are affine in the parameters which implies that constraints on the form

$$l(x) \leq f(x) \leq q(x) \tag{24}$$

are convex if $l(x)$ is convex and $q(x)$ is concave i.e., $-q(x)$ is convex and $f(x)$ is interpreted as one of the five functions.

This will prove useful since we can constrain the time-responses by specifying upper and lower bounds on them for each time sample. This is done using the constraint

$$\mathbf{e}_{\min} \leq \mathbf{e} \leq \mathbf{e}_{\max} \tag{25}$$

where $\mathbf{e}_{\min}$ and $\mathbf{e}_{\max}$ are vectors of the same size as $\mathbf{e}$ and where they, element-wise, satisfy $\mathbf{e}_{\min} \leq \mathbf{e}_{\max}$. This means that we can constrain the error, for a given input signal, to lie within an envelope.

As a special, and useful, case of (24) we could specify that the magnitude of the error or control signals, arising from a disturbance, should be no larger than some

value α, by including the constraints

$$|\mathbf{v}| \leq \alpha \tag{26}$$

where $\mathbf{v}$ is interpreted as either $\mathbf{e}_r$, $\mathbf{e}_d$, $\mathbf{u}_r$ or $\mathbf{u}_d$. This is of course a convex constraint since the absolute value of an affine expression is a convex function.

Note that the norms also can be used as constraints i.e.,

$$\|v(t)\| \leq \alpha \tag{27}$$

where α is a positive scalar.

Control signal activity Good tracking and disturbance attenuation often comes at the expensive of large control signals. In most applications however large control signals is undesirable. Therefore it can be important to include constraints or terms in the cost functions that penalize the use of large control signals in the optimization.

Saturation is easily handled within the proposed framework. By calculating the control signals components using the method in Sec. 3.1 we can constrain the control signal by introducing

$$u_{\min} \leq \mathbf{u} \leq u_{\max} \tag{28}$$

where $\mathbf{u}$ should be interpreted as either $\mathbf{u}_\mathrm{r}$ or $\mathbf{u}_\mathrm{d}$ and $u_{\min}$ and $u_{\max}$ are the lower and upper bound on the control signal respectively. Of course, the $u_{\min}$ and $u_{\max}$ can be time-dependent in the same way that was done in the constraint (25).

In a similar way as the control signal was calculated, the derivative of the control signal can be calculated as

$$\dot{u}(t) = \mathscr{L}^{-1}\big(sU(s)\big) \tag{29}$$

provided that U is strictly proper. This derivative can then be sampled using the procedure in Sec. 3.1. Once we have obtained a sampled version of the control signal derivative it can be used to impose rate-constraints. Denoting the sampled derivative by $\dot{\mathbf{u}}$ the rate constraint can be formulated as

$$\alpha \leq \dot{\mathbf{u}} \leq \beta. \tag{30}$$

Another way to limit the control signal activity is to measure the control signal activity by one appropriate norm $\|\dot{u}(t)\|$.

EXAMPLE 1—IAE MINIMIZATION

We illustrate the how the method can be used to design feedforward controllers from both set-point and measurable disturbance.

Consider the control structure in Figure 1 and let the processes be defined as

$$P_u(s) = \frac{\mathrm{e}^{-s}}{(0.5s+1)^4}, \quad P_d(s) = \frac{\mathrm{e}^{-0.3s}}{0.3s+1} \tag{31}$$

To ensure roll-off and thus attenuating measurement noise entering the system the PID controller $C(s) = k_p + k_i/s + k_d s$ is filtered with the low-pass filter $G_f(s) = 1/(0.1s+1)^2$. The feedback controller is tuned using the method presented in [Hast, K. J. Åström, et al., 2013] which minimizes the integrated error subject to robustness constraints on the maximum values of the sensitivity and complementary sensitivity function should be less than 1.4. The filter is taken into account when the feedback controller is designed, i.e., the controller is tuned for the process $\tilde{P}_u(s) = P_u(s)G_f(s)$. The resulting PID controller is

$$C(s) = 0.46 + \frac{0.39}{s} + 0.51s. \tag{32}$$

The feedforward controllers are PD-controllers equivalent to the set-point weighting in accordance with (2) and (3) with the same filters as the feedback controller. The feedforward controllers are thus

$$F_r(s) = \frac{k_p(b-1) + k_d s(c-1)}{(0.1s+1)^2}, \quad F_d(s) = \frac{\tilde{k}_p + \tilde{k}_d s}{(0.1s+1)^2}. \tag{33}$$

