Auto-Tuning of Multivariable PID Controllers Using Iterative Feedback Tuning

Auto-Tuning von PID-Mehrgrößenreglern mit Hilfe von Iterative Feedback Tuning

Steffen Sommer, Achim Kienle, Otto-von-Guericke-Universität Magdeburg

Herrn Prof. Dr.-Ing. habil. Ulrich Korn zum 70. Geburtstag gewidmet

Summary A new auto-tuning approach for centralized MIMO PID controllers and unknown process models is presented. Classical multivariable PID controller tuning methods are combined with iterative feedback tuning (IFT) to develop an auto-tuning method with a bounded number of tuning parameters. The method is demonstrated and validated with a steam generator example. Very good simulation results are obtained. ▸▸▸ **Zusammenfassung** Im vorliegenden Beitrag wird eine neue Auto-Tuning-Methode für zentrale PID-Mehrgrößenregler und unbekannte Prozessmodelle vorgestellt. Klassische Tuning-Methoden für zentrale PID-Mehrgrößenregler werden mit Hilfe des Iterative Feedback Tuning (IFT) auf eine Auto-Tuning-Methode mit einer begrenzten Anzahl von zu ermittelnden Reglerparametern erweitert. Die vorgeschlagene Methode wird in einer Simulationsstudie demonstriert und validiert. Die Ergebnisse sind sehr vielversprechend.

Keywords Auto-tuning, MIMO, PID, iterative feedback tuning, IFT ▸▸▸ **Schlagwörter** Auto-tuning, MIMO, PID, iterative feedback tuning, IFT

1 Introduction

A centralized (fully cross coupled) multivariable controller is necessary for systems with significant interactions where the control objective cannot be achieved by decentralized control.

Different classical model-free tuning methods for centralized multivariable PID controllers are available. An Overview of such tuning methods can be found in [13; 18]. Multivariable PI controllers are discussed in the text books [14; 15].

But the mentioned classical tuning approaches were not developed as auto-tuning methods. The controller tuning is based only on plant experiments. It is also an automated design procedure which can be implemented in process control systems. Auto-tuning methods are extremely popular in industry. Most published work on PID auto-tuning is restricted to SISO systems.

Typical auto-tuning methods for SISO processes include iterative feedback tuning (IFT) and relay feedback (RF) tuning. Both IFT and RF tuning are closed loop techniques. That is, it is not required to open the feedback loop for the tuning experiments.

Relay feedback tuning was introduced in [1]. A relay is implemented in the control loop as a controller to generate sustained oscillations. The corresponding ultimate gain and ultimate period are determined automatically and used for the controller parameterization with empirical rules. An overview of this technique is given in [2; 10; 21]. The RF method can be extended to the multivariable case for auto-tuning of centralized MIMO PID controllers [20].

The IFT method was originally suggested in [8]. Overviews are presented in [6; 7]. The advantage of the IFT technique is that it is an optimization based

DOI 10.1524/auto.2012.0969

© Oldenbourg Wissenschaftsverlag

Bereitgestellt von | Max-Planck-Gesellschaft - WIB6417
Angemeldet | 193.175.53.21
Heruntergeladen am | 25.03.14 15:44

approach, where no empirical rules are needed. The controller parameters are computed by minimizing an input and error based quadratic cost function, which is done by means of closed loop experiments. IFT can also be extended to the multivariable case to design multivariable controllers [5].

One problem occurs when dealing with auto-tuning of centralized MIMO PID controllers. The number of parameters can be very large. In this contribution, a new PID auto-tuning approach based on IFT with a fixed and reduced number of tuning parameters, independently from the process dimensions, is presented.

The outline of the article is as follows. Section 2 provides background on classical tuning methods for multivariable PID controllers. In Sect. 3, a brief introduction to IFT design for MIMO processes is given. Section 4 describes how to apply the IFT method for the PID controller tuning. The proposed method is validated and demonstrated in Sect. 5. Finally, a conclusion is given.

2 Multivariable PID Controller Tuning Basics

To develop an auto-tuning approach, classical multivariable PID controller tuning is considered. It is assumed that the process is stable and a square system (number of outputs r is equal to the number of inputs m). The state-space representation is given by:

$$\dot{x}(t) = Ax(t) + \begin{pmatrix} B_u & B_d \end{pmatrix} \begin{pmatrix} u(t) \\ d(t) \end{pmatrix} \tag{1}$$

$$y(t) = Cx(t) \tag{2}$$

with the state vector $x(t)$, the manipulated input vector $u(t)$, the output vector $y(t)$, and the vector of disturbance inputs $d(t)$. A, B_u, B_d, and C are appropriate state-space matrices. The process can also be described by the transfer function matrix $G(s)$:

$$Y(s) = G(s)\begin{pmatrix} U(s) \\ D(s) \end{pmatrix} = \begin{pmatrix} G_u(s) & G_d(s) \end{pmatrix} \begin{pmatrix} U(s) \\ D(s) \end{pmatrix} \tag{3}$$

with:

$$G_u(s) = C\big(sI - A\big)^{-1} B_u\,, \tag{4}$$

$$G_d(s) = C\big(sI - A\big)^{-1} B_d\,. \tag{5}$$

The control law of the PID controller in the time domain is given as follows:

$$u(t) = K_P\, e(t) + K_I \int_0^t e(\tau)\, d\tau + K_D \frac{de(t)}{dt} \tag{6}$$

