Part II. PID controller tuning using the multiple integration method

5. Introduction to PID control

PID controllers have been in use for a long time. The first (pneumatic) PI controllers came from the Foxboro and Taylor companies in 1934-1935. Within a few years, Taylor developed its first PID controller, the "Fulscope 100" (Babb, 1990; Bennett, 1993; Blickley, 1990). That PID controller performed excellently on several types of processes (especially temperature processes), but the problem which arose was how to tune such a controller. Without any guidelines for tuning, there were no prospects for selling the product. At that time, tuning of the PI controller required a relatively skilled engineer. The problem was significantly increased by the addition of the derivative (D) part. This led Ziegler and Nichols from the Taylor company to discover the famous "Ziegler-Nichols" tuning rules (Ziegler and Nichols, 1942). Those rules were derived in quite a heuristic way using the laboratory multicapacity system and a "Fulscope 100" PID controller (Blickley, 1990).

The Ziegler-Nichols tuning rules were the very first tuning rules for PID controllers, and it is surprising that they are still widely used today. The reason for their popularity lies in their simplicity and efficiency. This is why so many different tuning rules have been developed which are based on the same tuning procedures (see e.g. Clair, 1995).

After the work of Ziegler and Nichols, a variety of PID tuning methods have been developed. A good survey of these can be found in Gorez (1996) and Haalman (1966). In general, these methods can be divided into two main groups: *direct* and *indirect* tuning methods (Åström and Hägglund, 1993; Gorez, 1996). Direct tuning methods do not use an explicit process model, whilst indirect methods are based on the process model.

The most popular direct tuning methods are based on the measurement of particular points on the process Nyquist curve. The best-known rules are the Ziegler-Nichols tuning rules (Åström and Hägglund, 1995; Hang et al., 1991; Hang and Cao, 1993; Ho et al., 1995, 1996a; Thomas, 1991), and the refined Ziegler-Nichols rules (Åström and

Hägglund, 1995a; Åström and Hägglund, 1995b; Hang et al., 1991; Ho et al., 1995). Several methods based on the detection of one or two points of the process frequency response using relay excitation (Åström and Hägglund, 1984; Atherton, 1996) have become more popular in the last decade. In order to track the ultimate frequency if process dynamics vary, the measuring of more points on the Nyquist curve was proposed by Ho et al., (1996b). Use of the cross-correlation technique so as to detect the process ultimate gain and frequency was proposed in (Hang and Sin, 1991). Repeatedly changing the PI(D) controller parameters until the phase and gain margin are satisfied was proposed in Calcev and Gorez (1995), Denolin and Hanus (1996), Vrančić et al. (1996c), Vrančić and Peng (1994b, 1995a). Some direct tuning methods are also based on pattern recognition methods (Bristol, 1977), genetic algorithms (Wang and Kwok, 1993), and on the optimisation approach (Zhou and Birdwell, 1996).

The most popular indirect tuning methods are based on the process model in the frequency domain. PID controller tuning is usually achieved by placing the closed-loop poles (Persson and Åström, 1993; Shafiei and Shenton, 1994, Vrančić et al., 1993b, 1994), by tuning each term of the PID controller so as to maximise the stability margin according to the specified phase and amplitude margins (Thomas, 1991), or by finding the appropriate PID parameters from the derived IMC controller (Williams and Adeniyi, 1996). Some indirect tuning methods are based on a reduced process model (the first or second order model with pure time delay) (Ho et al., 1993). In this framework, several authors (Besherati and Gawthrop, 1991; Khan and Lehman, 1996; Leva et al., 1994) were particularly interested in accurately obtaining the process time delay. Obtaining an estimation of the process model by using the integration of the process step response was proposed in (Åström and Hägglund, 1995a; Nishikawa et al., 1984; Voda and Landau, 1995). PID controller tuning for such an estimated model is based on the symmetrical optimum technique (Kuhn, 1995, Voda and Landau, 1995), or on the optimal weighted integral of the squared error (Nishikawa et al., 1984).

Each tuning method has its advantages and drawbacks. In general, the more information that can be extracted from the process, the better are the tuning results which can be obtained. However, extracting more process information requires more extensive process identification and computations, with longer and more complicated experiments on the tested plant. Moreover, it has been recognised (Peterka and Åström, 1980) that developing *accurate models* for the process industry and identifying the parameters within them is often not worthwhile; rather the problem is how to determine the *controller* parameters.

On the other hand, tuning rules which require less process information (e.g. detection of the process gain, lag and rise times, or the process ultimate point) generally do not give the "optimal" response, since the available process information is based on few measured parameters. For instance, there are considerably different processes with the same pairs of lag and rise times (or ultimate points); clearly, these processes require different controller parameters. Several drawbacks of the Ziegler-Nichols tuning rules are reported in Åström and Hägglund (1995b), Hang et al. (1991), Hang and Cao (1993), Hang and Sin (1991), Thomas (1991), and Voda and Landau (1995). Some authors have suggested the introduction of the so-called set-point weighting approach (Hang et al., 1991; Hang and Cao, 1993), and the refinements of the Ziegler-Nichols

tuning rules (Åström and Hägglund, 1995a; Åström and Hägglund, 1995b). The improved closed-loop responses have shown that the refined tuning rules gave very good results on several types of processes, but tuning was still based on measurements of the process lag and rise time (or ultimate point), so their applicability to a wide spectrum of processes still remains an open problem.

Our aim was to find a *simple tuning procedure* which could be easily executed by the *real-time auto-tuning algorithm*, and which would give the "*optimal*" *controller settings* for a *large set of process models*. It was decided that the PID controller tuning should be based on the process step response, because it can usually be simply obtained on the real plant. Instead of measuring the lag and rise times (this can be tricky if the process step response is noisy), the process information was extracted from the process step response by using a function which is quite inert to process noise (Nishikawa et al., 1984), and can be simply executed by a real-time digital algorithm: an integration. We chose the multiple integration approach, the technique originally used for the process identification (see e.g. Isermann (1971), Rake (1987), Strejc (1959), and Vrančić et al. (1996b,d)).

It was also decided to meet all the demands of the "magnitude" ("modulus") optimum (MO) tuning goal (Åström and Hägglund, 1995a; Boucher and Tanguy, 1976; Hanus, 1975; Umland and Safiuddin, 1990). The MO technique guarantees fast, stable and non-oscillatory closed-loop behaviour for a large set of industrial processes.

It was found that there is a close relationship between the five areas, obtained from the process open-loop step response (by using the multiple integration approach), and the MO technique. The PID controller parameters were merely calculated from the process step response, regardless of the process order.

6. Derivation of PID controller parameters

The tuning procedure for the PID controller is given for those processes which can be approximated by the following transfer function:

$$G_P(s) = K_{PR} \frac{1 + b_1 s + b_2 s^2 + \cdots + b_m s^m}{1 + a_1 s + a_2 s^2 + \cdots + a_n s^n} e^{-sT_{del}} \quad , \tag{132}$$

where K_{PR} denotes the process steady-state gain, and a_1 to a_n and b_1 to b_m are the corresponding parameters ($m \leq n$) of the process transfer function. A parameter T_{del} represents the process pure time delay.

The PID controller is given by the following transfer function:

$$G_C(s) = \frac{U(s)}{E(s)} = K\left(1 + \frac{1}{sT_i} + \frac{sT_d}{1 + sT_f}\right), \tag{133}$$

where U and E denote the Laplace transforms of the controller output, and the control error ($e=w-y$), respectively. The controller parameters K, T_i, T_d, and T_f denote proportional gain, integral time constant, derivative time constant, and filter time constant, respectively.

The PID controller in a closed-loop configuration with the process is shown in Fig. 74, where d denotes a load disturbance.

A goal of tuning was to find such a controller that makes the closed-loop amplitude (magnitude) frequency response from the set-point to the plant output as flat and as close to unity as possible for a large bandwidth for a given plant and controller structure (see the solid line in Fig. 75).

This technique is called magnitude optimum (MO) (Umland and Safiuddin, 1990), modulus optimum (Åström and Hägglund, 1995a), or Betragsoptimum (Åström and Hägglund, 1995a; Kessler, 1955), and results in a fast and non-oscillatory closed-loop time response for a large class of process models.

The MO tuning goal can be achieved by setting $|G_{CL}(0)|=1$ and $d^n|G_{CL}(j\omega)|/d\omega^n=0$ at $\omega=0$ for as many n as possible (see Åström and Hägglund, (1995a); Boucher and Tanguy, (1976); Hanus, (1975); Umland and Safiuddin, (1990)), where $G_{CL}(s)$ denotes the closed-loop transfer function from w to y.

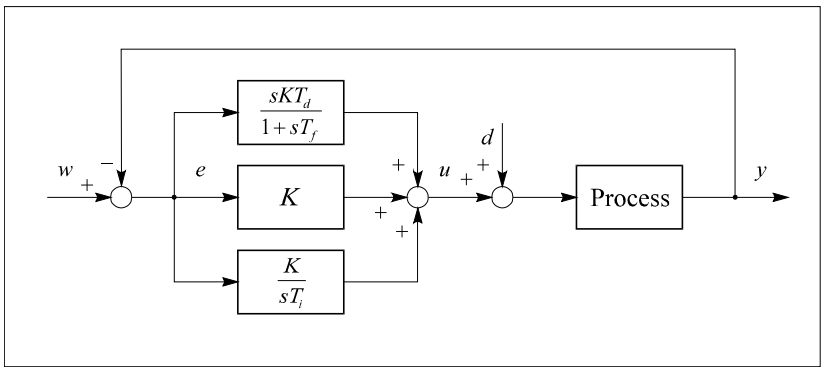

Fig. 74. The closed loop system with PID controller

Figures 75 and 76 illustrate the relation between the *closed-loop* frequency response (Fig. 75), and the *open-loop* frequency response (Fig. 76) (the well-known M and N circles in control theory). To achieve the same tuning goal as given above, the open-loop Nyquist curve of the process and controller should follow the vertical line with the real value -0.5 for the highest frequencies as possible (see solid line in Fig. 76).

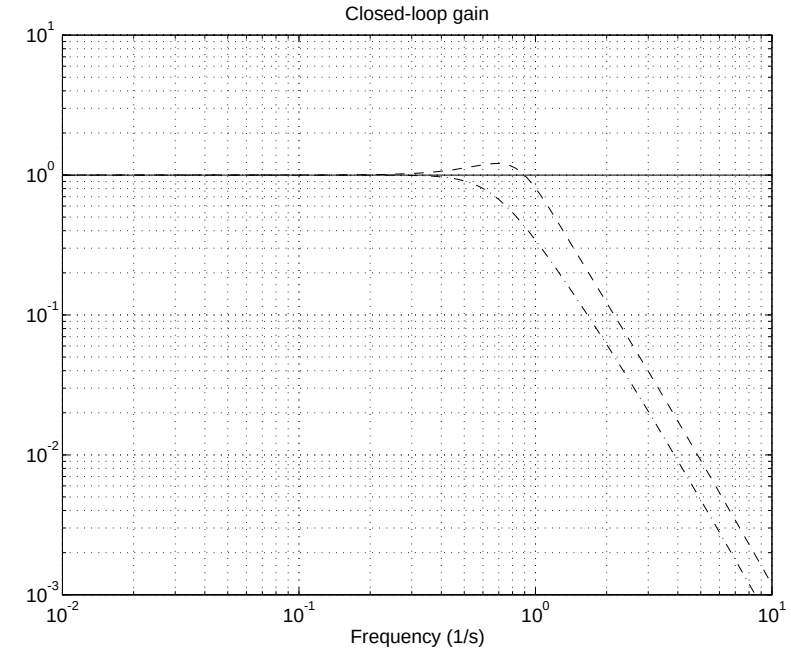

Fig. 75. Bode plot of the closed-loop magnitude frequency-response $|G_P(j\omega)G_C(j\omega)|/|(1+G_P(j\omega)G_C(j\omega))|$

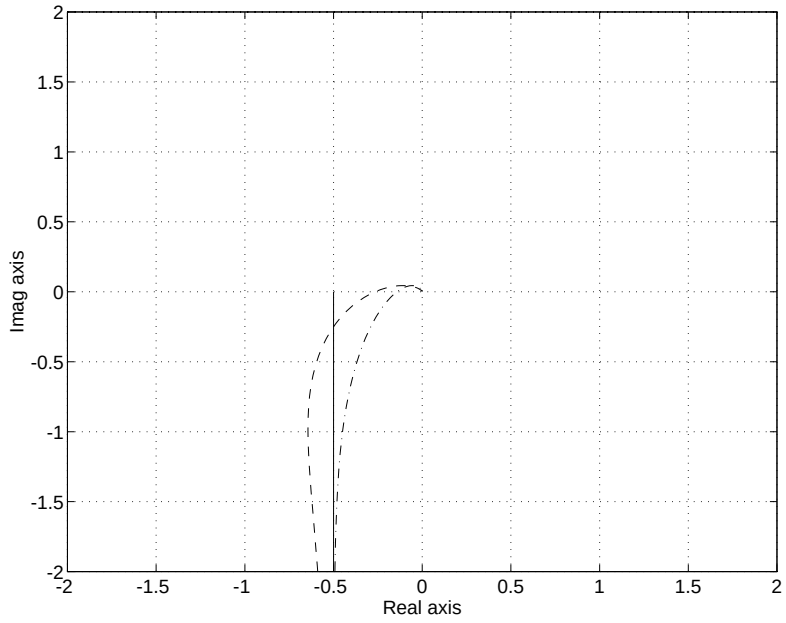

Fig. 76. Nyquist curve of the open-loop frequency response $G_P(j\omega)G_C(j\omega)$

When developing the pure time delay in (132) into the Taylor series:

$$e^{-sT_{del}} = 1 - sT_{del} + \frac{\left(sT_{del}\right)^2}{2!} - \frac{\left(sT_{del}\right)^3}{3!} + \cdots, \tag{134}$$

the open-loop system transfer function can be expressed from (132) and (133) in the following way:

$$G_C(s)G_P(s) = \frac{d_0 + d_1 s + d_2 s^2 + d_3 s^3 + \ldots}{c_0 s + c_1 s^2 + c_2 s^3 + c_3 s^4 + \ldots}, \tag{135}$$

where parameters c_i and d_i in (135) can be calculated by inserting (132), (133), and (134) into (135):

$$\begin{aligned}
c_0 &= T_i \\
c_1 &= a_1 T_i \\
c_2 &= a_2 T_i \\
c_3 &= a_3 T_i \\
c_4 &= a_4 T_i \\
c_5 &= a_5 T_i \\
&\vdots \\
d_0 &= KK_{PR} \\
d_1 &= KK_{PR}\left(b_1 + T_i - T_{del}\right) \\
d_2 &= KK_{PR}\left(b_2 + b_1 T_i + T_d T_i - T_{del}\left(T_i + b_1\right) + \frac{T_{del}^{\,2}}{2}\right) \\
d_3 &= K_{PR}K\left[\begin{array}{l} b_1 T_i T_d + b_2 T_i + b_3 - T_{del}\left(T_i T_d + b_1 T_i + b_2\right) + \\ + \frac{T_{del}^{\,2}}{2}\left(T_i + b_1\right) - \frac{T_{del}^{\,3}}{6} \end{array}\right], \\
d_4 &= K_{PR}K\left[\begin{array}{l} b_2 T_i T_d + b_3 T_i + b_4 - T_{del}\left(b_1 T_i T_d + b_2 T_i + b_3\right) + \\ + \frac{T_{del}^{\,2}}{2}\left(T_i T_d + b_1 T_i + b_2\right) - \frac{T_{del}^{\,3}}{6}\left(T_i + b_1\right) + \frac{T_{del}^{\,4}}{24} \end{array}\right] \\
d_5 &= K_{PR}K\left[\begin{array}{l} b_3 T_i T_d + b_4 T_i + b_5 - T_{del}\left(b_2 T_i T_d + b_3 T_i + b_4\right) + \frac{T_{del}^{\,2}}{2}\left(b_1 T_i T_d + b_2 T_i + b_3\right) - \\ - \frac{T_{del}^{\,3}}{6}\left(T_i T_d + b_1 T_i + b_2\right) + \frac{T_{del}^{\,4}}{24}\left(T_i + b_1\right) - \frac{T_{del}^{\,5}}{120} \end{array}\right] \\
&\vdots
\end{aligned} \tag{136}$$

Note that expressions (136) are valid when T_f=0 (the ideal derivative term). A derivation of the PID controller parameters when T_f≠0 is given in sub-chapter 7.1. However, in all examples and real-time experiments, the derivative filter time constant is set to T_f=T_d/10.