We let the external inputs r and d be unit-steps and want to minimize the IAE error with the constraints that the magnitude of the control signals that arises are less than 5. Furthermore we impose constraints on the overshoot of the responses. For the set-point case we allow less than 5 per cent overshoot and for the disturbance case we allow the overshoot to be the same as it were when no feedforward was used. By using $N = 2000$ uniform samples in the time interval $t = [0, 20]$, and applying them to the equations (14), (15), (17) and (18) we obtain the vectors and matrices in (20) and (21). The optimization problem can be formulated as

$$\begin{aligned} \underset{\mathbf{x}_r, \mathbf{x}_d}{\text{minimize}} \quad & \|\mathbf{e}_r^0 + \mathbf{Z}_r\mathbf{x}_r\|_1 + \|\mathbf{e}_d^0 - \mathbf{Z}_d\mathbf{x}_d\|_1 \\ \text{subject to} \quad & \mathbf{e}_r^0 + \mathbf{Z}_r\mathbf{x}_r \geq -0.05 \\ & \mathbf{e}_d^0 - \mathbf{Z}_d\mathbf{x}_d \leq 0.138 \\ & |\mathbf{u}_r^0 + \mathbf{W}_r\mathbf{x}_r| \leq 5 \\ & |\mathbf{u}_d^0 + \mathbf{W}_d\mathbf{x}_d| \leq 5 \end{aligned}$$

The optimization problem was solved using CVX in MATLAB and took less than 2 seconds to solve on an ordinary desktop computer. The optimal solution is

$$b^* = 1.35, \quad c^* = 2.56, \quad \tilde{k}_p^* = 0.16, \quad \tilde{k}_d^* = 0.80. \tag{34}$$

□

Responses to unit-step changes in both reference and disturbance can be seen in Figure 2. The performance is increased by using feedforward but at the cost of a more aggressive control signal. Using the optimal feedforward controllers the IAE is decreased by 32 and 49 per cent for a step change in the set-point and the disturbance, respectively.

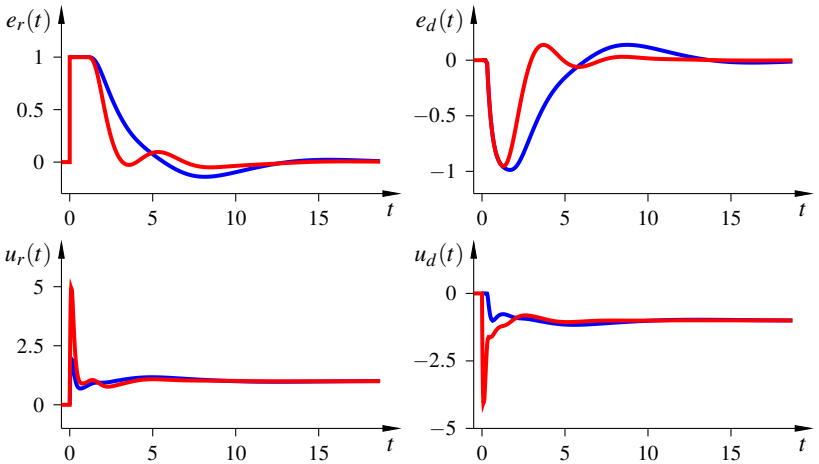

Figure 2. Responses for the control system with (red) and without (blue) feedforward. The upper plots show the errors when a unit-step is applied at the reference (left) and at the measurable disturbance (right). The lower plots show the corresponding control signals.

5. Tuning Rules for Set-Point Weighting

When deriving tuning rules two approaches are common; analytic based on simple, often first-order processes with time-delay (FOTD), models, or rules based on conclusions drawn for a large set of processes. We will adapt to the latter and use the same batch of processes that where used to derive the AMIGO PI and PID tuning rules [K. Åström and Hägglund, 2004]. The batch consists of nine process types, see (36), with various parameters, in total 134 processes. See [K. J. Åström and Hägglund, 2006], p.227 for the complete description of the processes with all parameter values. The batch consists of processes ranging form order 1 to 8, including integrating processes and processes with nonminimum phase behavior. All processes are characterized by having *essentially* monotone step-response i.e.,

$$\frac{\int_0^\infty g(t)dt}{\int_0^\infty |g(t)|dt} > 0.8 \tag{35}$$

where $g(t)$ is the impulse response of the process. We let the feedback parameters of the controller be determined by the AMIGO-tuning rules. These were derived so that the controller have good robustness and good disturbance rejection. The set-point response is given by $\mathbf{e}_r$ (20).