with the control error

$$e(t) = r(t) - y(t)\,, \tag{7}$$

where $r(t)$ is the vector of reference inputs. K_P, K_I, K_D are controller parameter matrices (matrix gains) regarding the proportional, integral, and derivative action. In the Laplace-domain, the control law with the controller transfer function $C_{\mathrm{PID}}(s)$ and the control error is given by:

$$U(s) = \left(K_P + K_I \frac{1}{s} + K_D s\right) E(s) = C_{\mathrm{PID}}(s) E(s)\,, \tag{8}$$

$$E(s) = R(s) - Y(s)\,. \tag{9}$$

Consider the basic approach for the matrix gains defined by:

$$K_I = \alpha \tilde{K}_I\,, \tag{10}$$

$$K_P = \beta \tilde{K}_P\,, \tag{11}$$

$$K_D = \delta \tilde{K}_D\,. \tag{12}$$

The matrix gains are divided into a scalar and a matrix. The scalars α, β, and δ are called fine tuning parameters. The matrices $\tilde{K}_I$, $\tilde{K}_P$, and $\tilde{K}_D$ are the rough tuning matrices, obtained by simple experiments. Basic approaches for their computation will be discussed subsequently.

An appropriate choice for a pure integral controller was presented in [4]:

$$\tilde{K}_I = G_u^{-1}(0) = \big(-CA^{-1}B_u\big)^{-1}\,. \tag{13}$$

$\tilde{K}_I$ (13) is the inverse of the static gain $G_u(0)$ of the process. The reason for that approach is as follows. If A is stable and rank $G_u(0) = r$ then there exists an $\alpha > 0$ such that the closed loop system is stable. There is a secondary effect. The process is decoupled at steady-state.

In order to improve the dynamic behavior of the feedback system, i. e. improve the speed of the response, a proportional gain can be added. A PI controller is obtained. If the I-controlled system is stable, an arbitrary proportional matrix $\tilde{K}_P$ and a sufficiently small β provide a stable closed loop. One approach for the P-term [3] is given as

$$\tilde{K}_P = G_u^{-1}(0) \tag{14}$$

which is the same as in the integral gain, where the process is decoupled by the controller at steady-state. Another approach which allows to decouple the control loop between $y(t)$ and $r(t)$ for small time values is presented in [16], namely:

$$\tilde{K}_P = \big(CB_u\big)^{-1}\,. \tag{15}$$

$\tilde{K}_P$ (15) represents the inverse of the impulse response matrix of the process (1) at $t = 0$. CB_u can also be identified by measuring the slopes of the process step response at $t = 0$. If the inverse of the impulse response matrix at $t = 0$ does not exist, the impulse response or slopes of the step response, respectively, can be evaluated at a small $t_0 > 0$ instead of $t = 0$ [12]:

$$\tilde{K}_P = \big(Ce^{At_0}B_u\big)^{-1}\,. \tag{16}$$

Bereitgestellt von | Max-Planck-Gesellschaft - WIB6417
Angemeldet | 193.175.53.21
Heruntergeladen am | 25.03.14 15:44

An alternative approach, when (15) cannot be determined correctly or does not exist, is Thompson's [19] proportional gain

$$\tilde{K}_P = \left[CA^{-1}\left(e^{At_e}-1\right)B_u\right]^{-1} \tag{17}$$

that decouples the process over its midrange of frequencies. t_e is the average e-folding time and represents the dominant time constant of the process. (17) is the inverse of the unit step response matrix

$$H(t) = CA^{-1}\left(e^{At}-1\right)B_u \tag{18}$$

of the process (1) at $t = t_e$.

For the derivative gain, the same matrix as in the proportional gain (15) can be used [11] to decouple the control loop between $y(t)$ and $r(t)$ for small time values if $||K_D||$ is kept relatively small:

$$\tilde{K}_D = \left(C\,B_u\right)^{-1} . \tag{19}$$

If (19) does not exist, the problem is solved as for the proportional gain (16)

$$\tilde{K}_D = \left(Ce^{At_0}B_u\right)^{-1} \tag{20}$$

with a small $t_0 > 0$. It is suggested in [19] to apply the derivative action only to each main loop. The appropriate rough tuning matrix is given by:

$$\tilde{K}_D = I . \tag{21}$$

In this contribution the rough tuning approach for the integral and proportional gain (13), (14) is also used for the derivative term to follow the concept of the static decoupling of the process:

$$\tilde{K}_D = G_u^{-1}(0) . \tag{22}$$

As mentioned above, the rough tuning matrices can be determined by means of simple open loop experiments. For example, the static gain of a process is usually known. The assignment of the fine tuning parameters is an heuristic approach. The basic idea in this article is to determine the fine tuning parameters α, β, δ using IFT. Therefore, a new auto-tuning method for the MIMO PID controller design is obtained. The advantage is that only three parameters have to be tuned, independently from r and m.

If more degrees of freedom are required, the approaches for the proportional and derivative action can be extended as follows:

$$K_P = \tilde{K}_P \begin{pmatrix} \beta_1 & & 0 \\ & \ddots & \\ 0 & & \beta_r \end{pmatrix} , \tag{23}$$

$$K_D = \tilde{K}_D \begin{pmatrix} \delta_1 & & 0 \\ & \ddots & \\ 0 & & \delta_r \end{pmatrix} . \tag{24}$$

3 Iterative Feedback Tuning for Linear Multivariable Systems

Iterative feedback tuning is an optimization based model-free method for finding controller parameters only by means of closed loop experiments. In this section, the IFT method is described for multivariable linear time invariant processes based on [5]. Compared to [5], an input and error based deterministic cost function and the unity feedback control system (Fig. 1) are considered.

