Control Engineering Practice 20 (2012) 355–370

Contents lists available at SciVerse ScienceDirect

Control Engineering Practice

journal homepage: www.elsevier.com/locate/conengprac

Robust optimization-based multi-loop PID controller tuning: A new tool and its industrial application

Rainer Dittmar [a,*], Shabroz Gill [b], Harpreet Singh [b], Mark Darby [c]

[a] West Coast University of Applied Sciences, Department of Electrical and Information Engineering, Fritz-Thiedemann-Ring 20, 25746 Heide, Germany
[b] IPCOS BV, Bosscheweg 135B, 5282 Boxtel, The Netherlands
[c] CMiD Solutions, 13106 Dogwood Blossom Trail, Houston, Texas 77065, USA

ARTICLE INFO

Article history:
Received 15 December 2009
Accepted 17 October 2011
Available online 29 November 2011

Keywords:
PID controller
Multiloop control
Tuning characteristics
Decentralized control
Nonlinear optimization
System identification

ABSTRACT

Modern process plants are highly integrated and as a result, decentralized PID control loops are often strongly interactive. The iterative SISO tuning approach currently used in industry is not only time consuming, but does also not achieve optimal performance of the inherently multivariable control system. This paper describes a method and a software tool that allows control engineers/technicians to calculate optimal PID controller settings for multi-loop process systems. It requires the identification of a full dynamic model of the multivariable system, and uses constrained nonlinear optimization techniques to find the controller parameters. The solution is tailored to the specific control system and PID algorithm to be used. The methodology has been successfully applied in many industrial advanced control projects. The tuning results that have been achieved for interacting PID control loops in the stabilizing section of an industrial Gasoline Treatment Unit as well as a Diesel Desulfurization plant are presented.

1. Introduction

One of the most important challenges facing the process industry today is optimizing the operation of complex units, without compromising the safety and integrity of the process equipment. Process complexity has increased significantly over the past two decades due to increased level of heat integration and the use of recycle streams. In addition, the need for increased process flexibility to deal with changing raw materials and alternate energy sources, as well as the need to adapt quickly to fluctuating throughput and quality targets, often means that the process dynamics will vary significantly over time and with operating point. The basic regulatory control layer of process plants almost always consists of a large number of decentralized SISO PID controllers. Although this approach is intrinsically inadequate for multivariable processes, it is a common practice in industry for decades now, and for good reasons: single-loop PID controllers are often effective and easy to implement, and decentralized structures are failure-tolerant. However, due to the situation described above, the interactions between these controllers are becoming more important, and tuning these control loops for good performance and adequate robustness is becoming a more and more challenging task.

A feedback system with n decentralized controllers is shown in Fig. 1. In multivariable notation, the controller $\mathbf{C}(s)$ is a diagonal matrix given by

$$\mathbf{C}(s) = \begin{bmatrix} C_1(s) & \cdots & 0 \\ 0 & \ddots & 0 \\ 0 & \cdots & C_n(s) \end{bmatrix} \tag{1}$$

The process transfer function matrix $\mathbf{G}(s)$ is

$$\mathbf{G}(s) = \begin{bmatrix} G_{11}(s) & \cdots & G_{1n}(s) \\ 0 & \ddots & 0 \\ G_{n1}(s) & \cdots & G_{nn}(s) \end{bmatrix} \tag{2}$$

The vector $\mathbf{y}(t) \in R^n$ denotes the process outputs (controlled variables), $\mathbf{u}(t) \in R^n$ is the vector of manipulated variables, and $\mathbf{r}(t) \in R^n$ the vector of reference signals or setpoints.

The industrial practice of single-loop PID controller tuning is still dominated by manual trial-and-error tuning. If tuning rules are used at all, it is the "classical" ones like Ziegler – Nichols, Cohen – Coon, Chien – Hrones – Reswick or λ-Tuning, which are based on simplified low order process models, and do not explicitely consider stability robustness issues, therefore often being inadequate in modern process units with more complex dynamics. Modern identification techniques are not typically used to develop models for PID tuning. Substantial progress has been achieved in PID controller design during the last decade (see, for example, Aström & Hägglund, 2006; Kristiansson & Lennartson, 2006; Skogestad, 2003), but not yet

* Corresponding author. Tel.: +49 481 8555 325; fax: +49 481 8555 301.
E-mail address: dittmar@fh-westkueste.de (R. Dittmar).

0967-0661/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.conengprac.2011.10.011

adequately adressed in further education and industrial practice. In addition, many tuning rules assume that all PID controller equations work as described in the textbooks, when in fact there is substantial variation between the different vendors. Different PID controller structures result due to the use of either the parallel or the serial form, using the control error or the PV by the Proportional (P) and Derivative (D) terms, alternative implementations of the derivative filter and others. Tuning SISO PID controllers in a multivariable environment is usually done in a time-consuming sequential and iterative way, starting with the most important loops, and heuristic detuning in case the interactions are significant. The results in terms of performance and robustness strongly depend on the particular application and on the experience of the personnel involved.

For a long time, vendors of automation systems such as Distributed Control Systems (DCS) and Programmable Logic Controllers (PLC) have been offering PID self-tuning functionality (tuning on demand). Unfortunately, they have found only limited application. This is also true for model based PID controller tuning software provided by the same or third-party vendors. Moreover, in most cases these tools are restricted to single-loop tuning applications, and do not support multi-loop tuning (Li, Ang & Chong, 2006). Examples are RaPID (Espinosa Oviedo, Boelen & van Overschee, 2006; Van Overschee & de Moor, 2000) and TaiJi PID (Zhu, 2004). In TiaJi PID, plant models are identified from open-loop or closed-loop tests using high-order ARX models and reduced to low-order transfer function models. PID controller parameters are calculated using IMC-based tuning rules. In the current release, tuning is restricted to single-loop applications. However, the user can simulate the multivariable system controlled by independently tuned single-loop PID controllers. In RaPID, the process model is identified using a combination of prediction error methods and subspace identification. Controller settings are calculated based on optimization with robustness and noise amplification constraints.