Following the procedure given by Hanus, (1975), the magnitude optimum can be achieved by moving the zeros of the function **Re**{G_P(jω)G_C(jω)}+1/2 toward ω=0. This can be done by fulfilling the following set of equations (Hanus, 1975):

$$\sum_{i=0}^{2n+1}(-1)^i d_i c_{2n+1-i} = \frac{1}{2}\sum_{i=0}^{2n}(-1)^i c_i c_{2n-i} \tag{137}$$

In order to find *three* PID parameters (K, T_i, and T_d), the first three equations (n=0..2) in (137) are to hold (Vrančić and Peng, 1997):

$$d_1c_0 - d_0c_1 + \frac{{c_0}^2}{2} = 0 \tag{138}$$

$$-d_1c_2 - d_3c_0 + d_0c_3 + d_2c_1 - c_0c_2 + \frac{{c_1}^2}{2} = 0 \tag{139}$$

$$d_1c_4 + d_3c_2 + d_5c_0 - d_0c_5 - d_2c_3 - d_4c_1 + c_0c_4 - c_1c_3 + \frac{{c_2}^2}{2} = 0 \tag{140}$$

When inserting (136) into (138) to (140), the following expressions are obtained:

$$K = -\frac{T_i}{2K_{PR}\left(T_i - a_1 + b_1 - T_{del}\right)} \tag{141}$$

$$T_i = -\frac{2KK_{PR}\left(a_1b_2 - a_2b_1 + a_3 - b_3 - T_{del}\left(a_1b_1 - a_2 - b_2\right) + \frac{{T_{del}}^2}{2}\left(a_1 - b_1\right) + \frac{{T_{del}}^3}{6}\right)}{2KK_{PR}\left(T_d\left(a_1 - b_1 + T_{del}\right) + a_1b_1 - a_2 - b_2 - T_{del}\left(a_1 - b_1\right) - \frac{{T_{del}}^2}{2}\right) + {a_1}^2 - 2a_2} \tag{142}$$

$$T_d = -\frac{2KK_{PR}\left(T_i\left(a_1b_3 - a_2b_2 + a_3b_1 - a_4 - b_4\right) + a_1b_4 - a_2b_3 + a_3b_2 - a_4b_1 + a_5 - b_5\right)}{2KK_{PR}T_i\left(a_1b_2 - a_2b_1 + a_3 - b_3 - T_{del}\left(a_1b_1 - a_2 - b_2\right) + \frac{T_{del}^{\ 2}}{2}\left(a_1 - b_1\right) + \frac{T_{del}^{\ 3}}{3!}\right)} -$$

$$- \frac{T_{del}\left(T_i\left(a_1b_2 - a_2b_1 + a_3 - b_3\right) + a_1b_3 - a_2b_2 + a_3b_1 - a_4 - b_4\right)}{2KK_{PR}T_i\left(a_1b_2 - a_2b_1 + a_3 - b_3 - T_{del}\left(a_1b_1 - a_2 - b_2\right) + \frac{T_{del}^{\ 2}}{2}\left(a_1 - b_1\right) + \frac{T_{del}^{\ 3}}{3!}\right)} +$$

$$+ \frac{\frac{T_{del}^{\ 2}}{2}\left(T_i\left(a_1b_1 - a_2 - b_2\right) + a_1b_2 - a_2b_1 + a_3 - b_3\right)}{2KK_{PR}T_i\left(a_1b_2 - a_2b_1 + a_3 - b_3 - T_{del}\left(a_1b_1 - a_2 - b_2\right) + \frac{T_{del}^{\ 2}}{2}\left(a_1 - b_1\right) + \frac{T_{del}^{\ 3}}{3!}\right)} -$$

$$- \frac{\frac{T_{del}^{\ 3}}{3!}\left(T_i\left(a_1 - b_1\right) + a_1b_1 - a_2 - b_2\right) + \frac{T_{del}^{\ 4}}{4!}\left(T_i - a_1 + b_1\right) - \frac{T_{del}^{\ 5}}{5!}}{2KK_{PR}T_i\left(a_1b_2 - a_2b_1 + a_3 - b_3 - T_{del}\left(a_1b_1 - a_2 - b_2\right) + \frac{T_{del}^{\ 2}}{2}\left(a_1 - b_1\right) + 3!\right)} +$$

$$+ \frac{T_i\left(2a_1a_3 - a_2^{\ 2} - 2a_4\right)}{2KK_{PR}T_i\left(a_1b_2 - a_2b_1 + a_3 - b_3 - T_{del}\left(a_1b_1 - a_2 - b_2\right) + \frac{T_{del}^{\ 2}}{2}\left(a_1 - b_1\right) + \frac{T_{del}^{\ 3}}{3!}\right)} \tag{143}$$

When solving (141) to (143), the following PID controller parameters can be expressed by the unknown process parameters:

$$K = \frac{\left[\begin{array}{l} a_1^{\ 3} - a_1^{\ 2}b_1 + a_1b_2 - 2a_1a_2 + a_2b_1 + a_3 - b_3 + \\ + T_{del}\left(a_1^{\ 2} - a_1b_1 - a_2 + b_2\right) + \frac{T_{del}^{\ 2}}{2}\left(a_1 - b_1\right) + \frac{T_{del}^{\ 3}}{6} \end{array}\right]}{2K_{PR}\left[\begin{array}{l} -a_1^{\ 2}b_1 + a_1a_2 + a_1b_1^{\ 2} - a_3 - b_1b_2 + b_3 + T_{del}\left(a_1 - b_1\right)^2 + \\ + T_{del}^{\ 2}\left(a_1 - b_1\right) + \frac{T_{del}^{\ 3}}{3} - T_d\left(a_1 - b_1 + T_{del}\right)^2 \end{array}\right]} \tag{144}$$

$$T_i = \frac{\left[\begin{array}{l} a_1^{\ 3} - a_1^{\ 2}b_1 + a_1b_2 - 2a_1a_2 + a_2b_1 + a_3 - b_3 + \\ + T_{del}\left(a_1^{\ 2} - a_1b_1 - a_2 + b_2\right) + \frac{T_{del}^{\ 2}}{2}\left(a_1 - b_1\right) + \frac{T_{del}^{\ 3}}{6} \end{array}\right]}{a_1^{\ 2} - a_1b_1 - a_2 + b_2 + T_{del}\left(a_1 - b_1\right) + \frac{T_{del}^{\ 2}}{2} - T_d\left(a_1 - b_1 + T_{del}\right)} \tag{145}$$

$$T_d = f\left(a_1 \cdots a_5, b_1 \cdots b_5, T_{del}\right) \tag{146}$$

Note that the explicit result for the derivative time constant is not given. The reason is that expression (146) would otherwise fill up several pages of this thesis.

In order for the method to be applied, the real process must be approximated by the transfer function (132), which requires an explicit identification of the parameters K_{PR}, a_1, a_2, a_3, a_4, a_5, b_1, b_2, b_3, b_4, b_5, and T_{del}. Note that the identified model of the process must be of the same or a higher order than the real process so as to avoid a modelling error which could appear due to the insufficient order of the model. Therefore, calculation of the PID controller parameters (based on expressions (144) to (146)) is frequently impossible in practice.

The *main benefit* of the proposed new tuning method is, however, to avoid this explicit identification by using the concept of multiple integration (Isermann, 1971; Rake, 1987; Strejc, 1959; Vrančić and Juričić, 1995).

Following Strejc, (1959), and considering (132), the following five areas can be expressed by integrating the process open-loop step response (y), after applying the step-change ΔU at the process input:

$$A_1 = y_1(\infty) = K_{PR}\left(a_1 - b_1 + T_{del}\right) \tag{147}$$

$$A_2 = y_2(\infty) = K_{PR}\left[b_2 - a_2 - T_{del}b_1 + \frac{{T_{del}}^2}{2}\right] + A_1 a_1 \tag{148}$$

$$A_3 = y_3(\infty) = K_{PR}\left[a_3 - b_3 + T_{del}b_2 - \frac{{T_{del}}^2 b_1}{2} + \frac{{T_{del}}^3}{3!}\right] + A_2 a_1 - A_1 a_2 \tag{149}$$

$$A_4 = y_4(\infty) = K_{PR}\left[\begin{array}{l} b_4 - a_4 - T_{del}b_3 + \dfrac{{T_{del}}^2 b_2}{2} - \\ - \dfrac{{T_{del}}^3 b_1}{3!} + \dfrac{{T_{del}}^4}{4!} \end{array}\right] + A_3 a_1 - A_2 a_2 + A_1 a_3 \tag{150}$$

$$A_5 = y_5(\infty) = K_{PR}\left[\begin{array}{l} a_5 - b_5 + T_{del}b_4 - \dfrac{{T_{del}}^2 b_3}{2} + \\ + \dfrac{{T_{del}}^3 b_2}{3!} - \dfrac{{T_{del}}^4 b_1}{4!} + \dfrac{{T_{del}}^5}{5!} \end{array}\right] + A_4 a_1 - A_3 a_2 + A_2 a_3 - A_1 a_4 \ , \tag{151}$$

where

$$y_1(t) = \int_0^t \left(K_{PR} - \frac{y(\tau)}{\Delta U} \right) d\tau , \tag{152}$$

$$y_2(t) = \int_0^t \left(A_1 - y_1(\tau) \right) d\tau , \tag{153}$$

$$y_3(t) = \int_0^t \left(A_2 - y_2(\tau) \right) d\tau , \tag{154}$$

$$y_4(t) = \int_0^t \left(A_3 - y_3(\tau) \right) d\tau , \tag{155}$$

$$y_5(t) = \int_0^t \left(A_4 - y_4(\tau) \right) d\tau . \tag{156}$$

In order to clarify the mathematical derivation, graphic representations of the first two areas (A_1 and A_2) are shown in Figures 77 and 78.

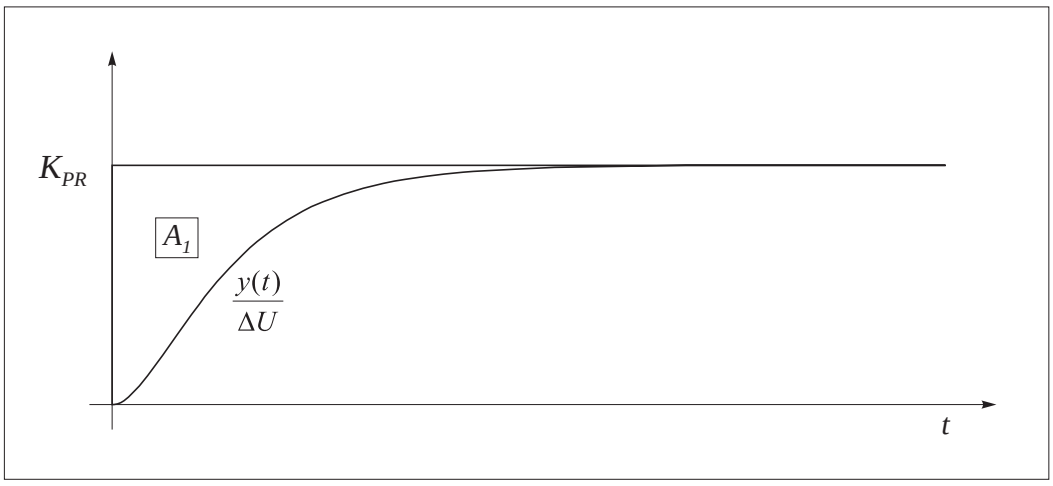

Fig. 77. The graphic representation of area A_1

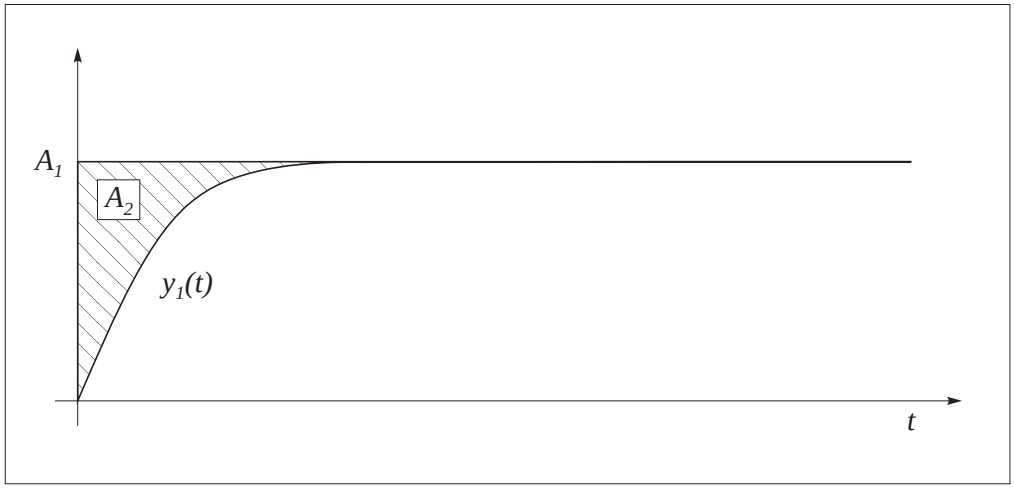

Fig. 78. The graphic representation of area A_2

When inserting the calculated areas (147) to (151), obtained from the process open-loop step response, see e.g. Vrančić (1995b), Isermann (1971) or Rake (1987), into equations (144) to (146), the following result is obtained:

$$K = \frac{A_3}{2\left(A_1 A_2 - A_3 K_{PR} - T_d A_1^{\,2}\right)} \tag{157}$$

$$T_i = \frac{A_3}{A_2 - T_d A_1} \tag{158}$$

$$T_d = \frac{A_3 A_4 - A_2 A_5}{A_3 A_3 - A_1 A_5} \tag{159}$$

Let us define factor α_D as:

$$\alpha_D = \alpha - T_d \frac{A_1^{\,2}}{K_{PR} A_3} \ , \tag{160}$$

where

$$\alpha = \frac{A_1 A_2}{K_{PR} A_3} - 1 \ . \tag{161}$$

When applying (160) to (157) and (158), the proportional gain (K), and the integral time constant (T_i) can be rewritten into:

$$K = \frac{0.5}{K_{PR} \alpha_D} \tag{162}$$

$$T_i = \frac{A_1}{K_{PR}\left(1 + \alpha_D\right)} \tag{163}$$

This form is more appropriate in cases when modifying some of the controller parameters, as will be shown later.