Controller parameters which have to be determined are collected in a vector

$$p = \begin{pmatrix} p_1 \\ \vdots \\ p_{n_p} \end{pmatrix} \in \boldsymbol{R}^{n_p} , \tag{25}$$

where n_p is the number of parameters. Because of the parameter dependent controller $C(s,p)$ the control signals $E(s,p)$, $U(s,p)$, and $Y(s,p)$ are also parameter dependent. The parameters are determined by solving the optimization problem

$$\min_{\text{w.r.t. } p\in \mathrm{R}^{n_p}} J(p) \tag{26}$$

subject to $C(s,p)$ stabilizing the closed loop. The deterministic cost function

$$J(p) = \frac{1}{2T_f}\int_0^{T_f}\left\{e^T(t,p)\,Q\,e(t,p) + u^T(t,p)\,R\,u(t,p)\right\}dt \tag{27}$$

is a quadratic criterion of the control error vector $e(t,p)$ and the manipulated variable vector $u(t,p)$. Here, T_f is the considered time interval or the duration of the tuning experiment respectively. Q and R are diagonal weighting matrices. In order to find the optimal controller parameters, the following set of equations has to be solved:

$$0 = \frac{\partial J(p)}{\partial p} = \frac{1}{T_f}\int_0^{T_f}\left\{\left[\frac{\partial e(t,p)}{\partial p}\right]^T Q\,e(t,p) + \cdots \right.$$
$$\left. \cdots + \left[\frac{\partial u(t,p)}{\partial p}\right]^T R\,u(t,p)\right\}dt . \tag{28}$$

(28) represents the necessary condition for optimality of the unconstrained problem (26). The solution of (28) can be computed iteratively by:

$$p_{k+1} = p_k - \gamma_k H_k^{-1}\frac{\partial J(p_k)}{\partial p} . \tag{29}$$

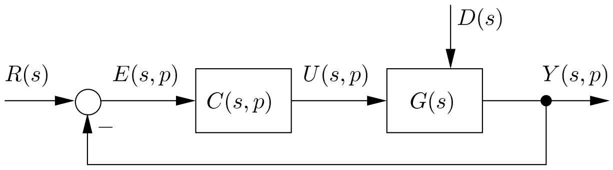


Figure 1 Unity feedback control system.

Bereitgestellt von | Max-Planck-Gesellschaft - WIB6417
Angemeldet | 193.175.53.21
Heruntergeladen am | 25.03.14 15:44

k denotes the current iteration step. γ_k is a positive real scalar and represents the step size. H_k is a positive definite matrix.

The gradient $\partial J(p)/\partial p$ (28) can be determined experimentally as demonstrated below. $e(t,p)$ and $u(t,p)$ are measured directly (see Fig. 1). $\partial e(t,p)/\partial p$ and $\partial u(t,p)/\partial p$ can be recorded by doing some additional experiments.

For convenience, the derivation is done in the Laplace domain. The derivatives of the control law

$$U(s,p) = C(s,p)E(s,p)\,, \tag{30}$$

the control error

$$E(s,p) = R(s) - Y(s,p)\,, \tag{31}$$

and the process output

$$Y(s,p) = G(s)\begin{pmatrix} U(s,p) \\ D(s) \end{pmatrix} \tag{32}$$

with respect to the parameter vector p are:

$$Y'(s,p) = G\begin{pmatrix} U'(s,p) \\ 0 \end{pmatrix}, \tag{33}$$

$$U'(s,p) = C(s,p)E'(s,p) + C'(s,p)E(s,p)\,, \tag{34}$$

$$E'(s,p) = 0 - Y'(s,p)\,. \tag{35}$$

$'$ denotes $\partial/\partial p$. The obtained equations are presented as a block diagram in Fig. 2.

It can be seen that the gradients are measured in a control loop where the error of the first experiment (see Fig. 1) is the input signal. Because of the block $C'(s,p)$, n_p additional experiments related to the derivatives of the controller with respect to each parameter $\partial C(s,p)/\partial p_i$, $i = 1, \ldots n_p$ have to be conducted. All in all, $1 + n_p$ tests have to be performed per iteration.

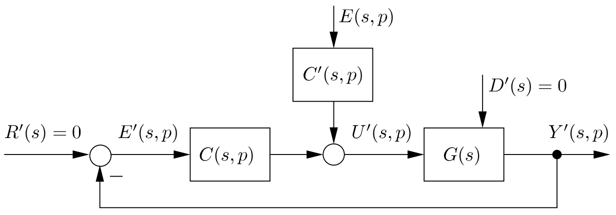


Figure 2 Gradient experiment.

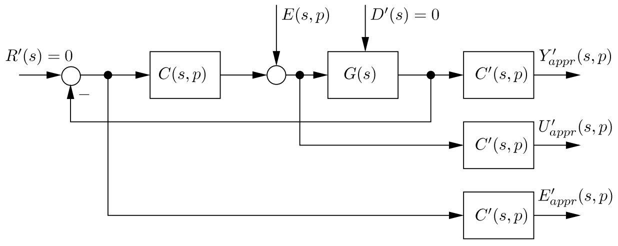


Figure 3 Approximated gradient experiment.

