ITERATIVE FEEDBACK TUNING OF PID CONTROLLER

SOBOTA JAROSLAV, SCHLEGEL MILOŠ
Department of Cybernetics, Faculty of Applied Sciences,
University of West Bohemia in Pilsen
jsobota@kky.zcu.cz, schlegel@kky.zcu.cz

Abstract: This paper is focused on iterative tuning of the 2DOF PID controller to minimize given quadratic cost function of the system output error and control effort. The main ideas of Iterative Feedback Tuning are introduced and some important modification to model-free iterative PID controller design is presented. In particular, the tuning process is divided into two parts. Firstly, the classical 1DOF PID controller is iteratively tuned. The integral and derivative time constants are not tuned independently of each other, but they are related by a ratio coefficient to avoid cancellation of the derivative part of the controller even in a low noise environment. The remaining parameters of 2DOF PID are then tuned independently in the next iterative procedure. Theoretical results are verified using a laboratory coupled tanks model.

Keywords: iterative feedback tuning, PID controller, model-free control design, quadratic cost function, Simulink

1 INTRODUCTION

This paper is focused on auto-tuning of the two-degree-of-freedom (2DOF) proportional--plus-integral-plus-derivative (PID) controller where the goal of the controller design is expressed as minimization of given quadratic cost function of the system output error and control effort.

The tuning method described in this report is meant to be widely used in industry, therefore the performance criterion must be minimized without knowledge of the controlled plant and disturbance model, since these are seldom known. Such optimization requires iterative gradient--based minimization procedure. The gradient of the cost function with respect to the controller parameters is a complicated function of the plant and disturbances, which are unknown, so its computation is the main challenging problem.

It is shown, that the Iterative Feedback Tuning (IFT) method can be used to solve this problem. This control design methodology has emerged in the nineties as a result of intense research efforts aimed at bridging the gap between system identification and robust control analysis and design [1]. The gradient is estimated from signals obtained from simple closed loop experiments on the controlled plant and new controller parameters can be evaluated.

The first part of this paper introduces the main ideas of IFT and presents some important modification to model-free iterative PID controller design, in which the controller parameters are iteratively tuned on the basis of successive experiments performed on the real plant, leading to better and better closed-loop behaviour. In particular, the tuning process is divided into two parts. Firstly, the classical 1DOF PID controller is iteratively tuned. The integral and derivative time constants are not tuned independently of each other, but they are related by a ratio coefficient to avoid cancellation of the derivative part of the controller even in a low noise environment. The remaining parameters of 2DOF PID are then tuned independently in the next iterative procedure.

The next part is concerned with practical design choices to be made to fulfil common control design requests such as minimizing the settling time, minimizing overshoot, minimizing the control effort, etc. These choices involve time-weighting of obtained signals to minimize overshoot and/or frequency weighting to emphasize or suppress specific frequency bands.

The final section describes the Simulink function block of the IFT PID controller and its application to a laboratory coupled tanks model. The results obtained confirm that this method can significantly improve the control performance in just a few iterations.

The main advantage of the developed algorithm is the fact, that it is model-free. The only requirement is a controller which ensures stability of the closed loop at the beginning of the iterative tuning process. Another plus of this method is that the tuning of the controller parameters is completely performed on the closed-loop system and that it is applicable to a wide variety of processes, including integrating processes, systems with complex poles, non-minimum phase systems, nonlinear systems in the vicinity of the working point and even systems with non-smooth nonlinearity (dead zone, backlash, ...).

2 ITERATIVE FEEDBACK TUNING METHOD

We consider an unknown plant described by the discrete time model

$$y(t) = P_0u(t) + P_0d_I(t) + d_O(t) + v(t), \tag{1}$$

where P_0 is a linear time-invariant operator, d_I and d_O stand for input and output disturbances, v is a measurement noise and t represents the discrete time index. The noise $v(t)$ is a zero mean weakly stationary random process (its first and second moments are constant with time).see [?]).

This system is to be controlled by a two-degrees-of-freedom controller:

$$u(t) = C_r(\rho)r(t) - C_y(\rho)y(t), \tag{2}$$

where $C_r(\rho)$ and $C_y(\rho)$ are linear time-invariant transfer functions parameterized by vector of controller parameters ρ. The dimension of ρ depends on the parameter count. The signal $r(t)$ is an external deterministic reference signal and it is independent of $v(t)$. It is possible for $C_r(\rho)$ and $C_y(\rho)$ to have common parameters. A block diagram of the closed loop system is represented in Fig.1.

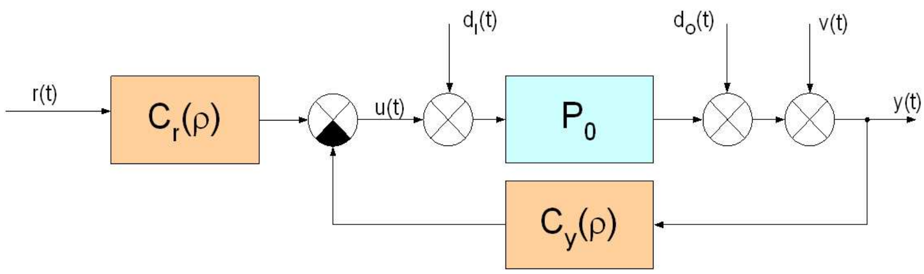

Figure 1 - Block diagram of the closed loop system

From now on, the ρ argument will be used to point out that the signal was obtained from the closed loop system with the controller $C(\rho) \triangleq \{C_r(\rho), C_y(\rho)\}$ operating. We will also omit the time argument of the signals to ease the notation. Thus, $y(\rho), u(\rho)$ will denote the output and the control input of the system (1) with controller (2) in feedback.

Let $\tilde{y}(\rho)$ be the difference between the achieved and the desired response of the closed loop system to a reference signal r:

$$\tilde{y}(\rho) \triangleq y(\rho) - y^D = \underbrace{\frac{C_r(\rho)P_0}{1 + C_y(\rho)P_0}}_{T_0(\rho)} r + \underbrace{\frac{P_0}{1 + C_y(\rho)P_0}}_{\frac{T_0(\rho)}{C_r(\rho)}} d_I + \underbrace{\frac{1}{1 + C_y(\rho)P_0}}_{S_0(\rho)} (d_O + v) - y^D. \quad (3)$$

This error consists of a contribution due to incorrect tracking of the reference signal r and an error due to the disturbances.