Note that the PI controller parameters can be expressed from (162) and (163) by applying T_d=0 in (160).

Now obviously only the process steady-state gain K_{PR}, and five areas A_1 to A_5 are needed to calculate the unknown PID controller parameters K, T_i, and T_d, by using expressions (159) to (163).

As can be seen from equations (147) to (155), the areas can be calculated from the process open-loop step response by a simple numerical integration, whilst the gain K_{PR} can be determined from the steady-state value of the process step response in the usual way.

The PID controller tuning procedure can therefore proceed as follows:

- measure the process step response,
- find the process steady-state gain K_{PR} and areas A_1, to A_5 (by using numerical integration (summation) from the start to the end of the process step response), and
- calculate the PID controller parameters by using expressions (159) to (163).

The main point is, however, that the PID controller parameters can be calculated exactly (according to the MO criterion) for a wide spectrum of process models (132), merely by measuring the process open-loop step response.

Let us now illustrate the proposed PID controller design in one example.

The following eight-order process model is chosen:

$$G_P(s) = \frac{1}{(1+s)^8} \ . \tag{164}$$

At first, a step-change ΔU=1 is applied to the process input. The process open-loop step response is shown in Fig. 79 above. The final steady-state of the process is $y(\infty)$=1, so the process steady-state gain $K_{PR}=y(\infty)/\Delta U$=1. A function $y_1(t)$ is obtained by numerically integrating the difference K_{PR}-$y(t)/\Delta U$, as given by (147) and (152). The function $y_1(t)$ is shown in Fig. 79 below. The final steady-state $y_1(\infty)$=8, which corresponds to area A_1. Similarly, area A_2 can be obtained by numerically integrating the difference between $A_1=y_1(\infty)$ and $y_1(t)$, as given by (148) and (153). Function $y_2(t)$ is given in Fig. 80 above. The final steady-state value of $y_2(t)$ corresponds to A_2 ($A_2=y_2(\infty)$=36). Similarly, as for $y_1(t)$ and $y_2(t)$ (A_1 and A_2), the remaining functions (y_3 to y_5) and areas (A_3 to A_5) can be calculated. Functions $y_3(t)$ to $y_5(t)$ are shown in Figs. 80 and 81.

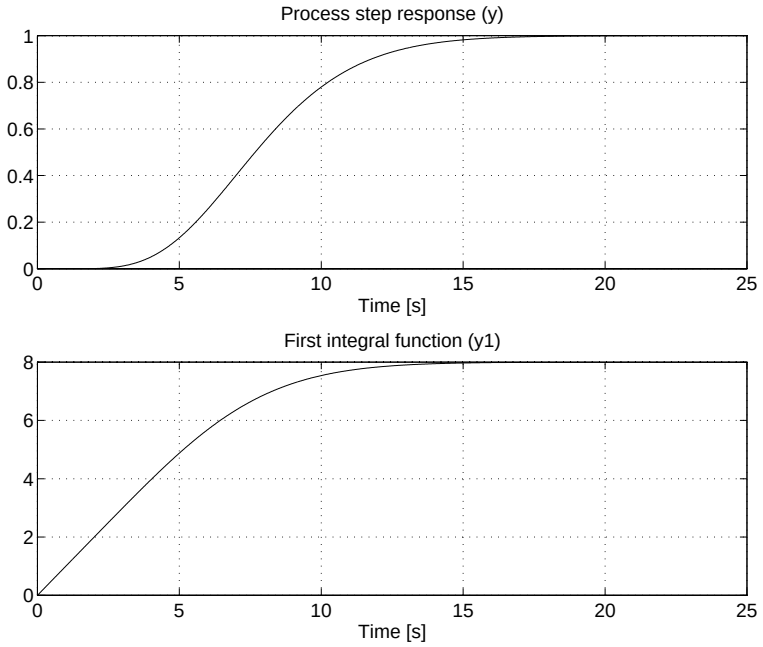

Fig. 79. Process step response (y) (above) and function $y_1(t)$ (below)

The following values of the process steady-state gain and the areas are obtained from the process step-response:

$$K_{PR}=1,\ A_1=8,\ A_2=36,\ A_3=120,\ A_4=330,\ A_5=792\ . \tag{165}$$

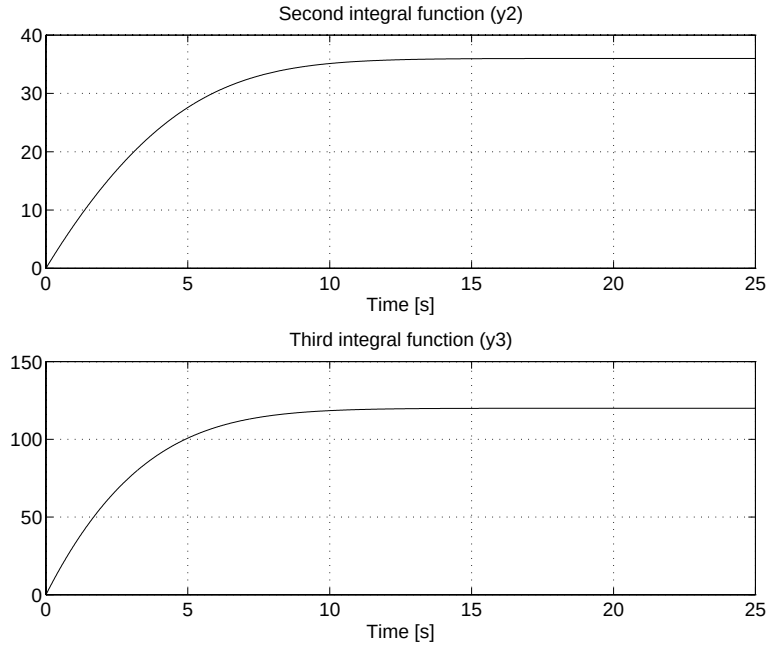

Fig. 80. Function $y_2(t)$ (above) and function $y_3(t)$ (below)

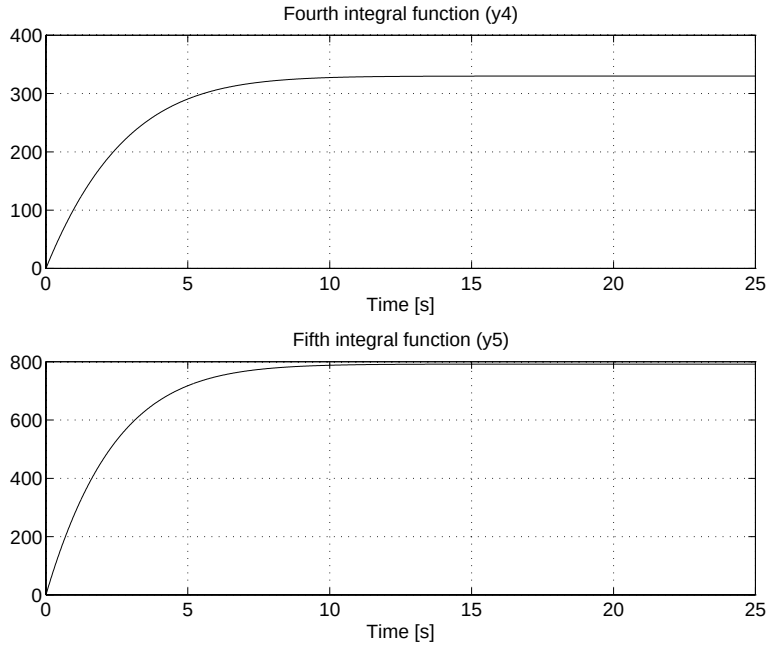

Fig. 81. Function $y_4(t)$ (above) and function $y_5(t)$ (below)

The optimal PID parameters are calculated from (165) by using expressions (159) to (163):

$$K = 0.75, \quad T_i = 4.8s, \quad T_d = 1.375s \ . \tag{166}$$

Figs. 82 and 83 show the closed-loop time responses on the reference change (w=1 at t=0s), and on the load-disturbance (d=1 at t=50s). It is clear that both closed-loop responses are quite acceptable, all according to the chosen tuning goal.

The Nyquist curve of the open-loop frequency response $G_C(j\omega)G_{PR}(j\omega)$ is shown in Fig. 84. It is clear that the Nyquist curve closely follows the vertical line with the real value -0.5 at lower frequencies, as prescribed by the MO tuning.

Fig. 85 shows the closed-loop frequency response. The closed-loop magnitude equals one at lower frequencies, all according to MO.

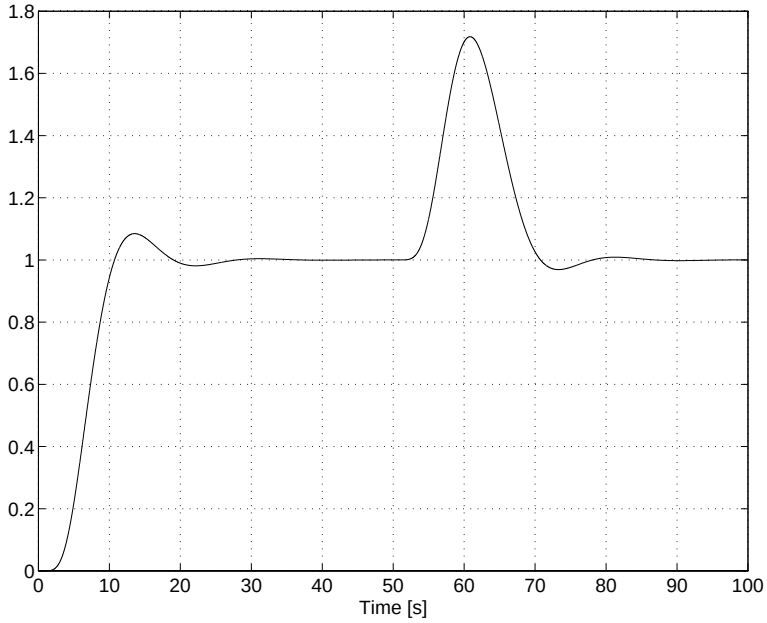

Fig. 82. Process output (y) during the closed-loop experiment

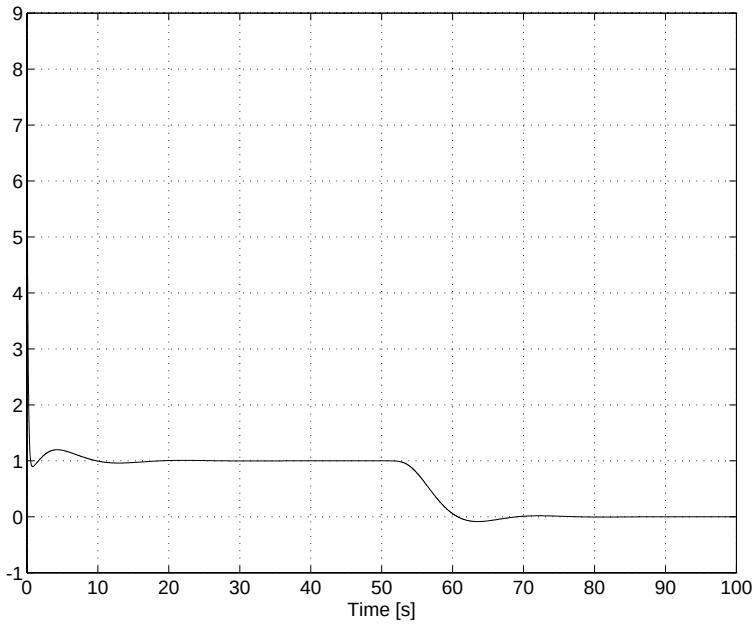

Fig. 83. Process input (u) during the closed-loop experiment

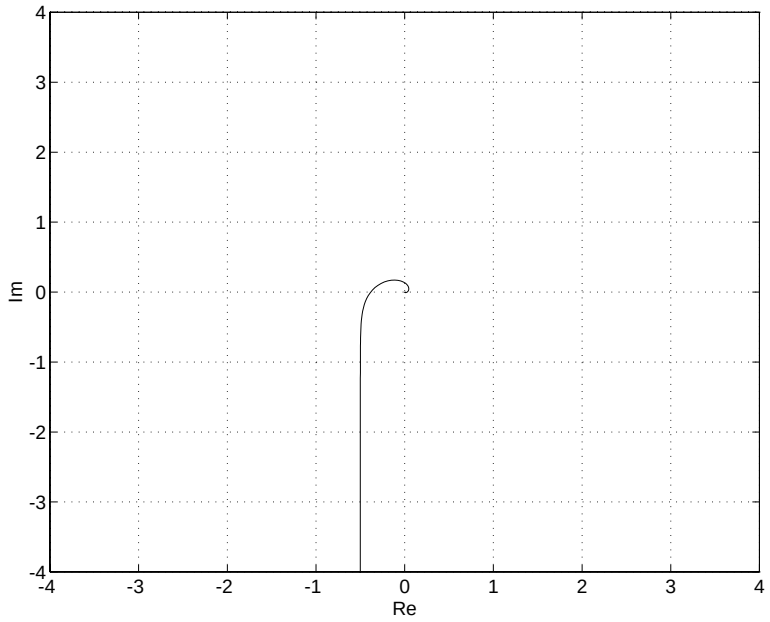

Fig. 84. Nyquist curve of the open-loop frequency response

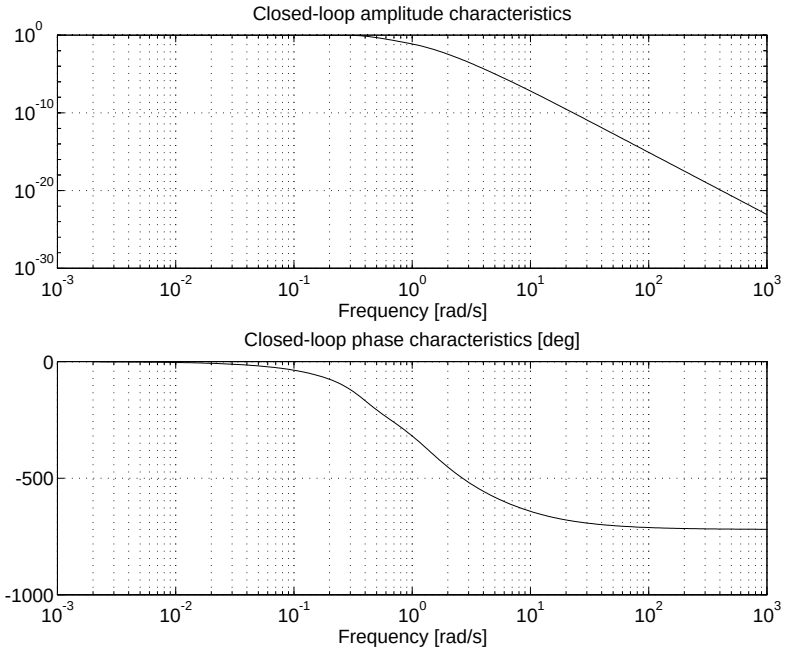

Fig. 85. Bode closed-loop frequency response

7. Some considerations and practical modifications of tuning approach

7.1 Influence of a derivative filter time constant on PID controller settings

The calculation of the PID controller parameters set out in the previous chapter holds for the ideal PID controller without a derivative filter (T_f=0). Let us now investigate a case where a derivative term filter is present, so as to fix the ratio between the derivative and the filter time constant:

$$T_f = \delta T_d \tag{167}$$

Expression (136) therefore becomes

$$
\begin{aligned}
c_0 &= T_i \\
c_1 &= T_i\left[a_1 + \delta T_d\right] \\
c_2 &= T_i\left[a_2 + a_1\delta T_d\right] \\
c_3 &= T_i\left[a_3 + a_2\delta T_d\right] \\
c_4 &= T_i\left[a_4 + a_3\delta T_d\right] \\
c_5 &= T_i\left[a_5 + a_4\delta T_d\right] \\
d_0 &= KK_{PR} \\
d_1 &= KK_{PR}\left[b_1 + T_i + \delta T_d\right] \\
d_2 &= KK_{PR}\left[b_2 + b_1\left(T_i + \delta T_d\right) + T_d T_i\left(1+\delta\right)\right] \\
d_3 &= KK_{PR}\left[b_3 + b_2\left(T_i + \delta T_d\right) + b_1 T_d T_i\left(1+\delta\right)\right] \\
d_4 &= KK_{PR}\left[b_4 + b_3\left(T_i + \delta T_d\right) + b_2 T_d T_i\left(1+\delta\right)\right] \\
d_5 &= KK_{PR}\left[b_5 + b_4\left(T_i + \delta T_d\right) + b_3 T_d T_i\left(1+\delta\right)\right]
\end{aligned}
\tag{168}
$$

Note that in expression (168) the process time delay is fixed to T_{del}=0 in order to simplify the mathematical derivation. However, the final result (169) does not depend on the time delay (T_{del}).