$$
\begin{aligned}
&P_1 : \frac{e^{-s}}{1+sT}, && P_4 : \frac{1}{(s+1)^n}, && P_7 : \frac{Te^{-sL_1}}{(1+sT)(1+sT_1)},\\
&P_2 : \frac{e^{-s}}{(1+sT)^2}, && P_5 : \prod_{i=0}^{3}\frac{1}{(1+\alpha^i s)}, && P_8 : \frac{1-\alpha s}{(s+1)^3},\\
&P_3 : \frac{1}{(s+1)(1+sT)^2}, && P_6 : \frac{e^{-sL_1}}{s(1+sT_1)}, && P_9 : \frac{1}{(s+1)((sT)^2+1.4sT+1)}.
\end{aligned}
\tag{36}
$$

5.1 IAE-minimization for PI controllers.

We first consider the problem of choosing the set-point parameter b so that it minimizes IAE subject to a unit step in the reference signal for a PI controller, i.e., $k_d = 0$. The error is given by

$$E(s) = S(s)(1 - P(s)F_r(s))\frac{1}{s} \tag{37}$$

with $C(s)$ given by Equation 1 that is tuned using the AMIGO-rules for each process in the batch. The error function is inversely Laplace transformed and sampled with $N = 1000$ sample points in appropriate time intervals. Minimization of IAE is then equivalent to

$$\underset{b}{\text{minimize}} \quad \|\mathbf{e_r}\|_1 \tag{38}$$

where $\mathbf{e}_r$ is defined as in Equation 20a. The optimal b parameters are plotted against the product of the stationary gain of the processes and the proportional gain in the feedback controller in Figure 3, where the integrating processes are excluded since their stationary gain are infinitely large. As can be seen from the figure the optimal set-point weight b^* decreases for large $k_pP(0)$. By fitting a function on the form

$$b^* = \frac{1}{\alpha k_p P(0)} + \beta \tag{39}$$

we obtain $\alpha = 1.90$ and $\beta = 0.7$. For the integrating process we obtain b^* in the range of $[0.732, 0.7532]$. We see that the tuning rule

$$b^*_{PI} = \frac{1}{2k_pP(0)} + 3/4 \tag{40}$$

capture also the integrating process. The left plot in Figure 4 shows the increase in performance when using set-point weighting designed using the tuning rule (40) to the IAE obtained when set-point weighting is not used, i.e., $b = 1$. We see that for 32 of the 134 processes in the batch the performance increase is less than 10 percent. These correspond to the processes for which b^* is close to one, i.e., the PI-controller without set-point weighting is optimal.

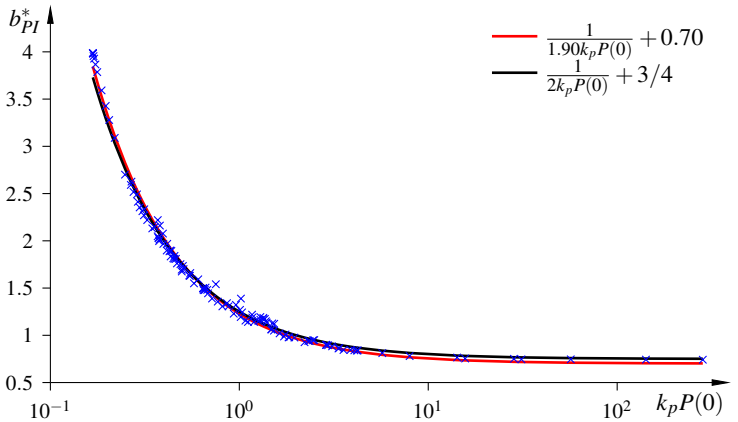

Figure 3. Optimal b parameter for PI controllers as function of the product of the static gain of the process and the controller proportional gain. The red curve is the one that best fits the obtained optimal solutions and the black curve is the simple tuning rule.