The control design objective for some controller of fixed structure parameterized by ρ can be naturally formulated as a minimization of some norm of $\tilde{y}(\rho)$ and control effort $u(\rho)$ over the controller parameters. We will use the following quadratic criterion:

$$J(\rho) = \frac{1}{2N} E\left[\sum_{t=1}^{N} (L_y \tilde{y}_t(\rho))^2 + \lambda \sum_{t=1}^{N} (L_u u_t(\rho))^2\right], \quad (4)$$

where the scalar λ stands for weighting of the control effort. The symbols L_y and L_u represent some filters, by which we can emphasize or suppress specific frequency bands of the output and control signals respectively. These filters can of course be omitted (i.e. set to 1) but they provide extra flexibility to the design. It is also useful to introduce some time-weighting in the criterion - it is an easy and efficient way to avoid oscillations in the closed loop system (this topic is discussed in more detail in section 4).

The goal of the control design is to find the optional controller parameters defined by

$$\rho^* = \arg\min_{\rho} J(\rho). \quad (5)$$

This means that we want to tune the closed loop response to some desired deterministic response of length N in a mean square sense.

If the desired closed loop response is defined as output of some reference model (i.e. $y^D = P_M r$), the problem is closely connected with the model reference adaptive control (MRAC). In MRAC it is necessary to know the model of the controlled plant to minimize a criterion of the same type as (4) with respect to controller parameters. The gradient of the criterion function is essential to such minimization and the model is used for its computation [2]. In contrary, the IFT approach [3] shows that the gradient can be obtained even if the model of the controlled process is unknown. It can be evaluated from input-output data collected on the real closed loop system during one special experiment.

2.1 Minimizing the criterion

To find the optimal controller parameters, we must solve the equation (5). This means we would like to find a solution for ρ to the equation

$$0 = \frac{\partial J}{\partial \rho}(\rho) = \frac{1}{N} E\left[\sum_{t=1}^{N} L_y \tilde{y}_t(\rho) L_y \frac{\partial \tilde{y}_t}{\partial \rho}(\rho) + \lambda \sum_{t=1}^{N} L_u u_t(\rho) L_u \frac{\partial u_t}{\partial \rho}(\rho)\right]. \quad (6)$$

If we were able to compute the gradient $\partial J/\partial\rho$ we could find the optimal parameters by the following iterative algorithm:

$$\rho_{i+1} = \rho_i - \gamma_i R_i^{-1}\frac{\partial J}{\partial\rho}(\rho), \tag{7}$$

where R_i is some appropriate positive definite matrix, typically a Gauss-Newton approximation of the Hessian of J and γ_i is a positive real scalar that determines the step size. The sequence γ_i must obey some constraints to ensure convergence to a local minimum (for details see [3]).

But how can we obtain all the signals needed for evaluation of the gradient in (6)? These can be captured during special experiments on the closed loop system formed by the real plant and the actual controller $C_r(\rho_i), C_y(\rho_i)$ in feedback.

2.1.1 Output related signals

We know from (3), that

$$\tilde{y}(\rho) = T_0(\rho)r + \frac{T_0(\rho)}{C_r(\rho)}d_I + S_0(\rho)(d_O + v) - y^D, \tag{8}$$

where S_0 and T_0 denote the sensitivity and complementary sensitivity function of the closed loop system. The derivative of $\tilde{y}(\rho)$ with respect to controller parameters is then given by

$$\begin{aligned}
\frac{\partial\tilde{y}}{\partial\rho}(\rho) &= \frac{\partial T_0}{\partial\rho}(\rho)r + \frac{\partial\left(\frac{T_0(\rho)}{C_r(\rho)}\right)}{\partial\rho}d_I + \frac{\partial S_0}{\partial\rho}(\rho)(d_O+v) = \frac{\partial y}{\partial\rho}(\rho) &(9)\\
\frac{\partial T_0}{\partial\rho}(\rho) &= \frac{\frac{\partial C_r}{\partial\rho}(\rho)P_0}{1+C_y(\rho)P_0} - \frac{\frac{\partial C_y}{\partial\rho}(\rho)P_0C_r(\rho)P_0}{(1+C_y(\rho)P_0)^2} = \frac{1}{C_r(\rho)}\frac{\partial C_r}{\partial\rho}(\rho)T_0(\rho) - &(10)\\
&\quad -\frac{1}{C_r(\rho)}\frac{\partial C_y}{\partial\rho}(\rho)T_0^2(\rho)\\
\frac{\partial\left(\frac{T_0(\rho)}{C_r(\rho)}\right)}{\partial\rho} &= -\frac{P_0^2\frac{\partial C_y}{\partial\rho}(\rho)}{(1+C_y(\rho)P_0)^2} = -\frac{1}{C_r^2(\rho)}\frac{\partial C_y}{\partial\rho}(\rho)T_0^2(\rho) &(11)\\
\frac{\partial S_0}{\partial\rho}(\rho) &= -\frac{P_0\frac{\partial C_y}{\partial\rho}(\rho)}{(1+C_y(\rho)P_0)^2} = -\frac{1}{C_r(\rho)}\frac{\partial C_y}{\partial\rho}(\rho)T_0(\rho)S_0(\rho) &(12)
\end{aligned}$$

Therefore (9) can be rewritten as follows:

$$\begin{aligned}
\frac{\partial y}{\partial\rho}(\rho) &= \frac{1}{C_r(\rho)}\frac{\partial C_r}{\partial\rho}(\rho)T_0(\rho)r - &(13)\\
&\quad -\frac{1}{C_r(\rho)}\frac{\partial C_y}{\partial\rho}(\rho)\left[\underbrace{T_0^2(\rho)r + \frac{T_0^2(\rho)}{C_r(\rho)}d_I + T_0(\rho)S_0(\rho)(d_O+v)}_{=T_0(\rho)y}\right] =\\
&= \frac{1}{C_r(\rho)}\left[\frac{\partial C_r}{\partial\rho}(\rho)T_0(\rho)r - \frac{\partial C_y}{\partial\rho}(\rho)T_0(\rho)y\right] =\\
&= \frac{1}{C_r(\rho)}\left[\left(\frac{\partial C_r}{\partial\rho}(\rho) - \frac{\partial C_y}{\partial\rho}(\rho)\right)T_0(\rho)r + \frac{\partial C_y}{\partial\rho}(\rho)T_0(r-y)\right]
\end{aligned}$$