Following similar derivations as for the ideal derivative term, the following expressions are derived:

$$
T_d^{\,4}\delta^3 A_3 + T_d^{\,3}\delta^2 A_1 A_3 - T_d^{\,2}\delta\left(A_5 - A_3 A_2\right) + T_d\left(-A_5 A_1 + A_3^{\,2}\right) + A_5 A_2 - A_4 A_3 = 0 \tag{169}
$$

$$
T_i = \frac{A_3}{A_2 - T_d A_1 - T_d^{\,2}\delta} \tag{170}
$$

$$
K = \frac{T_i}{2\left(A_1 - T_i\right)} \tag{171}
$$

It is clear that when applying δ=0 to expressions (169) to (171), the resulting PID controller parameters are the same as those obtained by using expressions (159) to (163). Generally, by using more terms in expression (169), a more accurate result can be obtained when applying δ>0. Expression (169) is of the fourth order and T_d can still be solved analytically, but the calculation is quite involved. However, in usual applications, when using small ratio δ (e.g. δ≤0.1), the difference between the calculated T_d from (159) and (169) is relatively small. When using greater values of δ, the derivative time constant T_d can usually be quite accurately calculated by using only the last three terms in expression (169):

$$-T_d^{\,2}\delta\left(A_5 - A_3A_2\right) + T_d\left(-A_5A_1 + A_3^{\,2}\right) + A_5A_2 - A_4A_3 = 0 \tag{172}$$

The analytical solution of (172) can be expressed in the following way:

$$T_{d_{1,2}} = \frac{-\left(A_3^{\,2} - A_5A_1\right) + \sqrt{\left(A_3^{\,2} - A_5A_1\right)^2 - 4\delta\left(A_3A_2 - A_5\right)\left(A_5A_2 - A_4A_3\right)}}{2\delta\left(A_3A_2 - A_5\right)} \tag{173}$$

Let us provide an example by using the following third-order process:

$$G_{PR} = \frac{1}{\left(1+s\right)^3} \tag{174}$$

The steady-state gain for the process (174) is K_{PR}=1. The areas A_1 to A_5 can be calculated from expression (147) to (151): A_1=3, A_2=6, A_3=10, A_4=15, A_5=21.

The PID controller parameters, when using the ideal derivative term (δ=0), are calculated from (159), (160), (162), and (163):

$$K = 2.31, \quad T_i = 2.47s, \quad T_d = 0.65s \tag{175}$$

Let us now calculate the PID controller parameters for the situation where the filter time constant is fixed to T_f=0.1T_d (δ=0.1), by using expressions (169) to (171):

$$K = 2.07, \quad T_i = 2.42s, \quad T_d = 0.61s \tag{176}$$

By using the approximate expression (173), and expressions (170), and (171), the following PID controller parameters are obtained:

$$K = 2.08, \quad T_i = 2.42s, \quad T_d = 0.61s \tag{177}$$

It is clear that the difference between the calculated PID controller parameters in (175), (176) and (177) is relatively small. The closed-loop responses, when using process (174), and controllers (175), (176), and (177), are given in Fig. 86. It is clear that only

slight differences exist between the closed-loop responses. Note that all three closed-loop responses were obtained by fixing the filter time constant to T_f=0.1·T_d.

However, the difference becomes more obvious when ratio δ is increased. When fixing the ratio δ=1 for the same process (174), the following PID controller parameters are obtained from (169) to (171):

$$K = 1.31, \quad T_i = 2.17s, \quad T_d = 0.41s \tag{178}$$

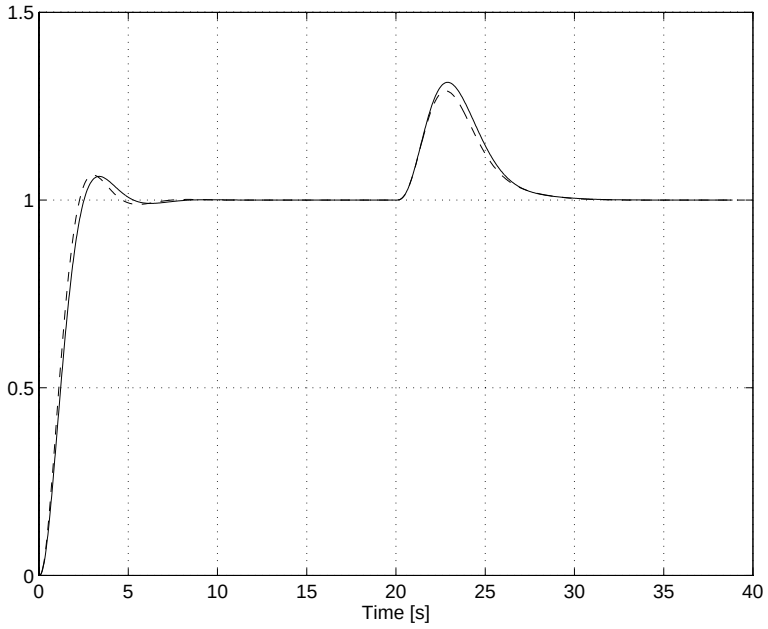

Fig. 86. Closed-loop process output response on the reference change and load disturbance (at t=20s, d=1 (see Fig. 74) is added to the process input) @ T_f=0.1·T_d (δ=0.1); __ controller parameters calculated by using the fourth-order expression (169), and the approximate expression (173), -- controller parameters calculated from basic expression for T_d (159)

When using the approximate formula for T_d (173), the following parameters are obtained:

$$K = 1.46, \quad T_i = 2.23s, \quad T_d = 0.44s \tag{179}$$

Here, the difference between (175), (178), and (179) is more obvious. The difference between the closed-loop responses is shown in Fig. 87. Note that all three process responses were obtained by fixing the filter time constant to $T_f=T_d$.

Therefore, it can be concluded that when using small values for the ratio $\delta=T_f/T_d$ (e.g. $\delta=0.1$ or less), expressions (159) to (163) generally give a very good approximation for the PID controller parameters. However, the exact value of the derivative time constant is obtained by solving the fourth-order expression (169) when $\delta\neq 0$.

In all of the following examples, expressions (159) to (163) are used for calculating the PID controller parameters, and the filter time constant is fixed to $T_f=T_d/10$ ($\delta=0.1$) unless otherwise stated.

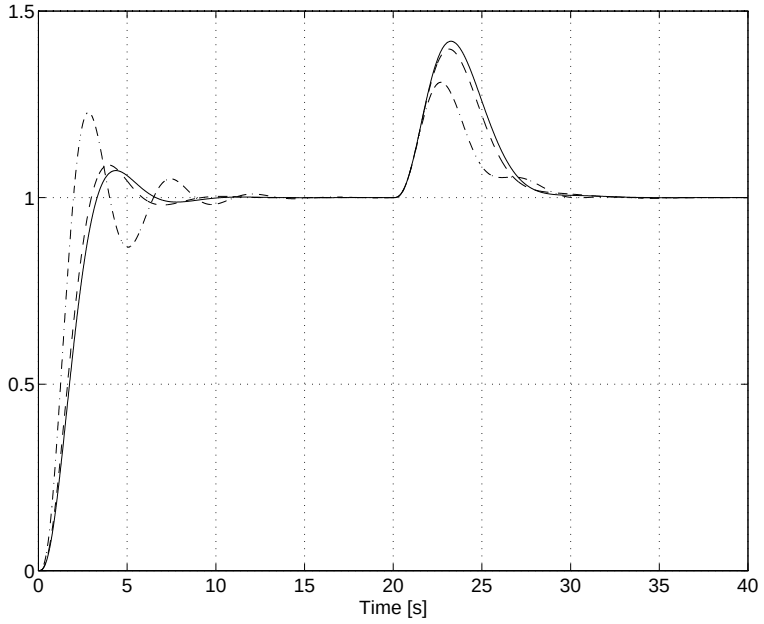

Fig. 87. Closed-loop process output response on the reference change and load disturbance (at t=20s, d=1 (see Fig. 74) is added to the process input) @ $T_f=T_d$ ($\delta=1$); __ controller parameters calculated by using the fourth-order expression (169), -- controller parameters calculated from approximate expression (173), -.- controller parameters calculated from basic expression for T_d (159)

7.2 modified tuning procedure for PID controllers using three areas

In practice, it is relatively difficult to obtain five multiple integrations from the step response without making a significant error. Such an error can occur due to load disturbances, process non-linearities and process noise if the integration time is too long. It is recognised (Rake, 1987) that as the subsequent integration is prone to errors, only low-order models can be obtained by this method. It is advised that the method should only be used for the *identification* of systems up to the third order (Isermann, 1971). If the process model has 3 poles and 2 zeros (without time delay), this implies that 5 subsequent integrations are required in order to explicitly identify the process parameters. Moreover, in our case the calculated areas are not used for the explicit identification of the process parameters, but for the calculation of the controller parameters. Therefore, moderate measurement errors are generally allowed.

However, in our experience, five integrations can only be successfully obtained without making significant errors from relatively undisturbed processes. Three subsequent integrations are much more likely to be successfully obtained in practice. The tuning algorithm for the PI controller, based on the calculation of three integrals (A_1 to A_3), was testeded to the process noise and non-linearity (Vrančić, 1995b). The closed-loop time responses showed quite acceptable results.

Therefore, a tuning procedure for the PID controller will be given which will require the calculation of only three areas (A_1 to A_3) and the process steady-state gain K_{PR}. The areas A_4 and A_5 are explicitly used only in the calculation of the derivative time constant (159). So, an alternative way of calculating T_d will be given which does not require parameters A_4 and A_5. To this end, the ratio $\rho=T_d/T_i$ is fixed to a reasonable value, e.g. $\rho=0.2$ to 0.25 (Vrančić et al., 1997c). Although such a concept is *not optimal* in the sense of MO, it does usually give quite reasonable tuning results.

In such a case, using expressions (160) and (163), the integral time constant can be expressed as:

$$T_i = \frac{A_2 - \sqrt{A_2^{\,2} - 4\rho A_1 A_3}}{2\rho A_1}, \tag{180}$$

where $T_d=\rho T_i$. The controller gain can be re-calculated from (162) and (163) as

$$K = \frac{0.5}{\frac{A_1}{T_i} - K_{PR}} \; . \tag{181}$$

The PID controller tuning procedure can therefore proceed as follows:

- measure a process step response;
- find a process steady-state gain K_{PR} and areas A_1, A_2, and A_3 (by numerical integration (summation) from the start to the end of response) or even areas A_4 and A_5, if the process is relatively undisturbed and mostly linear; and
- calculate PI controller parameters K and T_i (160), (162), and (163) by fixing T_d=0 or PID controller parameters by fixing the ratio $\rho=T_d/T_i$. The ratio ρ=0.2 to 0.25 usually results in quite a reasonable tuning. However, the optimal T_d is explicitly given by (159) if A_4 and A_5 can be successfully obtained.

An illustrative example can be found in Chapter 8.

7.3 modified tuning procedure for 2-degrees-of-freedom PI controllers

It is frequently claimed that a drawback of the MO tuning approach is that the process poles are cancelled by the controller zeros. This may lead to poor attenuation of load disturbances if the cancelled poles are excited by disturbances, and if they are slow compared to the dominant closed-loop poles (Åström and Hägglund, 1995a). However, the MO controller design given in Åström and Hägglund (1995a, page 166) is based on the second-order closed-loop transfer function. In our design procedure, the closed-loop transfer function is of the higher order (according to the process (132) plus controller (133) order), and our results therefore differ from those obtained by Åström and Hägglund (1995a), and Umland and Safiuddin (1990).

However, poorer disturbance rejection can be observed for dominantly first-order processes when using the PI controller, and dominantly first and second-order processes when using the PID controller. In such cases, disturbance rejection can be significantly improved by using a two-degrees-of-freedom PI (PID) controller (e.g. the generalised PID controller). Here, the controller parameters have to be recalculated according to the changed structure of the controller.

Let us derive the controller parameters for the simple two-degrees-of-freedom PI controller shown in Fig. 88 (see e.g. Åström et al., 1993; Åström and Hägglund, 1995b; Hang et al., 1991; Hang and Cao, 1993).

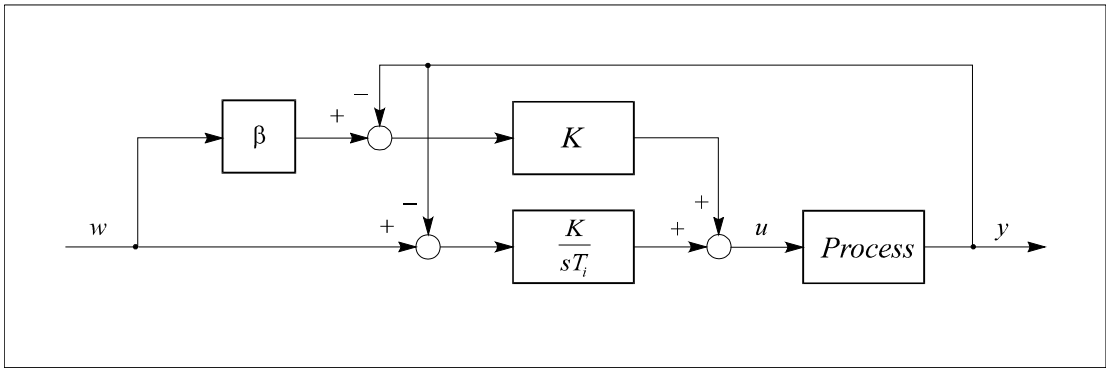

Fig. 88. A two-degrees-of-freedom PI controller.

In order to satisfy the magnitude optimum (MO) criterion, the closed-loop magnitude frequency-response should be as flat and as close to unity as possible for a large bandwidth, as shown by the solid line in Fig. 75.