The control signal at the time instance where the reference changes can be shown to be

$$\lim_{t\to 0} u(t) = k_p b. \tag{41}$$

Thus, using the tuning rule the initial control signal is $u(0) = \frac{1}{2P(0)} + 0.75k_p$. The proportional gain k_p is, in most tuning rules for PID controllers, inversely proportional to the static gain of the process. For processes with small static gain the initial control signal becomes large. A sound course of action in those cases would be to include the control signal in the optimization problem and solve it for that specific process. In order to verify that the control signals is not excessively magnified when using the tuning rule consider the control signal ratio

$$\Delta u_{\max} = \frac{\max\limits_t |u(t)|}{\max\limits_t |u_0(t)|} \tag{42}$$

where $u_0(t)$ is the control signal without set-point weighting i.e., $b = 1$. The right plot in Figure 4 shows the control signal ratio for all processes in the AMIGO batch. As can be seen in the figure, the largest magnitude of the control signal does not change more than 25 per cent when the tuning method is used. For most processes, the largest magnitude of the control signal increases.

5.2 IAE-minimization for PID controllers

When using set-point weighting in PID controllers both the b and c parameters need to be taken into account. Alas, the choice of the necessary derivative filter will

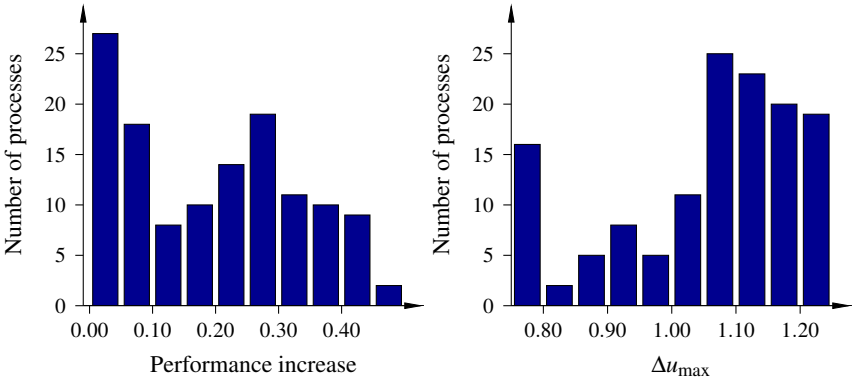

Figure 4. *Left:* Performance increase when using the tuning rule for PI controllers for the proportional set-point weight. *Right:* Change in largest control signal when using the tuning rule for PI controllers compared to control without set-point weighting.

influence the choice of both parameters. However, optimal set-point weights can be obtained by solving the optimization problem

$$\underset{b,c}{\text{minimize}} \quad \|\mathbf{e_r}\|_1 \tag{43}$$

where $\mathbf{e}_r$ is defined as in Equation 20a using a PID controller.

For many plants in process industry the set-point is changed in steps and the therefore the c parameter is set to zero in order to avoid unnecessarily large transients in the control signal. Using the tuning rule in (40) for PID control with $c = 0$ unfortunately does not give satisfactory results. Consider therefor a same optimization problem as the one stated in (38) but with $\mathbf{e}_r$ calculated for a PID controller and $c = 0$. For numerical reasons a fast first-order filter $D_f(s) = 1/(10^{-5}s+1)$ was used for the derivative part.

Figure 5 shows the optimal b parameter plotted against the product of the stationary gain of the process and the proportional gain of the feedback controller. The red curve shows the best fit for a function on the form (39) to the optimal b parameters and it is given by

$$b^*_{\text{PID}} = \frac{1}{2k_pP(0)} + 0.55 \tag{44}$$

The performance increase yielded when using the the tuning rule can be seen in the right plot of Figure 6. Using the tuning rule the performance for all processes increase but for 43 of them the increase in performance is less than 5 per cent. The peak values of the control signal decreases for most processes as can be seen in the right plot of Figure 6, which shows the change in control signal when using the tuning rule.

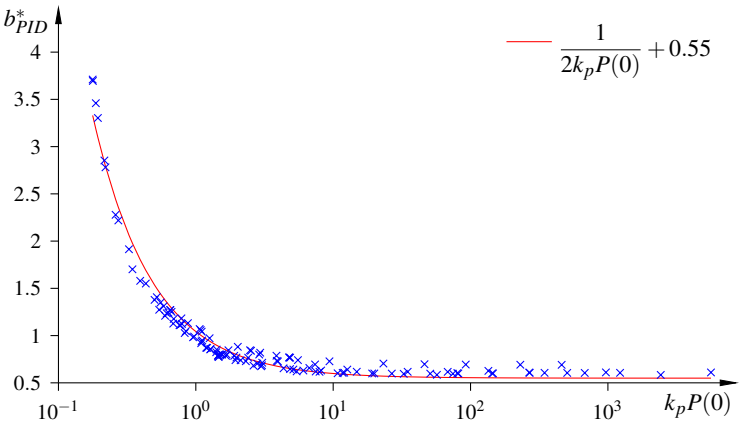

Figure 5. Optimal b parameter for PID controllers,with $c = 0$, as function of the product of the static gain of the process and the controller proportional gain. The red curve is the one that best fits the obtained optimal solutions.