The last term $(r-y)$ is the difference between the reference signal and the output of the closed loop system. The derivative $\partial y/\partial\rho(\rho)$ involves filtering of this signal through the closed loop system and that leads us to following idea: In each iteration i of the controller tuning, we will gather N-length vector of data under normal operating conditions. Next experiment will consist of using the signal $r-y$ constructed from the first experiment data as reference signal. The last experiment will use the same reference signal as the first experiment. Thus we have

$$r_i^1 = r, \quad y^1(\rho_i) = T_0(\rho_i)r + \frac{T_0(\rho_i)}{C_r(\rho_i)}d_{I_i}^1 + S_0(\rho_i)(d_{O_i}^1 + v_i^1), \tag{14}$$

$$r_i^2 = r - y^1(\rho_i), \quad y^2(\rho_i) = T_0(\rho_i)(r_i^1 - y^1(\rho_i)) + \frac{T_0(\rho_i)}{C_r(\rho_i)}d_{I_i}^2 + S_0(\rho_i)(d_{O_i}^2 + v_i^2), \tag{15}$$

$$r_i^3 = r, \quad y^3(\rho_i) = T_0(\rho_i)r + \frac{T_0(\rho_i)}{C_r(\rho_i)}d_{I_i}^3 + S_0(\rho_i)(d_{O_i}^3 + v_i^2) \tag{16}$$

If the disturbances and the noise in the second (15) and third (16) experiment were zero, we would get a perfect realization of signals needed in (13) to compute the sensitivity function $\partial y/\partial\rho(\rho)$. The disturbances and noise in the first experiment are not a problem. The noise in the last two experiments is not a big nuisance because of its zero mean. The only trouble are the disturbances d_I and d_O acting on the system during the second and third experiment. Since it is usually impossible to eliminate them on the real plant, the only thing we can do to improve the accuracy of evaluation of $\partial y/\partial\rho(\rho)$ is to inject some extra disturbances into the system during the first experiment (this increases the "signal to noise" ratio). Then we have

$$est\left[\frac{\partial y}{\partial\rho}(\rho_i)\right] \triangleq \frac{1}{C_r(\rho)}\left[\left(\frac{\partial C_r}{\partial\rho}(\rho_i) - \frac{\partial C_y}{\partial\rho}(\rho_i)\right)y^3(\rho_i) + \frac{\partial C_y}{\partial\rho}(\rho_i)y^2(\rho_i)\right] \tag{17}$$

and

$$\tilde{y}(\rho_i) = y^1(\rho_i) - y^D. \tag{18}$$

2.1.2 Input related signals

Now we need the signals $u(\rho)$ and $\partial u/\partial\rho$ required in (6). Output of the controller operating in the closed loop system is expressed as follows:

$$u(t) = \frac{C_r(\rho)}{1+P_0C_y(\rho)}r - \frac{P_0C_y(\rho)}{1+P_0C_y(\rho)}d_I - \frac{C_y(\rho)}{1+P_0C_y(\rho)}(d_O + v) \tag{19}$$

Therefore the control effort during the three experiments (14)-(16) is given by

$$u^1(\rho_i) = C_r(\rho_i)S_0(\rho_i)r - P_0C_y(\rho_i)S_0(\rho_i)d_{I_i}^1 - C_y(\rho_i)S_0(\rho_i)(d_{O_i}^1 + v_i^1) \tag{20}$$

$$\begin{aligned} u^2(\rho_i) &= C_r(\rho_i)S_0(\rho_i)r - C_r(\rho_i)S_0(\rho_i)y^2(\rho_i) - P_0C_y(\rho_i)S_0(\rho_i)d_{I_i}^2 - \\ &\quad -C_y(\rho_i)S_0(\rho_i)(d_{O_i}^2 + v_i^2) \end{aligned} \tag{21}$$

$$u^3(\rho_i) = C_r(\rho_i)S_0(\rho_i)r - P_0C_y(\rho_i)S_0(\rho_i)d_{I_i}^3 - C_y(\rho_i)S_0(\rho_i)(d_{O_i}^3 + v_i^3) \tag{22}$$

The following inference is focused on bringing out the sensitivity function $\partial u/\partial\rho$:

$$\frac{\partial u}{\partial\rho} = \frac{\partial C_r}{\partial\rho}(\rho)S_0(\rho)r + C_r(\rho)\frac{\partial S_0}{\partial\rho}(\rho)r - \frac{\partial C_y}{\partial\rho}(\rho)P_0S_0(\rho)d_I - \tag{23}$$

$$
\begin{aligned}
& -\frac{\partial S_0}{\partial \rho}(\rho) C_y P_0 d_I - \frac{\partial C_y}{\partial \rho}(\rho) S_0(\rho)(d_O + v) - \\
& -\frac{\partial S_0}{\partial \rho}(\rho) C_y(\rho)(d_O + v)
\end{aligned}
$$

$$
\frac{\partial S_0}{\partial \rho}(\rho) = -\frac{1}{C_r(\rho)} \frac{\partial C_y}{\partial \rho}(\rho) T_0(\rho) S_0(\rho) \tag{24}
$$

$$
\begin{aligned}
\frac{\partial u}{\partial \rho} &= S_0(\rho) \left[\frac{\partial C_r}{\partial \rho}(\rho) r - \frac{\partial C_y}{\partial \rho}(\rho) \left(T_0(\rho) r + P_0 d_I - \frac{C_y(\rho)}{C_r(\rho)} P_0 T_0(\rho) d_I \right) \right] - \\
& -S_0(\rho) \left[\frac{\partial C_y}{\partial \rho}(\rho) \left(\left(1 - \frac{C_y(\rho)}{C_r(\rho)} \right) T_0(\rho)(d_O + v) \right) \right] = \\
&= S_0(\rho) \left[\frac{\partial C_r}{\partial \rho}(\rho) r - \frac{\partial C_y}{\partial \rho}(\rho) \underbrace{\left(T_0(\rho) r + \frac{T_0(\rho)}{C_r(\rho)} d_I + S_0(\rho)(d_O + v) \right)}_{y(\rho)} \right] = \\
&= S_0(\rho) \left[\frac{\partial C_r}{\partial \rho}(\rho) r - \frac{\partial C_y}{\partial \rho}(\rho) y \right] = \\
&= \frac{\partial C_r}{\partial \rho}(\rho) S_0(\rho) r - \frac{\partial C_y}{\partial \rho}(\rho) S_0(\rho) r + \frac{\partial C_y}{\partial \rho}(\rho) S_0(\rho)(r - y)
\end{aligned} \tag{25}
$$