The design of interacting PID controllers in a multivariable environment is not a new topic in the process control literature. At least three research directions can be identified: (1) reduction of controller interactions by proper pairing of the manipulated and control variables, (2) design of decoupling networks, and (3) consideration of MIMO interactions in decentralized controller tuning. This paper is a contribution to the solution of the third problem. In the past, many design methods have been developed for multi-loop tuning. They can be classified into

1. detuning methods;
2. sequential loop closing methods;
3. independent design methods;
4. relay-feedback auto-tuning methods; and
5. optimization methods.

In detuning methods, each controller is first designed based on the corresponding diagonal element of the process transfer function matrix while ignoring the interactions from other loops. The controllers are then detuned to take the interactions into account. The price to be paid for the reduced interaction is more sluggish behavior of the individual PID loops. The most popular detuning method called "biggest log modulus tuning" (BLT) was developed by Luyben (1986). In its original version, individual PI loops are first independently tuned by the Ziegler – Nichols (ZN) rules. Then, for multiloop tuning, all ZN controller gains are divided by a common detuning factor $F>1$, and the ZN reset times are multiplied by the same factor. This detuning factor can be calculated based on the following considerations. The characteristic equation of the multivariable closed-loop system is

$$\det[\mathbf{I}+\mathbf{G}(j\omega)\mathbf{C}(j\omega)]=0 \qquad (3)$$

The scalar function

$$W(j\omega)=-1+\det[\mathbf{I}+\mathbf{G}(j\omega)\mathbf{C}(j\omega)] \qquad (4)$$

can be plotted in the complex plane as a function of frequency ω. The closer $W(j\omega)$ approaches the critical point $(-1,0)$, the closer the multivariable system is to closed-loop instability. The expression $W/(1+W)$ will be similar to the closed-loop servo transfer function $CG/(1+CG)$ in the SISO case. Luyben has shown that the multivariable system gives reasonable responses for setpoint changes and load disturbances, if F is selected to make the so-called "biggest log modulus" in the frequency range

$$L_{max}=\max_{\omega}\ L(j\omega)=\max_{\omega}\left\{20\log\left|\frac{W(j\omega)}{1+W(j\omega)}\right|\right\} \qquad (5)$$

equal to $2n$ (n being the number of loops to be tuned). This value for F can be found using an iterative procedure. This method has later been extended to PID controllers, and also to compensate for possible asymmetries in multi-loop interactions (Lee & Edgar, 2006; Monica, Yu & Luyben, 1988). Its application requires a dynamic MIMO process model. In Chien, Huang and Yang (1999), the detuning factor F is set to $F=1/RGA(\lambda_{ii})$, e.g. to the diagonal elements of the relative gain array, for systems with $0<RGA(\lambda_{ii})<1$. For loops with $RGA(\lambda_{ii})>1$, the controller parameters remain unchanged. If $RGA(\lambda_{ii})<0$, the recommendation is to switch the MV – CV pairings. In contrast to Luyben's method, dynamic models are required for the diagonal elements of the process $G_{ii}(s)$ only, while static gains for the off-diagonal elements $G_{ij,i\neq j}(0)$ are sufficient. In Xiong, Cai, He and He (2006), not only the static gains, but also the ultimate frequencies $\omega_{u,ij}$ are used to define the detuning factors.

In the sequential loop closing method (Hovd & Skogestad, 1994), the loops are closed one after the other, usually starting with the fastest loop. The dynamic interaction of this loop is considered when closing the next loop, and so on. The disadvantage of this approach is that the overall result depends on the order of loop closing and on the method used for individual controller design, and that iterations may be necessary.

In independent design, loop interactions, robust performance, and stability are considered first, and each controller is then designed independently of each other. Hovd and Skogestad (1993) make use of the μ-interaction measure and the structured singular value to develop bounds for the design of the individual loops. Chen and Seborg (2003) used multivariable Nyquist stability analysis to develop stability regions for SISO PI controllers, and proposed a tuning method, which guarantees closed-loop stability of the decentralized MIMO control system. Huang, Jeng, Chiang and Pan (2003) developed a method for multi-loop PI/PID controller tuning based on the effective open-loop transmission from u_i to y_i and phase/gain margin specifications. Their method assumes that the effective open-loop processes can be approximated by FOPDT or SOPDT models and is practically restricted to TITO systems.

If no analytical process model exists, tuning parameters may be calculated based on the multivariable generalization of relay-feedback autotuning methods, which are described, for example, in Campestrini, Filho and Bazanella (2009), Halevi, Palmor and Efrati (1997), and Yu (2007). Since the execution of sequential or simultaneous relay-feedback experiments under industrial conditions is difficult due to noisy signals, drift, and long duration in case of larger time constants, this approach has found limited application.

Optimization methods have also been used to address the decentralized control problem. Sourlas and Manousiouthakis (1995) developed a benchmark for the best achievable decentralized performance. Trierweiler, Müller and Engell (2000) and Pegel and Engell (2001) describe a generalization of the direct synthesis method to multivariable systems. First, considering the