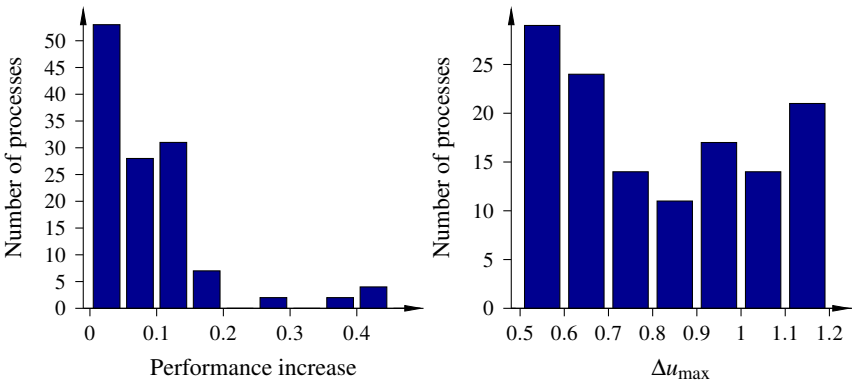

Figure 6. *Left:* Performance increase when using the tuning rule for PID controllers, with $c = 0$, for the proportional set-point weight. *Right:* Change in largest control signal when using the tuning rule for PID controllers compared to set-point weights $b = 1$ and $c = 0$.

5.3 Summary of the tuning rules

The tuning rules for how to choose the proportional set-point weight for PI and PID controllers with $c = 0$ can be summarized as

$$b^*_{\rm PI} = \frac{1}{2k_pP(0)} + 0.75, \quad b^*_{\rm PID} = \frac{1}{2k_pP(0)} + 0.55. \tag{45}$$

For the batch of processes that they were derived from they increase the performance with up to 45 per cent. The largest magnitude of the control signal is at most increased by 25 per cent.

The tuning rules can also be used to tune static feedforward controllers from a measurable disturbance if the disturbance dynamics $P_d = 1$. It can be seen from the equations in (6a), that the errors arising in this case only differs by the sign of the transfer functions. From this fact and Equation (9) we see that static controllers that minimizes IAE are

$$\tilde{k}^{\rm PI}_p = \frac{1}{2P(0)} - \frac{k_p}{4}, \quad \tilde{k}^{\rm PID}_p = \frac{1}{2P(0)} - 0.45k_p \tag{46}$$

for PI and PID feedback controllers, respectively.

6. Conclusions

In this paper we presented a method for finding the set-point weights in a PID controller by using convex optimization techniques. The problem of finding the proportional set-point weight that minimizes the integrated absolute error was solved for a batch of 134 processes. An approximate expression for how the optimal set-point weight depend on the stationary gain of the process and the proportional gain in the PI(D) controller was found and used as tuning rule. The tuning rules were evaluated for all processes in the batch and the resulting performance and control signal activity was discussed.

Acknowledgement

This work was supported by the Swedish Research Council through the LCCC Linnaeus Center, the ELLIIT Excellence Center and by the Swedish Foundation for Strategic Research through PICLU.

References

Andersen, M. S., J. Dahl, and L. Vandenberghe (2013). “CVXOPT: a Python package for convex optimization, version 1.1.6”. *Available at cvxopt. org*.

Andersen, M., J. Dahl, Z. Liu, and L. Vandenberghe (2011). "Interior-point methods for large-scale cone programming". *Optimization for machine learning*, pp. 55–83.

Åström, K. J. and T. Hägglund (2006). *Advanced PID Control*. Instrumentation, Systems, and Automation Society.

Åström, K. and T. Hägglund (2004). "Revisiting the ziegler–nichols step response method for PID control". *Journal of Process Control* **14**:6, pp. 635–650.