If we compare (25) to (21) and (22), we can see that, under the same assumptions as for the output related signals (zero disturbances and noise during the 2nd and 3rd experiment), the proposed experiments give us exactly what is needed for evaluation of $\partial u / \partial \rho$. Therefore we can write

$$
est\left[\frac{\partial u}{\partial \rho}(\rho_i) \right] \triangleq \frac{1}{C_r(\rho)} \left[\left(\frac{\partial C_r}{\partial \rho}(\rho_i) - \frac{\partial C_y}{\partial \rho}(\rho_i) \right) u^3(\rho_i) + \frac{\partial C_y}{\partial \rho}(\rho_i) u^2(\rho_i) \right] \tag{26}
$$

and

$$
u(\rho_i) = u^1(\rho_i). \tag{27}
$$

2.1.3 Gradient of the criterion

With the signals defined in the previous subsections, it is possible to formulate an experimentally based estimate of the gradient of J by taking

$$
est\left[\frac{\partial J}{\partial \rho}(\rho_i) \right] = \frac{1}{N} \sum_{t=1}^{N} \left(L_y \tilde{y}_t(\rho) L_y \ est\left[\frac{\partial y_t}{\partial \rho}(\rho) \right] + \lambda L_u u_t(\rho) L_u \ est\left[\frac{\partial u_t}{\partial \rho}(\rho) \right] \right) \tag{28}
$$

This estimate must be unbiased for the iterative algorithm to converge to a local minimum. For this reason we would need all three experiments even if no disturbances were acting on the controlled plant and the zero mean noise was present. If we used the signal $y^1(\rho_i)$ instead of $y^3(\rho_i)$ in (17) and $u^1(\rho_i)$ instead of $u^3(\rho_i)$ in (26), the obtained estimate of $\partial J / \partial \rho(\rho)$ would be biased [1].

2.1.4 Summary of the algorithm

We will just sum up the iterative algorithm in this subsection: With a controller $C(\rho_i) \triangleq \{C_r(\rho_i), C_y(\rho_i)\}$ operating on the plant, firstly generate the signals $y^1(\rho_i)$, $y^2(\rho_i)$, $y^3(\rho_i)$ (14)-(16) and $u^1(\rho_i)$, $u^2(\rho_i)$, $u^3(\rho_i)$ (20)-(22), where $d_{I_i}^1 >> d_{I_i}^2$, $d_{I_i}^1 >> d_{I_i}^3$, $d_{O_i}^1 >> d_{O_i}^2$ and $d_{O_i}^1 >> d_{O_i}^3$ if the disturbances are present. Then evaluate $\tilde{y}(\rho_i)$, $est\left[\frac{\partial y}{\partial \rho}(\rho_i)\right]$, $u(\rho_i)$ and $est\left[\frac{\partial u}{\partial \rho}(\rho_i)\right]$ using (18), (17), (27) and (26) and let the next controller parameters be computed by:

$$\rho_{i+1} = \rho_i - \gamma_i R_i^{-1} est\left[\frac{\partial J}{\partial \rho}(\rho_i)\right], \tag{29}$$

where $est[\partial J/\partial\rho(\rho_i)]$ is given by (28), where γ_i is a sequence of positive real numbers determining the step size and where R_i is a sequence of positive definite matrices, that are for example given by (51). Repeat the whole procedure, replacing i by $i+1$.

3 ITERATIVE FEEDBACK TUNING OF THE 2DOF PID CONTROLLER

This section is focused on application of the proposed algorithm to the standard 2DOF PID controller of the form

$$U(s) = K\left[bR(s) - Y(s) + \frac{R(s) - Y(s)}{T_i s} + \frac{T_d s\,(cR(s) - Y(s))}{\frac{T_d}{N_d}s + 1}\right], \tag{30}$$

where K, T_i, T_d, b, c and N_d are the controller parameters. But it is important to emphasize, that IFT is not limited to such simple controller. There is no limitation of the complexity (e.g. order or parameter count) of the controller except differentiability and computational capabilities. Our aim is to show, that this method can significantly improve the performance of the closed loop system without opening the control loop and with no additional hardware investments. We will illustrate it on a 2DOF PID, since PID control is still the most used in industry.

We have made some important modification to the above described iterative algorithm. Even if the IFT method is capable of tuning all the parameters of 2DOF PID at a time, our experience shows it is better to divide the tuning process into two subsequent procedures. Firstly, the classical 1DOF PID controller is iteratively tuned. The remaining parameters of 2DOF PID are then tuned independently in the next iterative procedure. This results in shorter experiments, because for a 1DOF controller where $C_r(\rho) = C_y(\rho)$ the equations (17) and (26) simplify and the third experiment (16) becomes unnecessary. Another simplification occurs when tuning the b and c controller parameters. The term $\partial C_y/\partial\rho(\rho)$ is then obviously zero, we can leave out the second experiment (15) and therefore we can use $y_t^1(\rho)$ instead of $y_t^3(\rho)$ in (17) and (26) so the third experiment is useless again. Another modification is that the integral and derivative time constants are not tuned independently of each other, but they are related by a ratio coefficient. Even if the noise is not dominant, the derivative part of the controller tends to zero without this restriction.

3.1 Closer look at the experiments

Now we will have a closer look at the experiments needed for iterative feedback tuning of the PID controller.

Firstly we would like to tune the 1DOF controller to optimal input disturbance rejection, so we inject some input disturbance $d_{I_i}^1$ into the system during the first experiment and we set

r_i^1 and $d_{O_i}^1$ to 0. The reference signal for the second experiment r_i^2 is defined by (15) and both $d_{I_i}^2$ and $d_{O_i}^2$ are set to zero. Equivalent technique would be used for output disturbance rejection.

Figure 2 shows how the system output during the first iteration could look like for some plant and some initial controller. The input disturbance is set to 1 for the first 15 seconds. Then it is set back to zero. The first experiment ends at t=30s and its output is used to create the reference signal for the second experiment. Note that the initial conditions of both experiments should be roughly identical and therefore we need stable closed loop at the beginning of the tuning process. After another 30 seconds the first iteration is completed and we are ready to estimate the gradient of the criterion with respect to controller parameters using the signals obtained during the second experiment and compute new controller parameters. The same procedure is then iteratively repeated with the updated controller until the algorithm converges to a local minimum or until satisfactory closed loop behavior is reached.