The system closed-loop transfer function is the following:

$$G_{CL}(s) = \frac{Y(s)}{W(s)} = \frac{G_P(s)K\left(1 + s\beta T_i\right)}{sT_i + G_P(s)K\left(1 + sT_i\right)}\ , \tag{182}$$

where $G_P(s)$ denotes the process transfer function (132). The corresponding *open-loop* transfer function (G_{OL}) can be derived from the following expression:

$$G_{CL}(s) = \frac{G_{OL}(s)}{1 + G_{OL}(s)} \tag{183}$$

The open-loop transfer function can be expressed by solving (182) and (183):

$$G_{OL}(s) = \frac{G_P(s)K\left(1 + s\beta T_i\right)}{sT_i + G_P(s)\left(1 - \beta\right)sKT_i} \tag{184}$$

By using the same method as for the one-degree-of-freedom PID controller (note that for the PI controller, only expressions (138), and (139) are to be satisfied), the following expressions are derived:

$$K^2\left(1-\beta^2\right)\left(K_{PR}{}^2A_3+A_1{}^3-2K_{PR}A_1A_2\right)+2K\left(K_{PR}A_3-A_1A_2\right)+A_3=0 \tag{185}$$

$$T_i=\frac{A_1}{K_{PR}+\frac{1}{2K}+\frac{KK_{PR}{}^2}{2}\left(1-\beta^2\right)} \tag{186}$$

From (185), the controller proportional gain K can be expressed in the following way:

$$K=\frac{\left(A_1A_2-K_{PR}A_3\right)-\sqrt{\left(K_{PR}A_3-A_1A_2\right)^2-\left(1-\beta^2\right)A_3\left(K_{PR}{}^2A_3+A_1{}^3-2K_{PR}A_1A_2\right)}}{\left(1-\beta^2\right)\left(K_{PR}{}^2A_3+A_1{}^3-2K_{PR}A_1A_2\right)}, \tag{187a}$$

if $K_{PR}A_3$-$A_1A_2 < 0$, and

$$K=\frac{\left(A_1A_2-K_{PR}A_3\right)+\sqrt{\left(K_{PR}A_3-A_1A_2\right)^2-\left(1-\beta^2\right)A_3\left(K_{PR}{}^2A_3+A_1{}^3-2K_{PR}A_1A_2\right)}}{\left(1-\beta^2\right)\left(K_{PR}{}^2A_3+A_1{}^3-2K_{PR}A_1A_2\right)}, \tag{187b}$$

if $K_{PR}A_3$-$A_1A_2 > 0$.

In cases when β=1, or $K_{PR}{}^2A_3$+$A_1{}^3$-$2K_{PR}A_1A_2$=0, the proportional gain can be calculated from (185) as:

$$K=\frac{A_3}{2\left(A_1A_2-K_{PR}A_3\right)} \tag{188}$$

Note that expression (188) is the same as expression (162), when using the PI controller (T_d=0).

Two examples were made in order to depict the above results. The first example was made with the following dominantly first-order process (the second process time constant is much shorter than the main time constant):

$$G_P(s)=\frac{1}{(1+s)(1+0.1s)} \tag{189}$$

The following values of the process steady state gain and areas are calculated: K_{PR}=1, A_1=1.1, A_2=1.11, A_3=1.111. The value of expression $A_3+A_1^3-2A_1A_2$=0, so the controller gain is calculated by using expression (188). The integral time constant is given by expression (186). Controller parameters were calculated for different values of parameter β:

β	*K*	T_i *[s]*
1	5.05	1.0009
0.8	5.05	0.5478
0.5	5.05	0.3676
0	5.05	0.3035

The closed-loop responses, when using all four of the PI controllers given above with different parameter β, are shown in Fig. 89. It is clear that disturbance rejection is significantly improved by using smaller values of β down to β=0.5. Further decreasing of factor β does not improve significantly the disturbance rejection, but rather degrades the tracking performance.

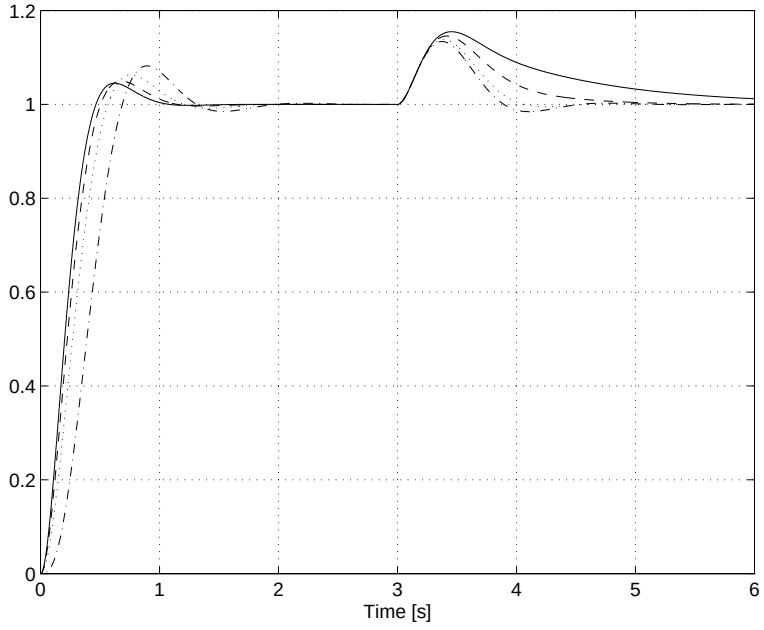

Fig. 89. Process (189) closed-loop responses when using different PI controller parameters and factors β*;* __ β*=1,* -- β*=0.8,* ... β*=0.5,* -.- β*=0.*

The other example was made by using the fifth-order process:

$$G_P(s) = \frac{1}{(1+s)^5} \tag{190}$$

The following values of the process steady state gain and areas are calculated: K_{PR}=1, A_1=5, A_2=15, A_3=35. This time the value of $A_3+A_1^3-2A_1A_2 \neq 0$, so the controller gain is calculated from expression (187) (except for β=1). As in the previous case, the PI controller parameters were calculated for the same values of parameter β:

β	*K*	T_i *[s]*
1	0.438	2.333
0.8	0.447	2.251
0.5	0.457	2.167
0	0.465	2.117

The closed-loop responses when using all four PI controllers with different parameter β are shown in Fig. 90. It is clear that the responses are almost identical. However, the decreased value of parameter β does not improve disturbance rejection significantly, whilst still degrading the tracking performance.

From our experience, based on numerous simulations on different process transfer functions and a real-time implementation of the algorithm to the plastic extruder machine, the most appropriate range of parameter β for achieving good tracking performance and disturbance rejection is:

$$\beta = 0.5 \ldots 0.8 \tag{191}$$

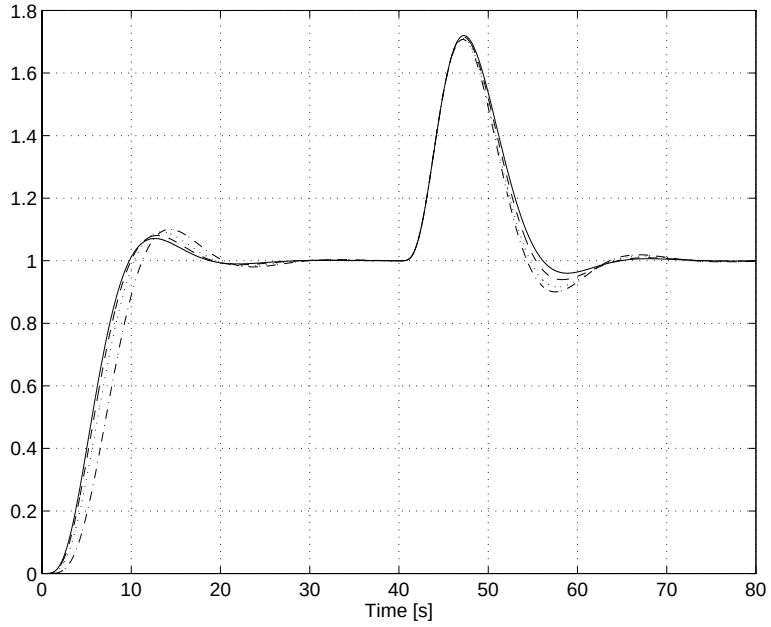

Fig. 90. Process (190) closed-loop responses when using different PI controller parameters and factors β*;* __ β*=1,* -- β*=0.8, ...* β*=0.5,* -.- β*=0.*

7.4 Improving some certain tuning rules by using the new tuning approach

This new tuning method is also suitable for improving the response of certain classical tuning rules (e.g. Ziegler-Nichols, Cohen-Coon, Chien-Hrones-Reswick, etc.). For example, the proportional gain of the PI controller can be obtained by using the Ziegler-Nichols (or any other) settings, whilst the integral time constant can be recalculated from expressions (162) and (163).

As an example, we use the process $G_{PI}(s)$ with T=1 (see Appendix E). The Ziegler-Nichols tuning rules give the following PI parameters: K=0.9 and T_i=3.3. The integral term time constant can be recalculated using the new tuning approach. At first, factor α_D is calculated from (162): $\alpha_D=0.5/(K \cdot K_{PR})=0.555$. The integral time constant T_i is then calculated from (163): T_i=2s/1.555=1.29s. The tuning results, when using the original Ziegler-Nichols tuning rules and the recalculated parameter T_i, are shown in Fig. 91. It is obvious that the recalculated PI controller gives better closed-loop performance.

Note that only area A_1 is used in the recalculation of parameter T_i. Such an approach can also be used for highly disturbed processes.

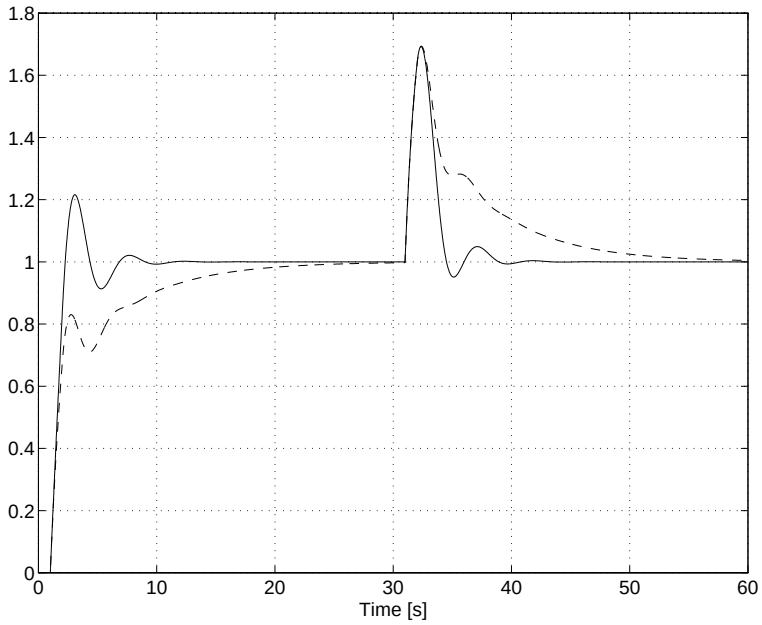

Fig. 91. The process response to the reference change and load disturbance (d=1 @ t=30s); __ K=0.9, T_i=1.29 (recalculated T_i); -- K=0.9, T_i=3.33 (Ziegler-Nichols)

7.5 Process noise and non-linearity

High-frequency process noise does not usually cause significant errors in calculating areas. However, if the noise is quite distinctive and the integration time is too long, fairly large errors can be obtained, as reported in (Vrančić, 1995b). The proposed scheme, by which the time of experiment is divided into the integration period and a determination of the process gain period, is given in Fig. 92. Such an approach can significantly improve the accuracy of the calculated areas (an example can be found in Vrančić, (1995b)).

The new tuning algorithm was also tested on non-linear processes. It was shown that tuning results obtained by using the PI controller are relatively robust to the process non-linearities (Vrančić, 1995b).

However, the higher the process noise, disturbances, and non-linearities, the less reliable is the information that can be obtained from the process step response.

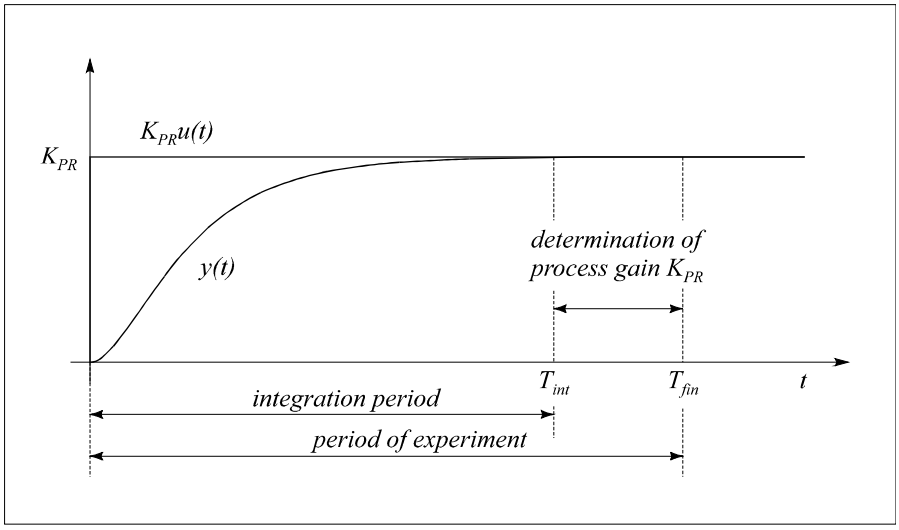

Fig. 92. The proposed approach divides the experimental period into two periods in order to improve the accuracy of the measured areas

7.6 Stability considerations

In spite of the considerably large set of tested process models (see Appendix E), the stability of the obtained PI or PID controller is not a priori guaranteed. A necessary stability condition is given by Hanus, (1975):

$$\frac{K_{PR}K}{T_i} > 0 \ . \tag{192}$$

Inequality (192) is a sufficient condition when using the controller which places *all* the zeros of the function $\mathbf{Re}\{G_P(j\omega)G_C(j\omega)\}+1/2$ toward $\omega=0$ (Hanus, 1975). This can be achieved by using the controller with an appropriately high degree. By using the PI controller, only four zeros can be moved toward $\omega=0$. The PID controller can increase the number of zeros by two, but this is still not a sufficient condition for the closed-loop stability when using higher-order processes.

In fact, the necessary condition is sometimes violated when using the PI (PID) controller on the processes with strong zeros or complex poles placed close to the imaginary axis.

Indeed, in such cases factors α or α_D can be $-1 \le \alpha \le 0$ ($-1 \le \alpha_D \le 0$)[1]. From (162) and (163), it can be seen that $-1 < \alpha_D < 0$ violates expression (192), if $A_1 > 0$. A typical example can be the following process:

$$G_P(s) = \frac{1+s}{(1+2s)(1+0.1s)} \ . \tag{193}$$

The calculated factor is $\alpha = -0.449$ ($\alpha_D = -0.476$). This results in an unstable closed-loop system. The open-loop Nyquist curve when using the calculated PI and PID controller (PI: K=-1.114, T_i=1.996s; PID: K=-1.05, T_i=2.1s, T_d=0.0952s) for the process (193) is shown in Fig. 93. It is obvious that the Nyquist curve follows the prescribed vertical line with the real value -0.5, as required by the MO criterion. However, both Nyquist curves start on the wrong side (from $+j\infty$) and the system is therefore unstable.

Sometimes, stability can be achieved simply by choosing a positive factor α (e.g. by inverting its sign) and recalculating the controller parameters.