It is suggested in [9] to use an approximated gradient to reduce the number of additional experiments. The number of such tests can be reduced when moving the block $C'(s,p)$ in Fig. 2 to the outputs (see Fig. 3).

Only one additional gradient experiment has to be performed per iteration. Therefore, $1+1=2$ experiments are necessary per iteration independently from the number of parameters n_p. The disadvantage of that strategy is that a commutation error occurs.

4 Multivariable PID Controller Auto-Tuning Using IFT

In order to determine the fine tuning parameters α, β, and δ in an auto-tuning way, the IFT method is applied to the classical PID controller tuning approach. For that purpose, the PID controller is parameterized as follows:

$$u(t) = p_1 \tilde{K}_P\, e(t) + p_2 \tilde{K}_I \int_0^t e(\tau)\, d\tau + p_3 \tilde{K}_D \frac{de(t)}{dt}\,, \tag{36}$$

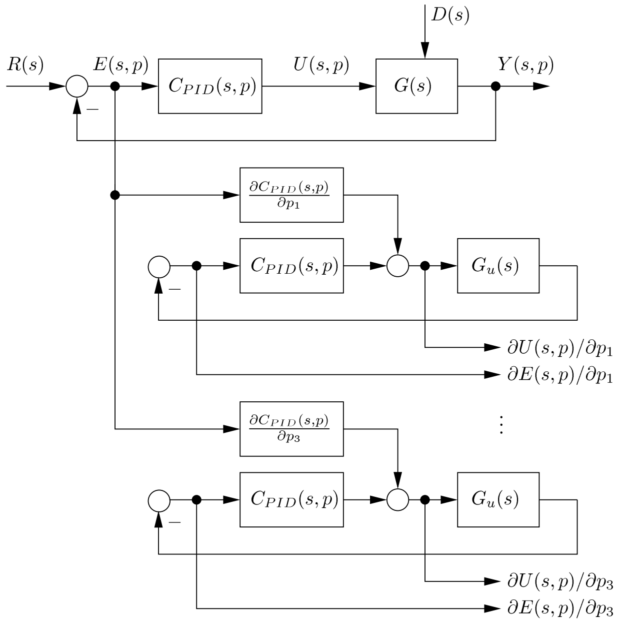
$$C_{\mathrm{PID}}(s,p) = p_1 \tilde{K}_P + p_2 \tilde{K}_I \frac{1}{s} + p_3 \tilde{K}_D\, s\,. \tag{37}$$

The parameters are collected in a vector with $n_p = 3$ elements:

$$p = \begin{pmatrix} p_1 \\ p_2 \\ p_3 \end{pmatrix} = \begin{pmatrix} \beta \\ \alpha \\ \delta \end{pmatrix}. \tag{38}$$

The experimental setup for finding the values of the parameter vector p (38) results from the general gradient experiment (Fig. 2), and is demonstrated in Fig. 4.

In a first experiment, the feedback control system is operated with an initial PID controller and a convenient input test signal (see Fig. 4, upper control loop). The



$D(s)$
$R(s)$
$E(s,p)$
$C_{PID}(s,p)$
$U(s,p)$
$G(s)$
$Y(s,p)$
$-$
$\frac{\partial C_{PID}(s,p)}{\partial p_1}$
$C_{PID}(s,p)$
$G_u(s)$
$\partial U(s,p)/\partial p_1$
$\partial E(s,p)/\partial p_1$
$\vdots$
$\frac{\partial C_{PID}(s,p)}{\partial p_3}$
$C_{PID}(s,p)$
$G_u(s)$
$\partial U(s,p)/\partial p_3$
$\partial E(s,p)/\partial p_3$

Figure 4 PID auto-tuning: the complete experiment.

Bereitgestellt von | Max-Planck-Gesellschaft - WIB6417
Angemeldet | 193.175.53.21
Heruntergeladen am | 25.03.14 15:44

resulting time responses $e(t,p)$ and $u(t,p)$ are recorded. The error signal $e(t,p)$ is then used as an input signal for $n_p = 3$ additional closed loop experiments. The derivatives of the vectors $e(t,p)$ and $u(t,p)$ with respect to each parameter can be measured as time responses in these experiments:

$$\frac{\partial e(t,p)}{\partial p_i} = \begin{pmatrix} \dfrac{\partial e_1(t,p)}{\partial p_i} \\ \vdots \\ \dfrac{\partial e_m(t,p)}{\partial p_i} \end{pmatrix}, \quad i = 1,\dots,3 \tag{39}$$

$$\frac{\partial u(t,p)}{\partial p_i} = \begin{pmatrix} \dfrac{\partial u_1(t,p)}{\partial p_i} \\ \vdots \\ \dfrac{\partial u_m(t,p)}{\partial p_i} \end{pmatrix}, \quad i = 1,\dots,3\,. \tag{40}$$

The recorded individual derivatives $\partial e_j(t,p)/\partial p_i$ and $\partial u_j(t,p)/\partial p_i$ with $j = 1,\dots,m$ and $i = 1,\dots,3$ in (39) and (40) are elements of the $m \times 3$ matrices $\partial e(t,p)/\partial p$ and $\partial u(t,p)/\partial p$ which are necessary to calculate the gradient $\partial J(p)/\partial p$ (28). All in all, $1 + n_p = 1 + 3 = 4$ experiments have to be performed per iteration. After that number of tests the parameters of the multivariable PID controller (36), (37) are updated using (29).