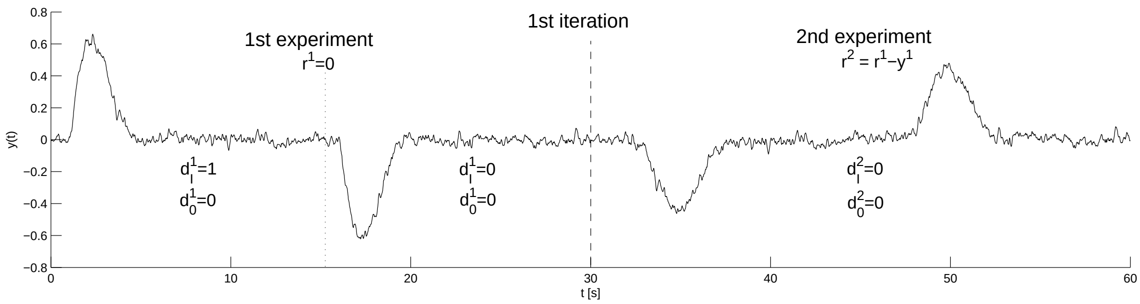

Figure 2 - First iteration of 1DOF PID tuning

The subsequent step is tuning of the setpoint change response of the closed loop system. We can shape the step response by modifying the parameters b and c without affecting previously tuned sensitivity function of the control loop. As mentioned above, the tuning iteration consists of just one experiment in this case. The setpoint is changed back and forth during this experiment and no disturbances are injected into the system. Figure 3 represents the output of the system during two consequent iterations. Note that the step response with an updated controller acting on the system improved.

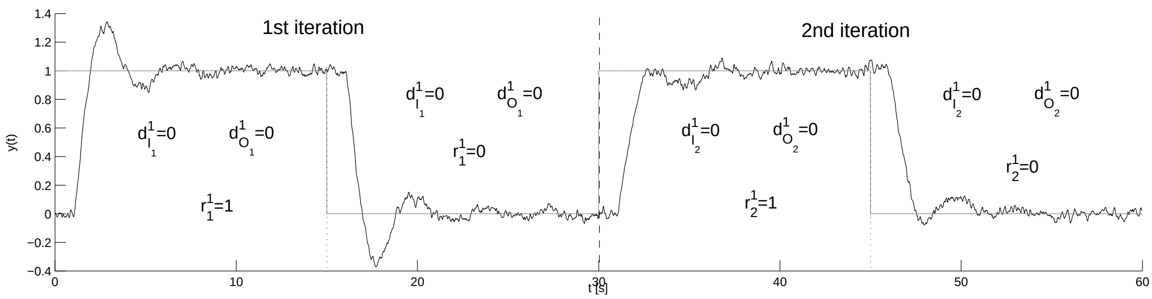

Figure 3 - Tuning of the step response - first two iterations

After bringing this insight into the problem, we will focus on the mathematical part in the following subsections.

3.2 Tuning of the 1DOF PID controller

For 1DOF PID, the parameters b and c are equal to one. The relation between T_i and T_d is defined by

$$T_d = f \cdot T_i, \tag{31}$$

where f is invariant during all experiments. The limitation of derivative gain N_d will remain constant as well. Thus we get from (30) with respect to (31) the following:

$$C_y(\rho) = C_r(\rho) = K\left[1 + \frac{1}{T_i s} + \frac{f T_i s}{\frac{f T_i}{N_d} s + 1}\right] = C(\rho) \tag{32}$$

Now we can compute all the terms needed in (17) and (26) to evaluate the estimates of $\frac{\partial y}{\partial \rho}(\rho)$ and $\frac{\partial u}{\partial \rho}(\rho)$:

$$\frac{1}{C(\rho)} = \frac{T_i}{K} \cdot \frac{f T_i s^2 + N_d s}{(1+N) f T_i^2 s^2 + (f + N_d) T_i s + N_d} \tag{33}$$

$$\frac{\partial C}{\partial \rho}(\rho) = \left[\frac{\partial C}{\partial K}(\rho),\ \frac{\partial C}{\partial T_i}(\rho)\right]^T \tag{34}$$

$$\frac{\partial C}{\partial K}(\rho) = \frac{(1+N_d) f T_i^2 s^2 + (f + N_d) T_i s + N_d}{f T_i^2 s^2 + N_d T_i s} \tag{35}$$

$$\frac{\partial C}{\partial T_i}(\rho) = \frac{K N_d^2 f s}{f^2 T_i^2 s^2 + 2 N_d f T_i s + N_d^2} - \frac{K}{T_i^2 s} \tag{36}$$

Thus we have

$$est\left[\frac{\partial y}{\partial \rho}(\rho)\right] = \left[est\left[\frac{\partial y}{\partial K}(\rho)\right],\ est\left[\frac{\partial y}{\partial T_i}(\rho)\right]\right]^T \tag{37}$$

$$est\left[\frac{\partial y}{\partial K}(\rho)\right] = \frac{1}{C(\rho)}\frac{\partial C}{\partial K}(\rho) y^2(\rho) = \frac{1}{K} y^2(\rho) \tag{38}$$

$$\begin{aligned} est\left[\frac{\partial y}{\partial T_i}(\rho)\right] = {} & \frac{\left(f T_i^2 s^2 + N_d s\right)\left(N_d^2 f s\right)}{\left((1+N_d) f T_i^2 s^2 + (f+N_d) T_i s + N_d\right)\left(f^2 T_i^2 s^2 + 2 N_d f T_i s + N_d^2\right)} y^2 - \\ & - \frac{f T_i s + N_d}{(1+N_d) f T_i^3 s^2 + (f+N) T_i^2 s + N_d T_i} y^2(\rho) \end{aligned} \tag{39}$$

and

$$est\left[\frac{\partial u}{\partial \rho}(\rho)\right] = \left[est\left[\frac{\partial u}{\partial K}(\rho)\right],\ est\left[\frac{\partial u}{\partial T_i}(\rho)\right]\right]^T \tag{40}$$

$$est\left[\frac{\partial u}{\partial K}(\rho)\right] = \frac{1}{C(\rho)}\frac{\partial C}{\partial K}(\rho) u^2(\rho) = \frac{1}{K} u^2(\rho) \tag{41}$$

$$\begin{aligned} est\left[\frac{\partial u}{\partial T_i}(\rho)\right] = {} & \frac{\left(f T_i^2 s^2 + N_d s\right)\left(N_d^2 f s\right)}{\left((1+N_d) f T_i^2 s^2 + (f+N_d) T_i s + N_d\right)\left(f^2 T_i^2 s^2 + 2 N_d f T_i s + N_d^2\right)} u^2 - \\ & - \frac{f T_i s + N_d}{(1+N_d) f T_i^3 s^2 + (f+N) T_i^2 s + N_d T_i} u^2(\rho) \end{aligned} \tag{42}$$

So, by filtering data obtained from real experiments with the closed loop system through above defined filters, we get an estimate of the cost function gradient (28) and we can evaluate new controller parameters using (29).