Bianchi, F. D., R. J. Mantz, and C. F. Christiansen (2008). "Multivariable PID control with set-point weighting via BMI optimisation". *Automatica* **44**:2, pp. 472–478.

Boyd, S. and C. Barratt (1991). *Linear Controller Design — Limits of Performance*. Prentice-Hall.

Boyd, S. and L. Vandenberghe (2004). *Convex Optimization*. Cambridge University Press.

Brosilow, C. and B. Joseph (2002). *Techniques of Model-Based Control*. Prentice Hall PTR, pp. 221–239.

Carrasco, D. S. and G. C. Goodwin (2011). "Feedforward model predictive control". *Annual Reviews in Control* **35**:2, pp. 199–206.

Giusto, A., A. T. Neto, and E. Castelan (1996). "H_∞ and H_2 design techniques for a class of prefilters". *Automatic Control, IEEE Transactions on* **41**:6, pp. 865–870.

Giusto, A. and F. Paganini (1999). "Robust synthesis of feedforward compensators". *Automatic Control, IEEE Transactions on* **44**:8, pp. 1578–1582.

Graichen, K., V. Hagenmeyer, and M. Zeitz (2005). "A new approach to inversion-based feedforward control design for nonlinear systems". *Automatica* **41**:12, pp. 2033–2041.

Grant, M. and S. Boyd (2008). "Graph implementations for nonsmooth convex programs". In: Blondel, V. et al. (Eds.). *Recent Advances in Learning and Control*. Lecture Notes in Control and Information Sciences. Springer-Verlag Limited, pp. 95–110.

Guzmán, J. L. and T. Hägglund (2011). "Simple tuning rules for feedforward compensators". *Journal of Process Control* **21**:1, pp. 92–102.

Hast, M., K. J. Åström, B. Bernhardsson, and S. Boyd (2013). "PID design by convex-concave optimization". In: *Proceedings European Control Conference*, pp. 4460–4465.

Hast, M. and T. Hägglund (2012). "Design of optimal low-order feedforward controllers". In: *IFAC Conference on Advances in PID Control*. Brescia, Italy.

Hast, M. and T. Hägglund (2014). "Low-order feedforward controllers: optimal performance and practical considerations". *Journal of Process Control* **24**:9, pp. 1462–1471.

Isaksson, A., M. Molander, P. Moden, T. Matsko, and K. Starr (2008). "Low-order feedforward design optimizing the closed-loop reponse". In: *Preprints, Control Systems*. Vanccouver, Canada.

Kose, I. and C. W. Scherer (2007). "Robust feedforward control of uncertain systems using dynamic IQCs". In: *Decision and Control, 2007 46th IEEE Conference on*. IEEE, pp. 2181–2186.

Leva, A. and L. Bascetta (2007). "Designing the feedforward part of 2-d.o.f. industrial controllers for optimal tracking". *Control Engineering Practice* **15**:8, pp. 909–921.

Lofberg, J. (2004). "YALMIP: a toolbox for modeling and optimization in MATLAB". In: *Computer Aided Control Systems Design, 2004 IEEE International Symposium on*. IEEE, pp. 284–289.

Piazzi, A. and A. Visioli (2001a). "Optimal inversion-based control for the set-point regulation of nonminimum-phase uncertain scalar systems". *Automatic Control, IEEE Transactions on* **46**:10, pp. 1654–1659.

Piazzi, A. and A. Visioli (2001b). "Optimal noncausal set-point regulation of scalar systems". *Automatica* **37**:1, pp. 121–127.

Piccagli, S. and A. Visioli (2009). "An optimal feedforward control design for the set-point following of MIMO processes". *Journal of Process Control* **19**:6, pp. 978–984.

Prempain, E. and I. Postlethwaite (2001). "Feedforward control: a full-information approach". *Automatica* **37**:1, pp. 17–28.

Research, C. (2012). *CVX: MATLAB software for disciplined convex programming, version 2.0*. `http://cvxr.com/cvx`.

Scorletti, G. and V. Fromion (2006). "Further results on the design of robust H_∞ feedforward controllers and filters". In: *Decision and Control, 2006 45th IEEE Conference on*. IEEE, pp. 3560–3565.

Vilanova, R. and A. Visioli (2012). *PID Control in the Third Millennium: Lessons Learned and New Approaches*. Springer. Chap. 7, pp. 207–233.

Visioli, A. (2004). "A new design for a PID plus feedforward controller". *Journal of Process Control* **14**:4, pp. 457–463.