The partial derivatives of the controller transfer function matrix $C(s,p)$ (37) with respect to the parameters are needed for the additional gradient experiments. They are given by:

$$C'_{\text{PID}}(s,p) = \frac{\partial C_{\text{PID}}(s,p)}{\partial p} = \begin{pmatrix} \tilde{K}_P \\ \tilde{K}_I \frac{1}{s} \\ \tilde{K}_D\, s \end{pmatrix}. \tag{41}$$

The number of experiments can be reduced when using the approximated gradient as discussed in Sect. 2. The corresponding experiment is described in Fig. 5. Only $1 + 1 = 2$ experiments are necessary per iteration.

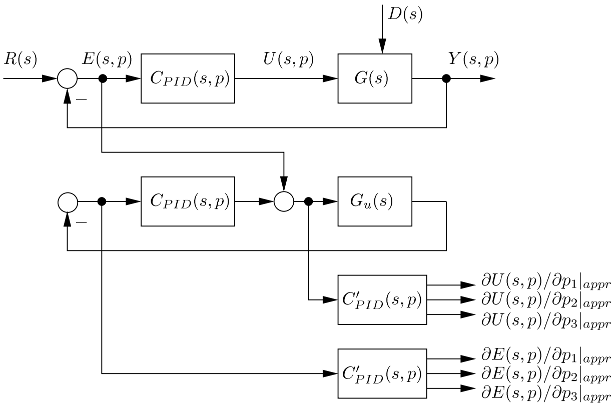

$D(s)$
$R(s)$
$E(s,p)$
$C_{PID}(s,p)$
$U(s,p)$
$G(s)$
$Y(s,p)$
$-$
$C_{PID}(s,p)$
$G_u(s)$
$-$
$C'_{PID}(s,p)$
$\partial U(s,p)/\partial p_1|_{appr}$
$\partial U(s,p)/\partial p_2|_{appr}$
$\partial U(s,p)/\partial p_3|_{appr}$
$C'_{PID}(s,p)$
$\partial E(s,p)/\partial p_1|_{appr}$
$\partial E(s,p)/\partial p_2|_{appr}$
$\partial E(s,p)/\partial p_3|_{appr}$

Figure 5 PID auto-tuning: the approximated gradient experiment.

The introduced design procedure can be summarized as follows.

1. Initialization phase ($k = 0$): choose initial parameter vector $p(k = 0)$, weighting matrices Q and R, time interval T_f, and convergence tolerance ϵ.
2. Compute gradient $\partial J(p_k)/\partial p$ (28) by means of experiments (see Fig. 4 or Fig. 5), if $|\partial J(p_k)/\partial p| < \epsilon$ then stop
3. Parameter update: determine γ_k and H_k, compute p_{k+1} (29), $k = k + 1$, go to 2.

5 Design Example

The introduced method is demonstrated with the following simulation example. A steam generator [17] that produces steam of high pressure and temperature to drive a steam turbine for the generation of electricity in a thermal power station is considered. The manipulated variables $u_1(t)$ and $u_2(t)$ are fuel feed and water supply. The controlled variables $y_1(t)$ and $y_2(t)$ are the steam pressure and the steam enthalpy. The disturbance $d(t)$ includes the heating value of the fuel and the slagging of the boiler. The dynamic behavior of the steam generator can be described by a linearized state-space system with the following matrices:

$$A = \begin{pmatrix} -15 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 22.5 & 0 & -3.06 & 0 & 0 & 0 & 0 & 0 & 0 \\ 14.505 & 5 & -6.246 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -4.5 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 15 & -16.4835 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2.19 & 0 & -18.12 & 0 & 0 \\ 0 & 0 & 0 & 0 & 37.155 & 15 & -32.967 & 0 & 0 \\ 14.505 & 5 & 0.7728 & 0 & 2.478 & 1 & -1.7299 & -1.8183 & 0 \\ 18.4095 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -5.4546 \end{pmatrix}$$

$$B_u = \begin{pmatrix} 10 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 60 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & -19.092 \end{pmatrix}, \quad B_d = \begin{pmatrix} 0 \\ 15 \\ 9.67 \\ 0 \\ 0 \\ 0 \\ 0 \\ 9.67 \\ 12.273 \end{pmatrix}$$

$$C = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}. \tag{42}$$

All state variables represent deviations from their operating points. The model contains a scaling of time. One time unit represents a five minute interval. A controller is needed that compensates a step change in the disturbance signal after one time unit.