3.3 Tuning of the remaining parameters of the 2DOF PID controller

For the 2DOF PID controller, we have

$$C_r(\rho) = K\left[b + \frac{1}{T_i s} + c\frac{fT_i s}{\frac{fT_i}{N_d}s + 1}\right] \tag{43}$$

$$C_y(\rho) = K\left[1 + \frac{1}{T_i s} + \frac{fT_i s}{\frac{fT_i}{N_d}s + 1}\right], \tag{44}$$

where f, N_d, K and T_i are fixed. The parameter vector ρ is thus $\rho = [b, c]^T$. But the parameter c is mostly set to zero, which means that the derivative action is calculated on y and not on the control error to avoid excessive control action at setpoint change. Vector ρ then degrades to a scalar and we get

$$\frac{1}{C_r(\rho)} = \frac{T_i s}{KbT_i s + K} \tag{45}$$

$$\frac{\partial C_r}{\partial b}(\rho) = K \tag{46}$$

$$\frac{\partial C_y}{\partial b}(\rho) = 0 \tag{47}$$

Therefore we get from (17) and (26)

$$est\left[\frac{\partial y}{\partial b}(\rho)\right] = \frac{T_i s}{bT_i s + 1} y^1(\rho) \tag{48}$$

$$est\left[\frac{\partial u}{\partial b}(\rho)\right] = \frac{T_i s}{bT_i s + 1} u^1(\rho) \tag{49}$$

These estimates can be used in (28) to compute the gradient of the criterion upon which the computation of an updated controller is based (29).

4 DESIGN CHOICES

This section is practice-oriented, it deals with practical design choices to be made to fulfil common control design requests such as minimizing the settling time, minimizing overshoot, minimizing the control effort, etc. These choices involve time-weighting of obtained signals to minimize overshoot and/or frequency weighting to emphasize or suppress specific frequency bands. Some issues regarding convergence rate of the iterative algorithm are also discussed.

4.1 Frequency weighting

The filters L_y and L_u can be used to shape the sensitivity function of the tuned closed loop. If we use a filter L_y, which magnifies some spectrum of input signal, then it will result in attenuation of the same spectrum in the closed loop system. This means that if we used a filter L_y with derivative character, then the tuning process would converge to a control loop, which is not sensitive to high frequencies (noise) and whose response to setpoint change tends to non-oscillatory behavior. The derivative filter L_u must often be incorporated to ensure smooth control action or to exclude non-zero steady-state input from the cost function in the case we are controlling non-integrating process. Detailed illustration of use of these filters can be found in [4].

4.2 Time weighting

Undesirable overshoot is often obtained when using the criterion in the form of (4) along with desired system response $y^D = r$, because too much emphasis is put on the transient phase. An easy way how to shorten the settling time with respect to acceptable overshoot is to introduce some non-negative time weighting factors in the criterion:

$$J(\rho) = \frac{1}{2N} E\left[\sum_{t=1}^{N}(w_t^y L_y \tilde{y}_t(\rho))^2 + \lambda \sum_{t=1}^{N}(w_t^u L_u u_t(\rho))^2\right]. \tag{50}$$

The simplest possibility how to choose the weighting factor w_t^y is to set it to zero during the transient period and to one afterwards. Combined with w_t^u set permanently to one, it provides acceptable overshoot in the resulting closed loop system. Figure 4 shows the difference between the initial and the final step response for IFT with time weighting and without it.

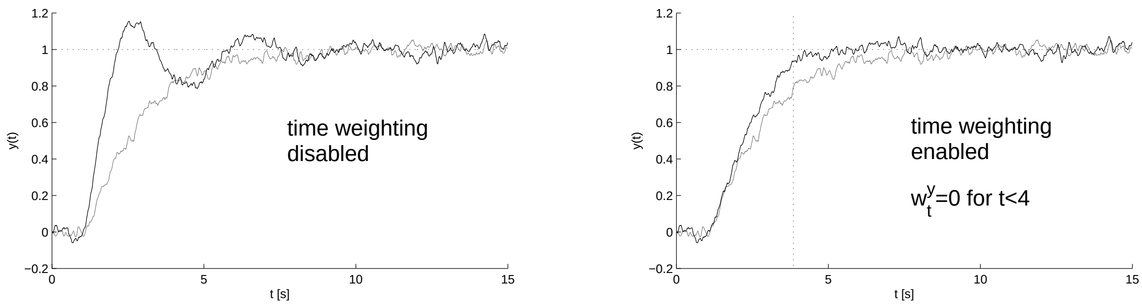

Figure 4 - Illustration of time weighting influence (gray - initial controller)

4.3 Convergence rate

The speed of convergence is influenced mainly by two factors: non-negative scalars γ_i and positive definite matrices R_i involved in (29).

Parameter γ_i determines the step size in the search direction. Its values during the iterative tuning process should satisfy $\sum_{i=1}^{\infty} \gamma_i = \infty$ and $\sum_{i=1}^{\infty} \gamma_i^2 < \infty$ [1]. The smaller the step size is, the slower the algorithm converges but the smaller risk of making an incorrect step we undertake.

There are many possible choices for the matrices R_i. An identity matrix gives pure negative gradient direction, but our experience shows that choice of

$$R_i = \frac{1}{N}\sum_{t=1}^{N}\left(L_y est\left[\frac{\partial y_t}{\partial \rho}(\rho)\right] L_y est\left[\frac{\partial y_t}{\partial \rho}(\rho)\right]^T + \lambda L_u est\left[\frac{\partial u_t}{\partial \rho}(\rho)\right] L_u est\left[\frac{\partial u_t}{\partial \rho}(\rho)\right]^T\right) \tag{51}$$

results in much faster convergence.

4.4 Controlled system properties

Since the only requirement of the IFT method is a stable closed loop at the beginning of the tuning process, there is no limitation for structure or order of the controlled system. It is applicable to a wide variety of processes, including integrating processes, systems with complex poles, non-minimum phase systems, nonlinear systems in the vicinity of the working point and even systems with non-smooth nonlinearity (dead zone, backlash, ...).

5 LIBRARY BLOCK FOR SIMULINK

We have created a Simulink function block which includes standard 2DOF PID controller and autotuner based on the described IFT methodology. It is a part of the RexLib [6] function block library. Schemes designed using blocks from this library can be used not only for simulation purposes in the Simulink environment, but they are also ready for downloading to any open device based on the C platform.