As an example, we can choose α=0.2, and α_D=0.1 for the process (193). The modified derivative term time constant can be calculated from (160) as:

$$T_d = \frac{A_3(\alpha - \alpha_D)}{A_1^2} \ , \tag{194}$$

whilst the other two parameters of the PID controller can be calculated directly from (162) and (163) (K=5, T_i=1s, T_d=0.348s). Note that the PI controller parameters can be calculated from (162) and (163) by substituting $\alpha_D = \alpha$ (K=2.5, T_i=0.917s).

The open-loop Nyquist curves when using the modified PI and PID controller parameters are shown in Fig. 94. It is clear that the closed-loop system is now stable. This is also confirmed by the process closed-loop time responses as shown in Fig. 95.

However, such a technique does not work in all cases (Vrančić, 1995b). Therefore, we will pay more attention to such cases in our further research.

[1] If $\alpha < -1$ or $\alpha_D < -1$, a stable PI (PID) controller is obtained with unstable zero (see Vrančić, (1995)).

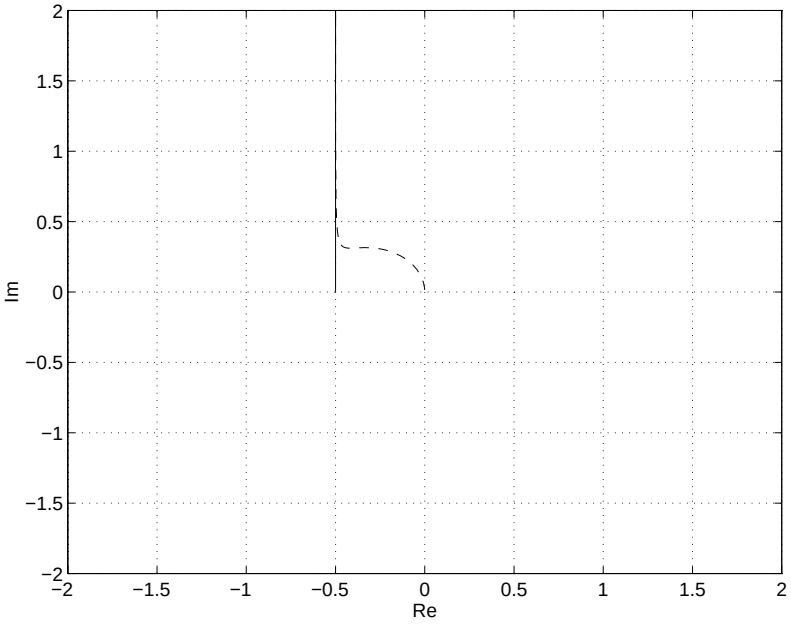

Fig. 93. The open-loop Nyquist curve when using: __ the PID controller, -- the PI controller

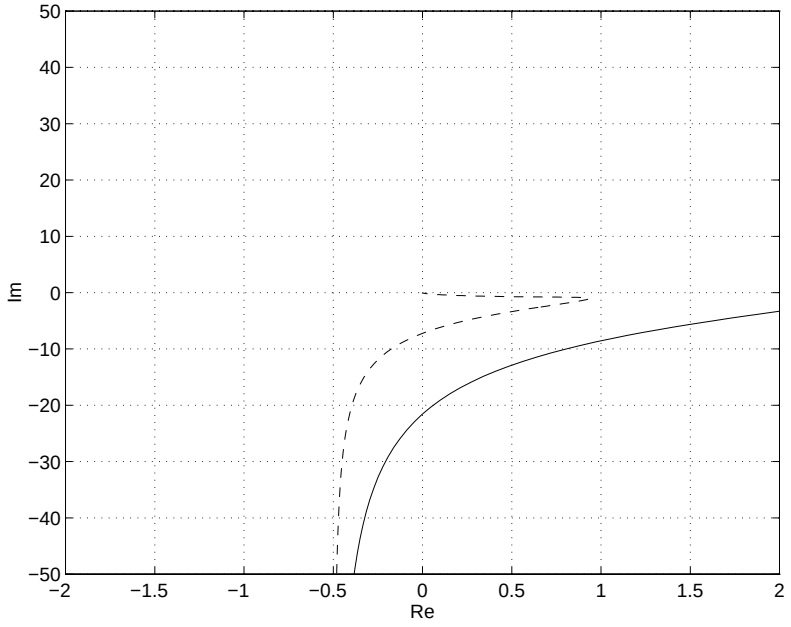

Fig. 94. The open-loop Nyquist curve when using the modified controller parameters for: __ the PID controller, -- the PI controller

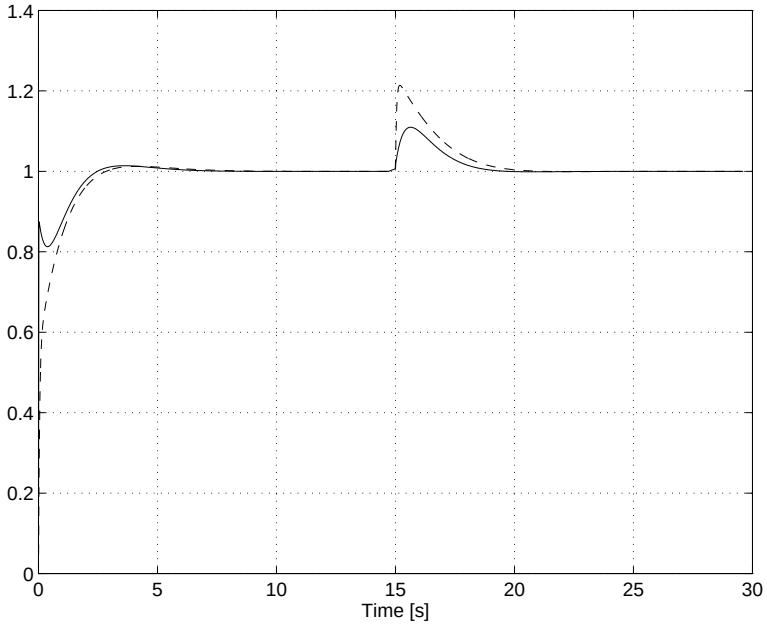

Fig. 95. The process closed-loop time response when using the modified controller parameters for: __ the PID controller, -- the PI controller

8. Simulation examples

A simulation experiment was performed to depict the above results in more detail, during which tracking and control performances were tested.

The following third-order system was chosen:

$$G_P(s) = \frac{1}{(1+s)^3} \ . \tag{195}$$

From the process step response, the following parameters were detected:

$$K_{PR} = 1,\ A_1 = 3,\ A_2 = 6,\ A_3 = 10,\ A_4 = 15,\ A_5 = 21 \ . \tag{196}$$

The PI controller parameters were calculated from (160), (162), and (163):

$$K = 0.625,\ T_i = 1.667 \ . \tag{197}$$

In order to speed-up the closed-loop performance, the PID controller was used.

The optimal T_d was calculated from (159), and the corresponding PID controller parameters were:

$$K = 2.31,\ T_i = 2.467,\ T_d = 0.649 \tag{198}$$

The PID controller parameters, when applying the modified tuning procedure which uses only three areas (see Chapter 7.2), were also calculated by fixing the following ratios: $\rho=T_d/T_i$=0.2, 0.25 and 0.29 (see expressions (180) and (181)). The calculated PID controller parameters were:

ρ	K	T_i	T_d
0.2	1.19	2.113	0.423
0.25	1.87	2.367	0.592
0.29	7.77	2.819	0.817

The process closed-loop responses are shown in Fig. 96. It is clear that the process closed-loop response on the reference change and on disturbance when using the PID controller (198) is quite faster than the response given when using the PI controller (197) (see upper figure). From the lower figure, it can be seen that the process response, when using the PID controller with ρ=0.29, results in quite a higher overshoot than when using all the other controllers. An additional slight increase of ρ would result in a negative K and in an unstable closed-loop response.

In general, the proportional gain K increases upon increasing factor ρ. By testing several process transfer functions and laboratory plants, it was found that ratio ρ=0.2 is a relatively safe choice.

The new tuning method was also compared to the Ziegler-Nichols (ZN), Cohen-Coon (CC) and Chien-Hrones-Reswick[2] (CHR) step-response tuning methods (Åström and Hägglund, 1995b; Haalman, 1966; Hang et al, 1991; Šega, 1991; Zupančič, 1996).

In comparison to these methods, the new method gives time responses with a very small overshoot, without oscillations, and with a better or comparable settling time for a large set of processes (see Appendix E) (Vrančić et al., 1995a), all of which is in accordance with the chosen tuning criterion (MO).

In order to illustrate the tuning results, a comparison of time responses obtained by the mentioned methods for three of the tested processes is given. The chosen processes are G_{P1} with T=1 (process a), G_{P5} with n=5 (process b), and G_{P7} with T=10 (process c) (see Appendix E).

The calculated controller parameters for all four tuning methods, for PI and PID controller (optimal T_d), are shown in Tables 2.2 and 2.3.

In order to test the behaviour of the closed-loop system, a unity step change in reference at the beginning of the experiment and a unity step load disturbance at the process input in the middle of the experiment were chosen.

[2] For tracking with a 20% overshoot.

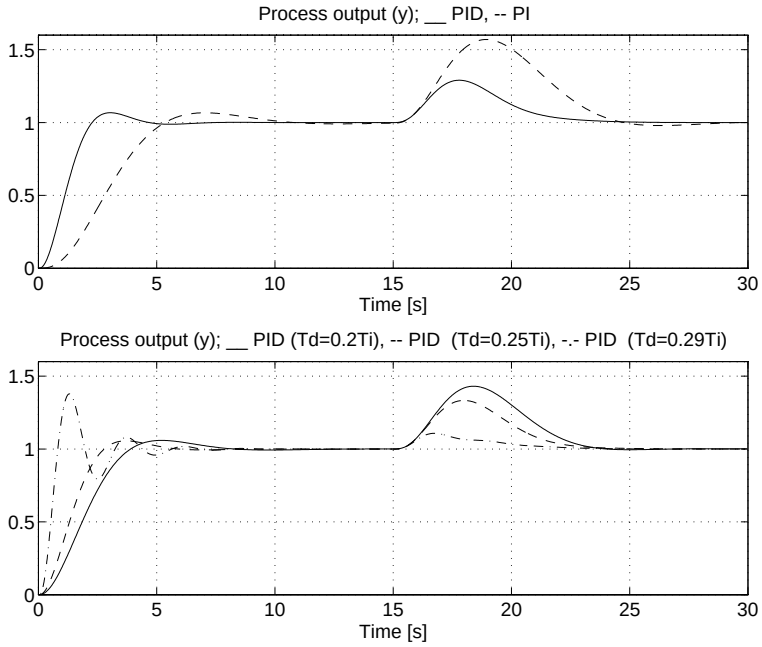

Fig. 96. The closed-loop process time responses (y) when using the PI (--) and the PID (__) controller(upper figure), and three different PID controllers obtained by fixing the ratios $\rho=T_d/T_i=0.2$ (__), 0.25 (--), and 0.29 (-.-) (lower figure). Load disturbance d=1 appears @ t=15s.

The results, obtained by simulation using a MATLAB-SIMULINK programme package, are shown in Figs. 97 and 98. Note that the CC settings gave an unstable closed-loop response for processes b and c (for PI and PID controllers), the CHR settings gave an unstable closed-loop response for process c (for PI and PID controllers), and ZN settings gave an unstable response for process c for PID controllers, and are therefore not shown.

Table 2.2. Calculated PI controller parameters.

Process	***New tuning method***		***ZN***		***CC***		***CHR***	
	K	T_i	***K***	T_i	***K***	T_i	***K***	T_i
a	0.571	1.067	0.9	3.3	0.983	1.138	0.6	1.0
b	0.437	2.33	2.19	6.93	2.28	3.81	1.463	5.12
c	0.088	1.95	0.194	15.44	0.277	2.02	0.129	1.008

Table 2.3a. Calculated PID controller parameters.

Process	***New tuning method***			***ZN***		
	K	T_i	T_d	***K***	T_i	T_d
a	1.03	1.34	0.26	1.2	2	0.5
b	1.08	3.41	0.95	2.93	4.2	1.05
c	0.126	2.62	0.71	0.26	9.36	2.34

Table 2.3b. Calculated PID controller parameters.

Process	***CC***			***CHR***		
	K	T_i	T_d	***K***	T_i	T_d
a	1.58	1.81	0.31	0.95	1.35	0.47
b	3.5	4.44	0.71	2.32	6.91	0.99
c	0.54	5.58	0.92	0.205	1.36	2.2

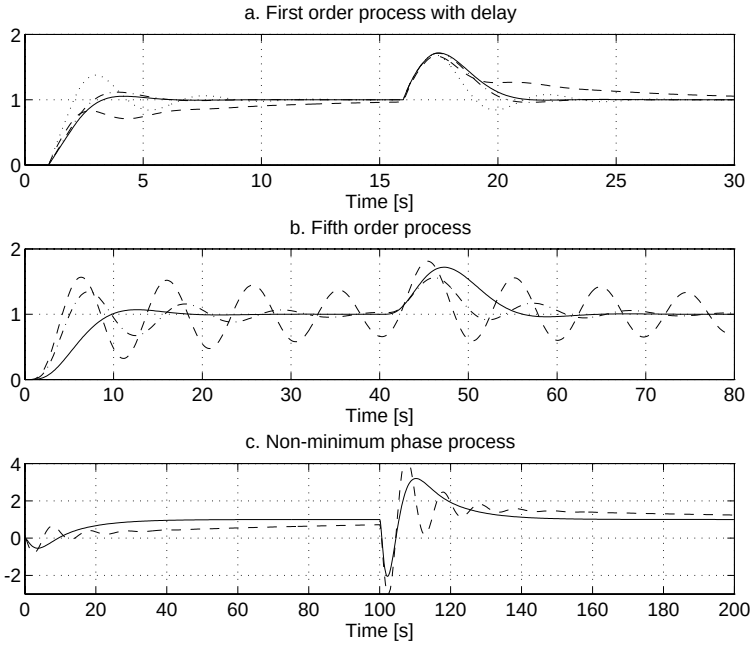

Fig. 97. Process output (y) closed-loop responses for test processes when using PI controller; __ new tuning method, -- ZN, ··· CC, -·- CHR,

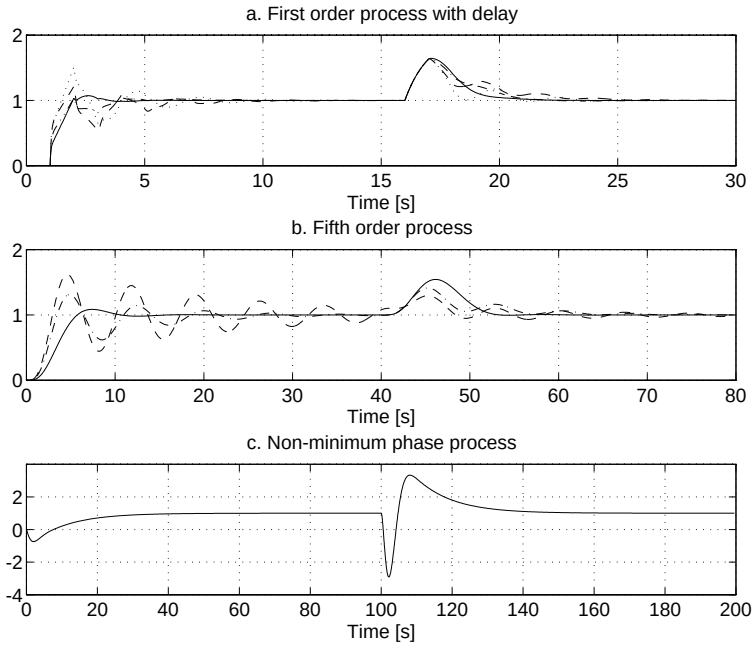

Fig. 98. Process output (y) closed-loop responses for test processes when using PID controller; __ new tuning method, -- ZN, ··· CC, -·- CHR,

9. Experiments with real-time auto-tuning algorithm

9.1 Description of the auto-tuning algorithm

The auto-tuning algorithm, made in the Pascal programme language, using programme unit CONTROL (Vrančić et al., 1993c), has been built up to show the advantages of using the proposed tuning method in the auto-tuning controllers (Đapić, 1997; Vrančić et al., 1997b).