The rough tuning matrices are chosen as the inverse of the static gain matrix of the process (13):

$$\begin{aligned} \tilde{K}_P = \tilde{K}_I = \tilde{K}_D = G_u^{-1}(0) &= \begin{pmatrix} 18.922 & 0.384 \\ 2.25 & -3.5002 \end{pmatrix}^{-1} \\ &= \begin{pmatrix} 0.0522 & 0.0057 \\ 0.0335 & -0.282 \end{pmatrix}. \end{aligned} \tag{43}$$

Bereitgestellt von | Max-Planck-Gesellschaft - WIB6417
Angemeldet | 193.175.53.21
Heruntergeladen am | 25.03.14 15:44

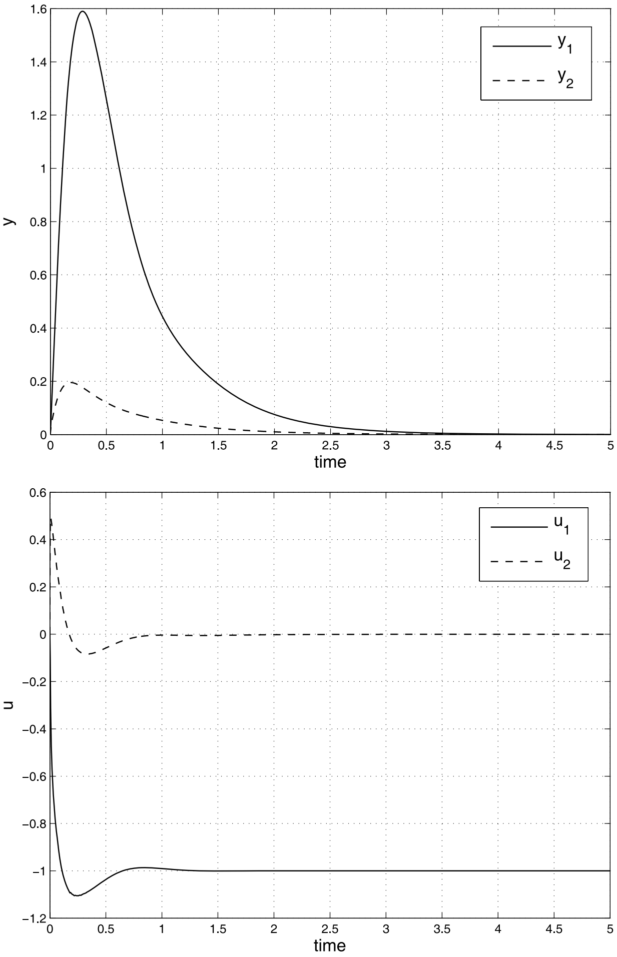


Figure 6 Closed loop disturbance responses, initial MIMO PID controller.

The initial fine tuning parameters are determined as:

$$\alpha_0 = 15,\ \beta_0 = 10,\ \delta_0 = 1\,. \tag{44}$$

For the derivative action the filter $G_F(s) = 1/(0.01s+1)$ is applied. The disturbance step responses using the initial controller are shown in Fig. 6.

Now, the iteration procedure is described. In this theoretical study, the IFT experiments are simulation experiments. During the first experiment a step change in the disturbance $d(t) = \sigma(t)$ is considered. The reference input of the feedback system does not change: $r(t) = \begin{pmatrix}0 & 0\end{pmatrix}^T$. The duration of one tuning experiment is $T_f = 4$, because after that duration the closed loop variables reach their steady-state values with the initial controller (see Fig. 6). Q and R are set to identity. An approximation of the Hessian of $J(p)$ is used for H_k. It contains only the first derivatives of $e(t,p)$ and $u(t,p)$ with respect to p. The second derivatives are neglected. Therefore, the matrix H_k can also be computed by means of the measured gradients. A step size of $\gamma = 1$ is taken for the iterative computation of the parameter vector (29). All integrals are approximated using a summation. After 11 iterations, the cost function is reduced from $J_0 = 0.6605$ to its minimum of $J = 0.5243$. The optimal fine tuning parameter values are found to be

$$\alpha = 197.272,\ \beta = 42.7738,\ \delta = 3.5509\,. \tag{45}$$

Approximately the same minimum, $J = 0.5245$, was achieved after 11 iterations with the approximated gradient. The tuning parameters change a little:

$$\alpha = 223.5653,\ \beta = 43.9733,\ \delta = 3.0591\,. \tag{46}$$

To compare the results, two reference control systems are considered. At first, two decentralized PID controllers are designed. The controller parameters are determined using sequential relay feedback with Tyreus-Luyben tuning rules. The controller parameters are given by:

$$K_{P1} = 1.429,\ K_{I1} = 2.3198,\ K_{D1} = 0.0635\,,$$

$$K_{P2} = 7.145,\ K_{I2} = 231.9805,\ K_{D2} = 0.0158\,. \tag{47}$$

The second reference controller is a MIMO PI controller designed using a pole placement (i. e. model based) design approach from [17]. The controller matrix gains are:

$$K_P = \begin{pmatrix} 2.828 & 13.51 \\ 0.145 & -3.167 \end{pmatrix},\ K_I = \begin{pmatrix} 10.86 & 51.3 \\ 1.459 & -5.890 \end{pmatrix}. \tag{48}$$