The IFT PID controller block is meant to be used in control schemes like the one shown in Figure 5. When the TUNE input is set to zero, the block works as a standard 2DOF PID

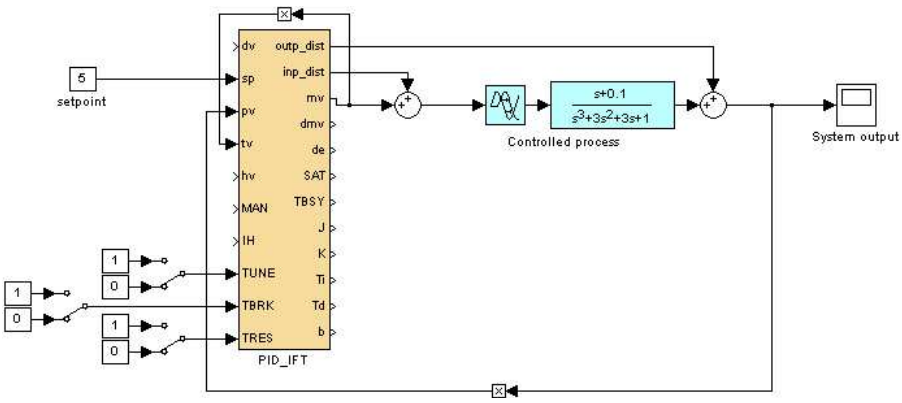

Figure 5 - Typical control loop with the IFT PID controller

controller of the form (30) with respect to (31). One can of course set the controller parameters manually. These will serve as the initial conditions. Setting the TUNE input to one means starting the IFT based autotuning procedure. Then the experiments described in section 3.1 are performed. All the design specifications affecting the controller update during the iterative process (frequency weighting, time weighting, ...) have been implemented in our block. Naturally, they must be formulated before the tuning procedure is started. The input TBRK serves for aborting the autotuning operation, while the TRES input re-sets the initial controller parameters. Anytime we need to review the actual controller parameters, we can do so by reading the appropriate outputs of the block.

6 THE TWO TANK EXPERIMENT

The IFT scheme has been applied to the optimal tuning of 2DOF PID controller operating on the coupled tanks model with the following design choices: sample time $0.1s$, $N_d = 4$, $f = 0.25$, step size $\gamma_i = 0.6$, R_i defined by (51), control weighting $\lambda = 0.1$, $y^D = r$, $w_t^u = 1$, $w_t^y = 0$ for $t < 120s$, $L_y = 1$ and $L_u = \frac{10s}{10s+1}$. The two tanks model is a nonlinear system, therefore the obtained controller parameters are optimal only in the vicinity of the working point. The level of the water surface in the second tank varies between 1.5 and 2.6 in our experiment.

Figure 6 shows the system output during the whole iterative process. Firstly the 1DOF

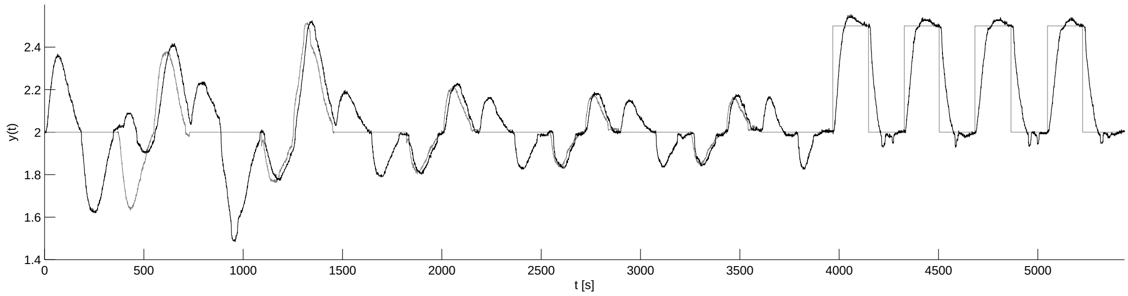

Figure 6 - System output during the whole iterative process (gray - reference signal)

controller is tuned to reject step input disturbance (5 iterations) and secondly the setpoint change response is optimized (3 iterations). The controller parameter change and related criterion minimization can be found in Figure 7 and 8. We can see that in just a few iterations the closed

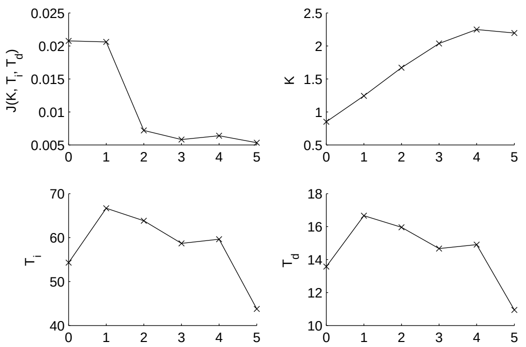

Figure 7 - Cost function and controller parameters after n-th iteration - 1DOF controller

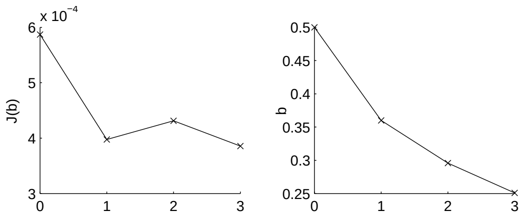

Figure 8 - Cost function and controller parameters after n-th iteration - 2DOF controller

loop behavior significantly improved. Even if not all the steps made by the tuning algorithm in the parameter space lead to better performance, the IFT algorithm converges to a local minimum in general. The wrong decisions may be caused by too big step size or some disturbances acting on the system during the gradient estimation.

The system outputs with the initial and the final controller can be compared in Figure 9. We can see that the improvement is magnificent, especially in the input disturbance rejection framework.

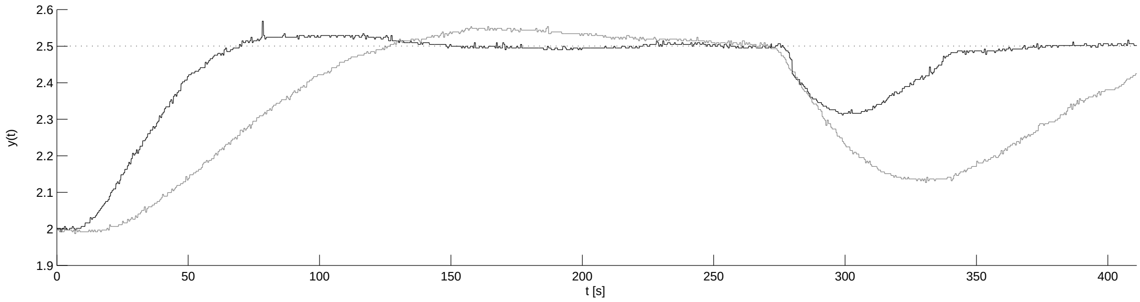

Figure 9 - Comparison of the system output with the initial (gray) and the final controller

7 FEEDFORWARD COMPENSATOR TUNING

In some cases when the disturbance is measurable, the IFT approach can also be used for feedforward compensator tuning. We will illustrate it on a simple example. The structure of the control loop is given in Figure 10.