The block scheme of the auto-tuning algorithm is given in Fig. 99.

9.1.1 Inserting parameters

At first, the algorithm requires some parameters for proper initialisation:

- sampling time (T_S),
- amplitude of the step-change at the process input (ΔU),
- maximum allowable open-loop proportional gain K^*K_{PR} (K_{max}), so as to reduce the transfer of the quantisation noise from the A/D converter to the output of the derivative term, and
- approximate process main time constant (T_{main}).

The last parameter (T_{main}) does not have to be accurate. It is generally enough to estimate the range of the value (e.g. 1s, 10s, 100s...).

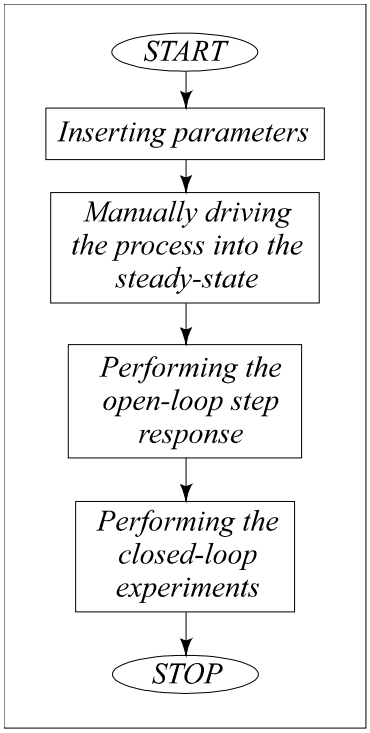

Fig. 99. Block diagram of the auto-tuning algorithm

9.1.2 Manually driving the process into the steady-state

After inserting the parameters, the algorithm switches into the manual mode and the process has to be driven to the desired steady-state. When the process output settles, we can start the next stage of the algorithm, at which the open-loop step-response is performed.

9.1.3 Performing the open-loop step response

At first, a standard deviation (σ_1) and a mean value ($\bar{y}_1$) of the process output signal is measured by using the recursive algorithms, during the period $0<t\leq t_1=T_{main}/4$ (see Fig.

100 and a block-diagram in Fig. 101). Then, at $t=t_1=T_{main}/4$, a step-change ΔU is applied to the process input. After $t=t_1$, five integrals of $y(t)-\bar{y}_1$ are calculated recursively, where $y(t)$ denotes the process open-loop step response. Time instants t_1 to t_n are changed in the following way:

$$t_{i+1}=\begin{cases} t_i+\dfrac{T_{main}}{4} & ;\ t_i < T_{main} \\ 1.25\cdot t_i & ;\ t_i \geq T_{main} \end{cases} . \tag{199}$$

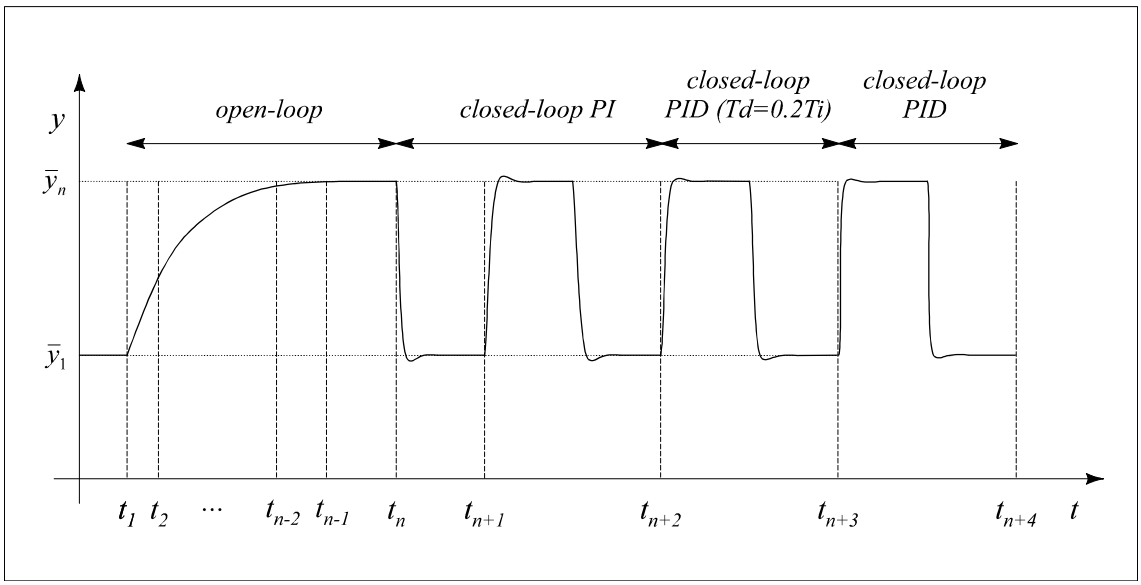

Fig. 100. Process output during the open-loop and the closed-loop experiments performed by the auto-tuning algorithm

In time intervals $t_{i-1} \leq t \leq t_i$ $(i=2...n)$, the process deviation σ_i, and the process mean value $\bar{y}_i$ are recursively calculated. The multiple integrations of the process step response are also recursively calculated from $t=t_1$ and are terminated at $t=t_{n-1}$, when the standard deviation becomes $\sigma_{n-1}\leq 2\cdot\sigma_1$ or when $\sigma_{n-1}\leq\sigma_{max}/40$, where

$$\sigma_{\max} = \max_{k=1\ldots n-1} \sigma_k \ . \tag{200}$$

The steady-state gain of the process is calculated at $t=t_n$ in the following way:

$$K_{PR} = \frac{\bar{y}_n - \bar{y}_1}{\Delta U}, \tag{201}$$

At the same instant, the areas A_1 to A_5 are calculated:

$$A_1 = K_{PR}\left(t_{n-1} - t_1\right) - \frac{I_1}{\Delta U}, \tag{202}$$

$$A_2 = A_1\left(t_{n-1} - t_1\right) - K_{PR}\frac{\left(t_{n-1} - t_1\right)^2}{2} + \frac{I_2}{\Delta U}, \tag{203}$$

$$A_3 = A_2\left(t_{n-1} - t_1\right) - A_1\frac{\left(t_{n-1} - t_1\right)^2}{2} + K_{PR}\frac{\left(t_{n-1} - t_1\right)^3}{6} - \frac{I_3}{\Delta U}, \tag{204}$$

$$A_4 = A_3\left(t_{n-1} - t_1\right) - A_2\frac{\left(t_{n-1} - t_1\right)^2}{2} + A_1\frac{\left(t_{n-1} - t_1\right)^3}{6} - K_{PR}\frac{\left(t_{n-1} - t_1\right)^4}{24} + \frac{I_4}{\Delta U}, \tag{205}$$

$$A_5 = A_4\left(t_{n-1} - t_1\right) - A_3\frac{\left(t_{n-1} - t_1\right)^2}{2} + A_2\frac{\left(t_{n-1} - t_1\right)^3}{6} - A_1\frac{\left(t_{n-1} - t_1\right)^4}{24} + K_{PR}\frac{\left(t_{n-1} - t_1\right)^5}{120} - \frac{I_5}{\Delta U}, \tag{206}$$

where I_1 to I_5 are recursively calculated multiple integrations of the process step response:

$$I_1 = \int_{t_1}^{t_{n-1}} y(\tau)d\tau, \tag{207}$$

$$I_2 = \int_{t_1}^{t_{n-1}}\left[\int_{t_1}^{\tau_1} y(\tau)d\tau\right]d\tau_1, \tag{208}$$

$$I_3 = \int_{t_1}^{t_{n-1}} \left[\int_{t_1}^{\tau_3} \left[\int_{t_1}^{\tau_2} y(\tau_1) d\tau_1 \right] d\tau_2 \right] d\tau_3 \ , \tag{209}$$

$$I_4 = \int_{t_1}^{t_{n-1}} \left[\int_{t_1}^{\tau_4} \left[\int_{t_1}^{\tau_3} \left[\int_{t_1}^{\tau_2} y(\tau_1) d\tau_1 \right] d\tau_2 \right] d\tau_3 \right] d\tau_4 \ , \tag{210}$$

$$I_5 = \int_{t_1}^{t_{n-1}} \left[\int_{t_1}^{\tau_5} \left[\int_{t_1}^{\tau_4} \left[\int_{t_1}^{\tau_3} \left[\int_{t_1}^{\tau_2} y(\tau_1) d\tau_1 \right] d\tau_2 \right] d\tau_3 \right] d\tau_4 \right] d\tau_5 \ , \tag{211}$$

where the process step response $y(t)$ is approximated by the linear function between two samples:

$$y(t) = y(k-1) + \frac{y(k) - y(k-1)}{T_S} \left(t - t(k-1) \right) \ ; \ \ t(k-1) < t \le t(k) \, , \tag{212}$$

as given in Fig. 102.

After the process steady-state gain K_{PR} and areas A_1 to A_5 are obtained, the PI and the PID controller parameters are derived from expressions (159), (160), (162), (163), and (180), and (181) by fixing ρ=0.2.

Note that the calculated proportional gain of the PI and PID controller for the dominantly first-order process or the proportional gain of the PID controller for dominantly second-order process can become very high or even infinite (K=∞). In such a case, the open-loop gain is restricted to:

$$K \le \frac{K_{\mathbf{max}}}{K_{PR}} \ , \tag{213}$$

where K_{max} is a given user-defined parameter (see Chapter 9.1.1).

By solving expressions (162), and (213), the factor α_D can be recalculated:

$$\alpha_D = \frac{0.5}{K_{\max}} \ . \tag{214}$$

When applying (214) to (160) and (163), the remaining two PID parameters can be obtained.

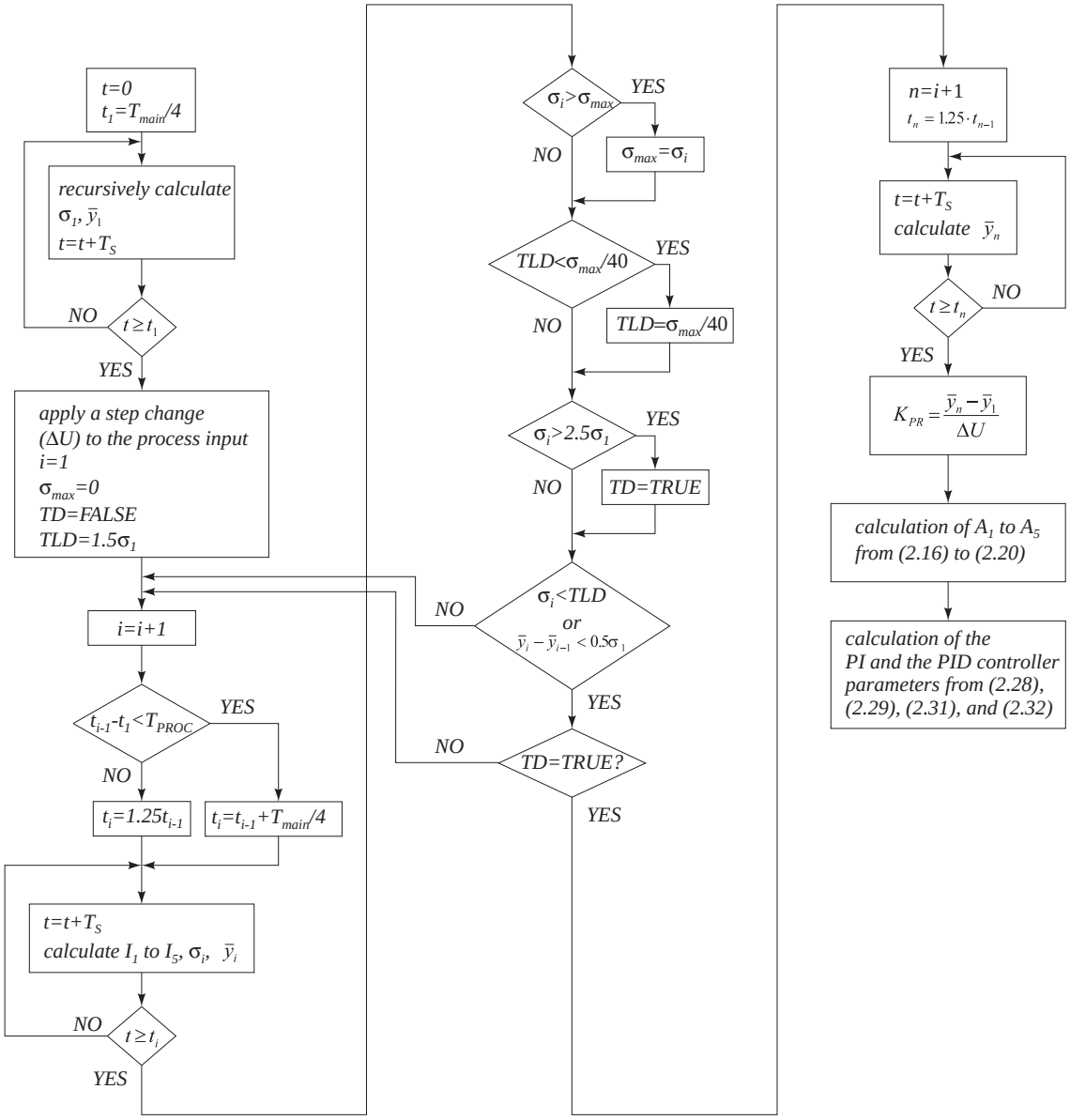

Fig. 101. Block-diagram of the auto-tuning algorithm whilst performing the open-loop step response.

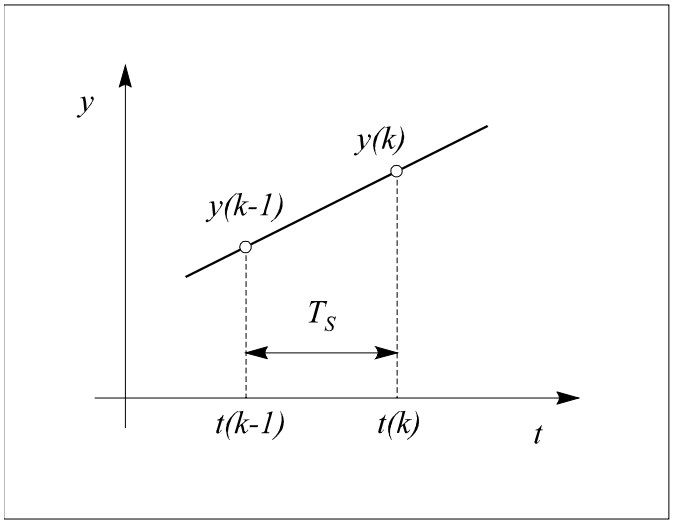

Fig. 102. The continuous-time approximation of the process step-response between two discrete samples.