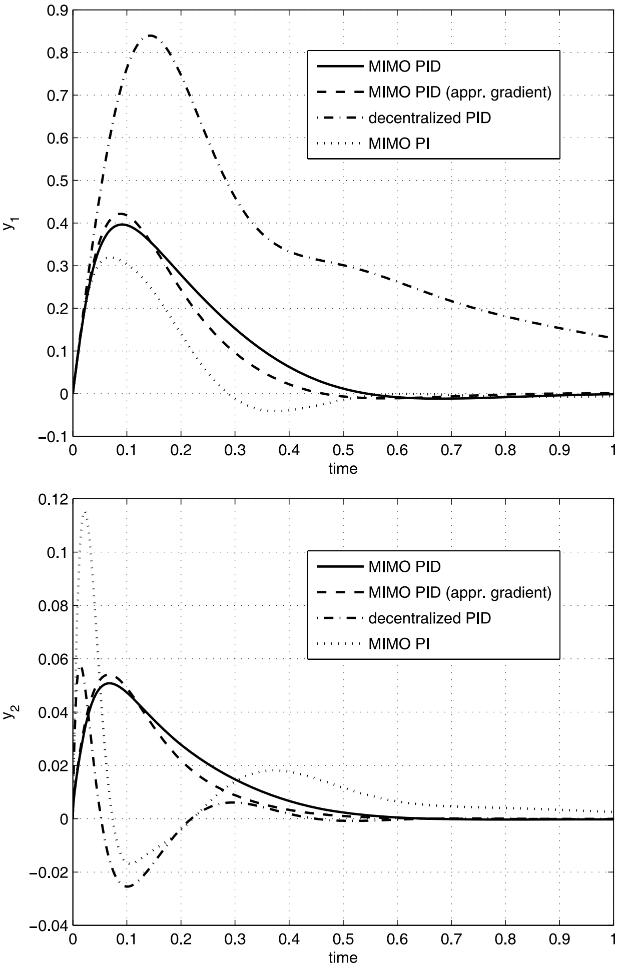


Figure 7 Closed loop disturbance responses, controlled variables.

Bereitgestellt von | Max-Planck-Gesellschaft - WIB6417
Angemeldet | 193.175.53.21
Heruntergeladen am | 25.03.14 15:44

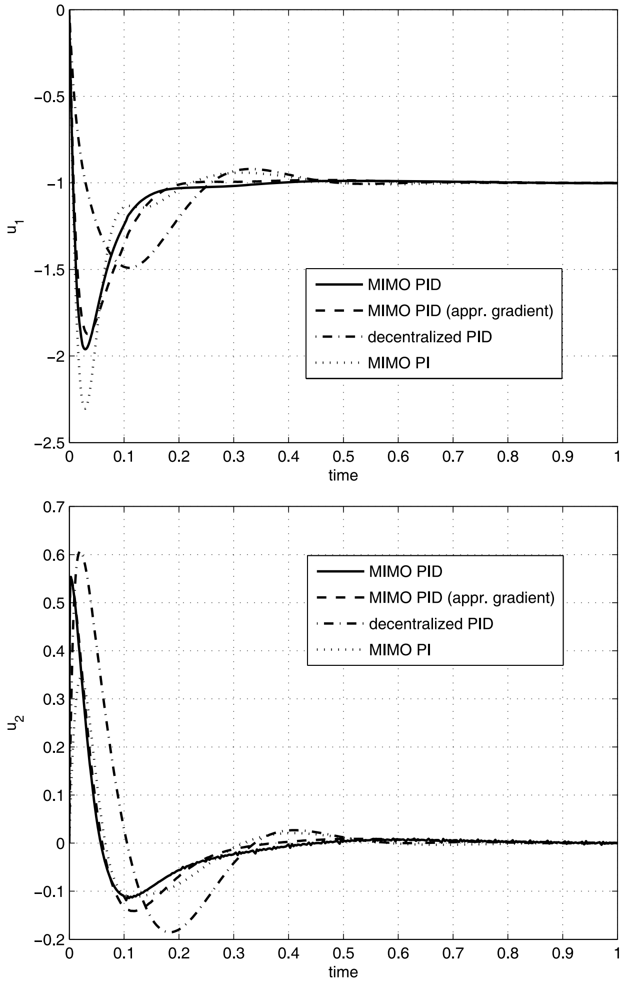


Figure 8 Closed loop disturbance responses, manipulated variables.

The simulation results of the disturbance responses are presented in Figs. 7 and 8.

Compared to the decentralized reference control scheme, the disturbance is rejected after one time unit as desired with the multivariable PID controllers designed using IFT. The controller computed with the approximated gradient provides only slightly different results. The reference MIMO PI controller gives similar results regarding the output response $y_1(t)$ with a larger amplitude in the curve of the manipulated variable $u_1(t)$ when compared with the MIMO PID control results. The MIMO PI controller provides less positive results concerning $y_2(t)$ with smaller amplitudes in $u_2(t)$ when compared to both MIMO PID control and decentralized PID control.

6 Conclusion

A new model-free auto-tuning approach for centralized PID controllers based on IFT is presented. It is shown that IFT can be successfully applied to the tuning of classical multivariable PID controllers. Only three parameters have to be determined during the auto-tuning procedure independently from the number of process inputs and outputs. The number of experiments can be decreased by using an approximated gradient.

The method is demonstrated and validated with a steam generator example. It confirms the benefit of the proposed strategy. Very good simulation results are obtained.

As a future work, the convergence of the method using the approximated gradient, especially for the considered controller structure, has to be proved. Furthermore, the introduced new auto-tuning procedure has to be tested with a real plant.

References

[1] K. Åström and T. Hägglund, *Automatic tuning of simple regulators with specifications on phase and amplitude margins*. In: Automatica, 20(5):645–651, 1984.

[2] K. Åström and T. Hägglund, *Advanced PID Control*. ISA, 2006.

[3] E. Davison, P. Taylor, and J. Wright, *On the application of tuning regulators to control a commercial heat exchanger*. In: IEEE Transactions on Automatic Control, 25(3):361–375, 1980.

[4] E. J. Davison, *Multivariable tuning regulators: The feedforward and robust control of a general servomechanism problem*. In: IEEE Transactions on Automatic Control, 21(1):35–47, 1976.