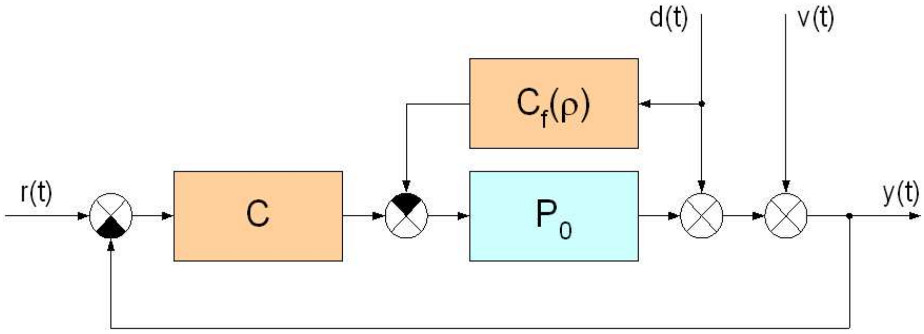

Figure 10 - Block diagram of the closed loop system with measurable disturbance

Output of the system represented in Figure 10 is given by

$$y_t(\rho) = \frac{CP_0}{1 + CP_0} r_t + \frac{1 - C_f(\rho)P_0}{1 + CP_0} d_t + \frac{1}{1 - CP_0} v_t \tag{52}$$

Theoretically, it would be optimal to set the feedforward filter $C_f(\rho)$ to P_0^{-1}, so the disturbance would be completely rejected. We will iteratively minimize the control criterion, which is

$$J(\rho) = \frac{1}{2N} \sum_{t=1}^{N} \tilde{y}_t(\rho)^2, \tag{53}$$

where $\tilde{y}_t(\rho) = y_t(\rho) - y_t^D$, y_t^D is the desired system response again. The gradient of the cost function is given by

$$\frac{\partial J}{\partial \rho}(\rho) = \frac{1}{N} \sum_{t=1}^{N} \tilde{y}(\rho) \frac{\partial y}{\partial \rho}(\rho) \tag{54}$$

and we have the partial derivative of the system output with respect to the controller parameters:

$$\frac{\partial y}{\partial \rho}(\rho) = -\frac{1}{C}\frac{\partial C_f}{\partial \rho}(\rho)\frac{CP_0}{1+CP_0}d. \tag{55}$$

This means we can estimate the gradient from two closed loop experiments:

$$r_i^1 = 0; \qquad d_i^1 = d \tag{56}$$

$$r_i^2 = d; \qquad d_i^2 = 0 \tag{57}$$

Even if these experiments may seem unrealizable because of the requirement of zero disturbance during the second experiment, there are some systems in industry, where these two experiments can be successfully performed. The disturbance d must, of course, be much greater than the noise v.

Then we can evaluate new controller parameters from

$$\rho_{i+1} = \rho_i - \gamma_i R_i^{-1} est\left[\frac{\partial J}{\partial \rho}(\rho_i)\right], \tag{58}$$

where R_i is defined for example by

$$R_i = \frac{1}{N}\sum_{t=1}^{N} est\left[\frac{\partial y_t}{\partial \rho}(\rho_i)\right] est\left[\frac{\partial y_t}{\partial \rho}(\rho_i)\right]^T \tag{59}$$

All the modifications described in previous sections, such as time or frequency weighting, can easily be introduced in the criterion in this case as well, we omitted it just to simplify the explanation.

8 CONCLUSIONS

In this article, we have described an iterative control optimization method. The greatest advantage of the described approach is that the gradient of the design criterion is computed from measured closed loop data. This means that it is not model-based. The algorithm converges to a local minimum of the cost function under the assumption of a stable initial control loop.

It is an easy and straightforward method. The criterion function is clearly expressed and allows us to take into account common control design requirements, such as minimizing the settling time, minimizing overshoot, minimizing control effort or suppressing some frequency regions.

This algorithm is well suited for PID loops, but it is by no means limited to the tuning of such simple controller. Optimizing the performance of a PID loop is still an up-to-date topic in industry, because more advanced control techniques (for example predictive control) rely on good performance of the primary loops, which typically include the PID controllers.

Function block implementing this methodology was created. It was tested in the Simulink environment and also on a real system. Results of these experiments show the promising capabilities of this optimization technique. On the other hand, the IFT method typically requires more data than other available model-free methods for tuning of PID controllers [7].

References

[1] HJALMARSSON, H.; GEVERS, M.; GUNNARSSON, S.; LEQUIN, O. Iterative feedback tuning: theory and applications. *IEEE Control Systems Magazine*, 1998. Vol. 18, No. 4, pages 26–41. ISSN 0272-1708

[2] ÅSTRÖM, K.J.; WITTENMARK, B. *Adaptive Control*. Amsterdam : Addison-Wesley, 1995. 590 pages. ISBN 0-201-55866-1

[3] HJALMARSSON, H.; GUNNARSSON, S.; GEVERS, M. A convergent iterative restricted complexity control design scheme. In: *Proceedings of the 33rd IEEE Conference on Decision and Control*, Orlando, Florida, 1994. Pages 1735–1740. ISBN 0-78031-968-0

[4] HJALMARSSON, H.; GUNNARSSON, S.; GEVERS, M. Model-free tuning of a robust regulator for a flexible transmission system. *European Journal of Control*, 1995. No. 1(2), pages 148–156. ISSN 1435-5671

[5] ÅSTRÖM, K.J.; HÄGGLUND, T. *PID Controllers: Theory, Design, and Tuning*. Instrument Society of America, 1995. 351 pages. ISBN 1-55617-516-7

[6] BALDA, P.; SCHLEGEL, M.; ŠTĚTINA, M. The new REX control system for design and simulation in Matlab/Simulink environment. *Automatizace*, 2003. Vol. 46, No.2, pages 100–103. ISSN 0005-125X.

[7] SCHLEGEL M., BALDA P., ŠTĚTINA M. Robust PID autotuner - method of moments. *Automatizace*, 2003. Vol. 46, No. 4., pages 242–246. ISSN 0005-125X