[5] H. Hjalmarsson, *Efficient tuning of linear multivariable controllers using iterative feedback tuning*. In: Int. J. Adapt. Control Signal Process., 13:553–572, 1999.

[6] H. Hjalmarsson, *Iterative feedback tuning – an overview*. In: Int. J. Adapt. Control Signal Process., 16:373–395, 2002.

[7] H. Hjalmarsson, M. Gevers, S. Gunnarsson, and O. Lequin, *Iterative Feedback Tuning: Theory and Applications*. In: IEEE Control Systems Magazine, pages 26–41, August 1998.

[8] H. Hjalmarsson, S. Gunnarsson, and M. Gevers, *A convergent iterative restricted complexity control design scheme*. In: Proc. 33rd IEEE CDC, pages 1735–1740, Orlando, Florida, 1994.

[9] H. Jansson and H. Hjalmarsson, *Gradient approximations in iterative feedback tuning for multivariable processes*. In: Int. J. Adapt. Control Signal Process., 18:665–681, 2004.

[10] M. Johnson and M. Moradi, *PID Control*. Springer, 2005.

[11] H. N. Koivo, *Tuning of a multivariable PID-controller for unknown systems*. In: Proc. 19th IEEE CDC, pages 1158–1159, 1980.

[12] H. N. Koivo and S. Pohjolainen, *Tuning of multivariable PI controllers for unknown systems with input delay*. In: Automatica, 21(1):81–91, 1985.

[13] H. N. Koivo and J. T. Tanttu, *Tuning of PID controllers: Survey of SISO and MIMO techniques*. In: Proc. of the IFAC Congress on Intelligent Tuning and Adaptive Control, pages 75–80, Singapore, 1991.

[14] U. Korn and U. Jumar, *PI-Mehrgrößenregler*. Oldenbourg, 1991.

[15] J. Lunze, *Regelungstechnik 2*. Springer, 6th edition, 2010.

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

[17] G. Roppenecker, *Zeitbereichsentwurf linearer Regelungen*. Oldenbourg, 1990.

[18] J. T. Tanttu and J. Lieslehto, *A comparative study of some multivariable PI controller tuning methods*. In: Proc. of the IFAC Congress on Intelligent Tuning and Adaptive Control, pages 357–362, Singapore, 1991.

[19] S. Thompson, *Multivariable PID controller for unidentified plant*. In: Journal of Dynamic Systems, Measurement, and Control, 104(3):270–274, 1982.

[20] Q.-G. Wang, B. Zou, T.-H. Lee, and Q. Bi, *Auto-Tuning of Multivariable PID Controllers from Decentralized Relay Feedback*. In: Automatica, 33(3):319–330, 1997.

[21] Cheng-Ching Yu, *Autotuning of PID Controllers*. Springer, 2nd edition, 2006.

Received: November 15, 2011

Bereitgestellt von | Max-Planck-Gesellschaft - WIB6417
Angemeldet | 193.175.53.21
Heruntergeladen am | 25.03.14 15:44

Dr.-Ing. Steffen Sommer is a research assistant at the Institute for Automation Engineering at the University of Magdeburg. Main research fields: control of reactive distillation processes, auto-tuning of multivariable controllers, energy management.

Address: Institute for Automation Engineering,
Otto-von-Guericke University Magdeburg, Uni-Platz 2, 39106 Magdeburg, Germany,
e-mail: steffen.sommer@ovgu.de

Prof. Dr.-Ing. Achim Kienle is head of the chair for Automation/Modeling at the Institute for Automation Engineering at the University of Magdeburg and head of the Process Synthesis and Process Dynamics research group at the Max Planck Institute for Dynamics of Complex Technical Systems Magdeburg. Main research fields: Modeling and control of chemical and biological systems, energy management.

Address: Institute for Automation Engineering,
Otto-von-Guericke University Magdeburg, Uni-Platz 2, 39106 Magdeburg, Germany,
e-mail: achim.kienle@ovgu.de

Leichter Einstieg in den Hardwareentwurf

Harald Flügel
FPGA-Design mit Verilog
2010 | 229 S. | Br. | € 34,80
ISBN 978-3-486-59234-4

Ein Großteil elektronischer Steuerung wird heute mit FPGAs (Field Programmable Gate Arrays) realisiert. Entworfen werden diese Bausteine mit Hardwarebeschreibungssprachen. Verilog ist eine sehr verbreitete und für Einsteiger gut geeignete Beschreibungssprache.

Dieses Buch führt in die Methode der FPGA-Entwicklung mit der Sprache Verilog ein und macht Neulingen den Einstieg so einfach wie möglich. Es erläutert die grundlegenden Konzepte der Sprache Verilog und stellt in einer Reihe praxisnaher Beispiele die Designmethoden der FPGA-Programmierung vor.

Der Einstieg in die FPGA-Programmierung – so leicht wie möglich.

Das Buch richtet sich an Studierende der Elektrotechnik und Informatik sowie an Hardwareentwickler in der Praxis.

Bestellen Sie in Ihrer Fachbuchhandlung oder
direkt bei uns: Tel: 089/45051-248, Fax: 089/45051-333
verkauf@oldenbourg.de

Oldenbourg

Bereitgestellt von | Max-Planck-Gesellschaft - WIB6417
Angemeldet | 193.175.53.21
Heruntergeladen am | 25.03.14 15:44