In order to achieve an even more robust auto-tuning algorithm for the PID controllers, the proportional gain of the PID controller is additionally limited to four times the proportional gain of the PI controller. As factor α corresponds to the PI controller and α_D corresponds to the PID controller (160), this limitation can be expressed in the following way:

$$\alpha_D \geq \frac{\alpha}{4} \ . \tag{215}$$

9.1.4 Performing closed-loop experiments

After calculating areas A_1 to A_5, the PI and the PID controller parameters (the calculation is very fast due to the recursive way of numerical integration), the algorithm switches into automatic mode (closed-loop).

After switching to automatic mode, the reference step changes are applied (only for testing purposes), first by using the PI controller (using expressions (160) (T_d=0), (162) and (163)) from $t_n<t\leq t_{n+2}$, then by applying the PID controller obtained by fixing the ratio $\rho=T_d/T_i$=0.2 (see (180) and (181)) from $t_{n+2}<t\leq t_{n+3}$, and then by using the PID controller using expressions (159), (160), (162), and (163) from $t_{n+3}<t\leq t_{n+4}$.

Note that in practical realisation of the PID controllers, the implementation of the appropriate anti-windup protection is of high importance. In this auto-tuning algorithm, the conditioning technique is applied as an anti-windup protection (see part I of this thesis).

9.2 Real-time experiments on laboratory plants

Four real-time experiments were performed on laboratory plants.

The first experiment was made on a very simple model of the third-order process, comprised of resistors, capacitors, and the operational amplifiers (R-C chain) as given in Fig. 103[3]. The process is very linear with almost no noise (in fact, only the quantisation noise of the A/D converter is present).

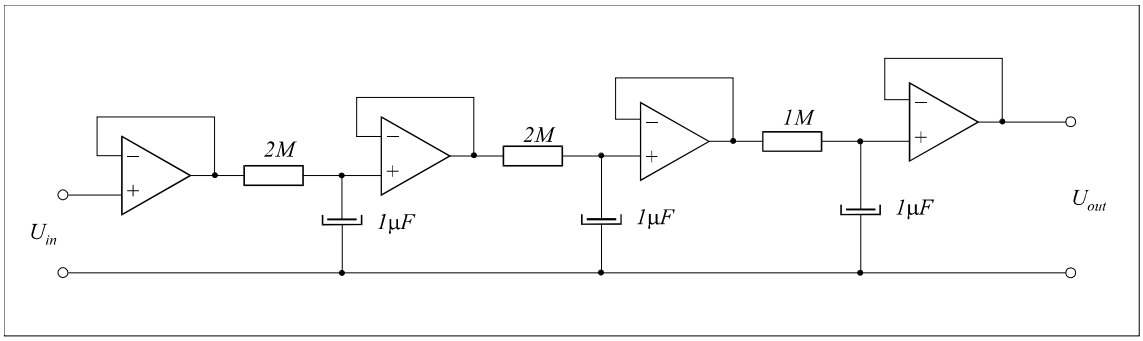

Fig. 103. The R-C chain "plant"

Fig. 104 shows the results of the real-time experiment on the R-C chain, when using the auto-tuning algorithm. After the process open-loop step response is obtained (from t=2s to t=18s - see the step-change of the process input), the following values of the process gain, areas, and factors α and α_D are obtained:

[3] The input filters of the Burr-Brown acquisition system are not shown in Fig. 103, but it should be noted that they contribute additional dynamics to the system.

K_{PR}	A_1	A_2	A_3	A_4	A_5	α	α_D
0.66033	3.0872	9.6234	24.521	54.086	105.57	0.835	0.172

The following PI and PID controller parameters were obtained from expressions (160), (162), (163), (159), (180) and (181):

	K	T_i	T_d
PI	0.907	2.548	
PID (T_d=0.2T_i)	1.656	3.209	0.642
PID (159) - (modified α_D)	3.627	3.868	1.064

Note that the PID controller parameters (159) are modified so as to comply with the restriction $\alpha_D \geq \alpha/4$.

The closed-loop responses (see Fig. 104) are very good for all three controllers used. It is obvious that the closed-loop response becomes faster when using a controller with higher proportional gain without an increase of the plant overshoot. Moreover, the overshoot is even smaller when a controller with maximum gain (PID controller) is used. The closed-loop responses in Fig. 104 are shown in more detail in Fig. 105.

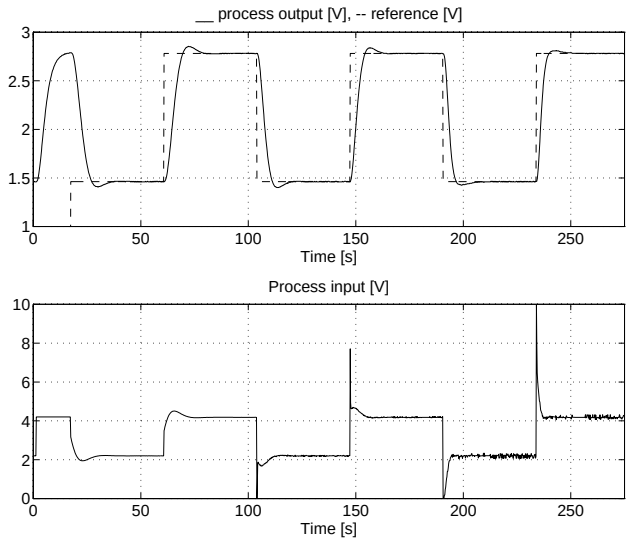

Fig. 104. The system responses under the auto-tuning algorithm for the R-C process

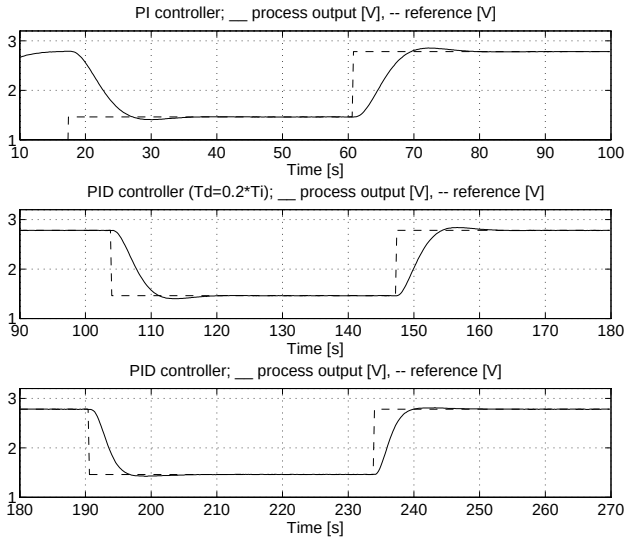

Fig. 105. The closed-loop responses under the real-time auto-tuning algorithm (the detailed view of Fig. 104)

The second experiment was made on a motor-generator plant, as shown in Fig. 106. The plant input is the voltage on the amplifier input (U_{in}) which drives the motor, and the output is the speed of the motor-generator system measured at the output of the speed-to-voltage converter (U_{out}). Both input and output signals are in the range from 0 to 10V.

The motor-generator plant is quite non-linear. The steady-state input-output characteristic of the process is given in Fig. 107, where the solid line represents the characteristics obtained when increasing the process input voltage, whilst the dashed line represents the characteristics obtained when decreasing the input voltage.

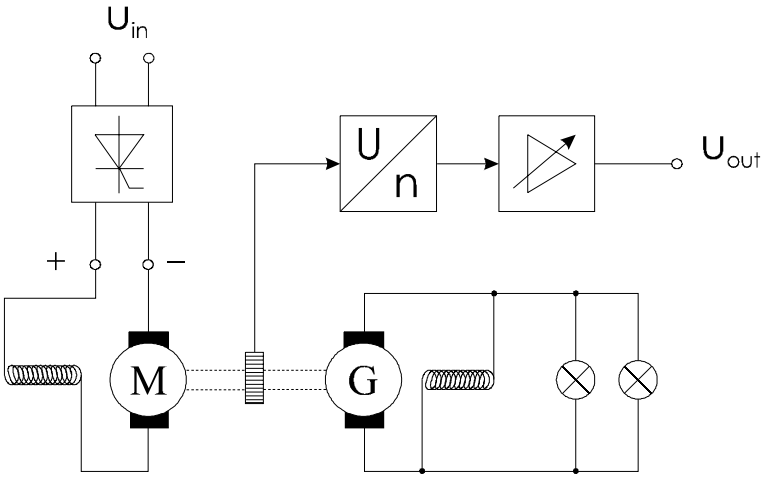

Fig. 106. Motor-generator laboratory set-up

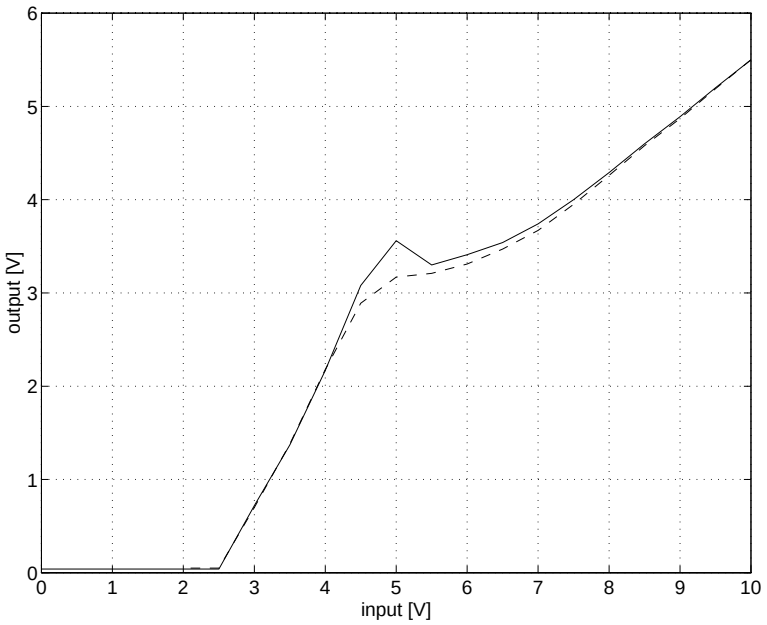

Fig. 107. The static characteristics of the motor-generator set-up; __ increasing the input voltage, -- decreasing the input voltage.

The system time response, when driven by the real-time auto-tuning algorithm, is shown in Fig. 108. It can be seen that the open-loop step response takes 0.65s (from t=0.25s to t=0.9s), whilst the sampling time is T_S=0.01s.

The following values of the process gain, areas, and factors α and α_D are obtained:

K_{PR}	A_1	A_2	A_3	A_4	A_5	α	α_D
0.644	0.1221	$1.435 \cdot 10^{-2}$	$1.311 \cdot 10^{-3}$	$1.001 \cdot 10^{-4}$	$6.607 \cdot 10^{-6}$	1.076	0.3702

The following PI and PID controller parameters were obtained from (160), (162), (163), (159), (180) and (181):

	K	T_i	T_d
PI	0.721	0.0914	
PID (T_d=0.2T_i)	1.148	0.1131	0.0226
PID (159) - (modified α_D)	2.096	0.1384	0.0399

The system response (see Fig. 108) is very good for all three controllers. It is obvious that the process response becomes faster as proportional gain increases. Different process transients at low and high reference levels indicate non-linear process characteristics.

The closed-loop responses in Fig. 108 are shown in Fig. 109.

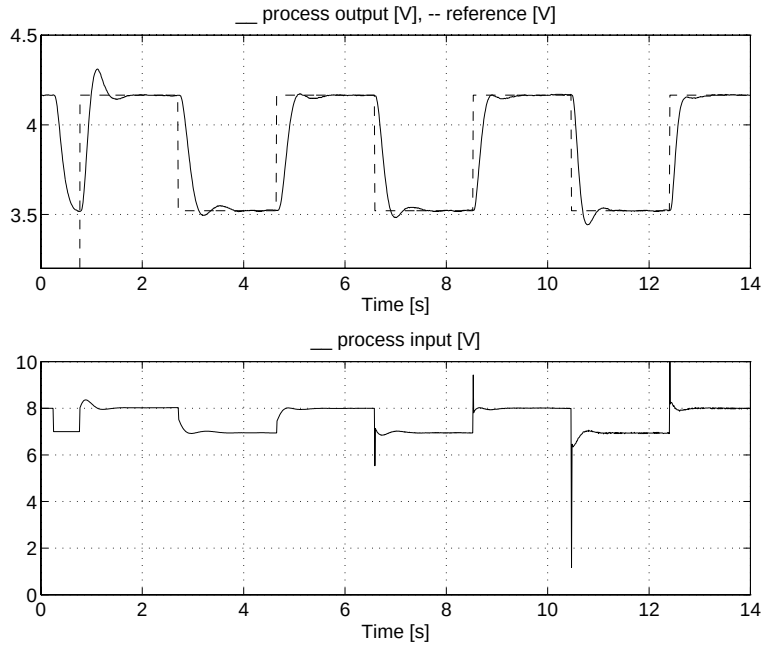

Fig. 108. The system responses under the auto-tuning algorithm for the motor-generator set-up

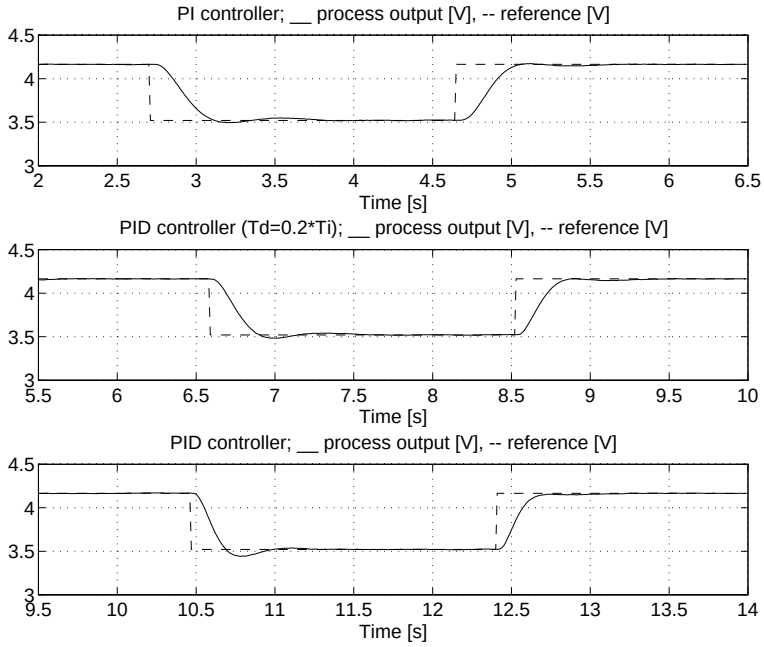

Fig. 109. The closed-loop responses under the real-time auto-tuning algorithm (the detailed view of Fig. 108)

The third experiment was made on a pneumatic plant, as depicted in Fig. 110. The plant input is the current reference (i) on the servo-driven valve, and the output is the pressure p_1 between valves V_1 and V_2 (transferred to voltage by the pressure-to-voltage transmitter). The input current is in the range of 4-20mA, whilst the output voltage is in the range from 0 to 1V.

The pneumatic plant is also non-linear. The steady-state input-output characteristic of the process is given in Fig. 111, where the solid line represents the characteristics obtained when increasing the current at the process input, whilst the dashed line represents the characteristics obtained when decreasing the input current.

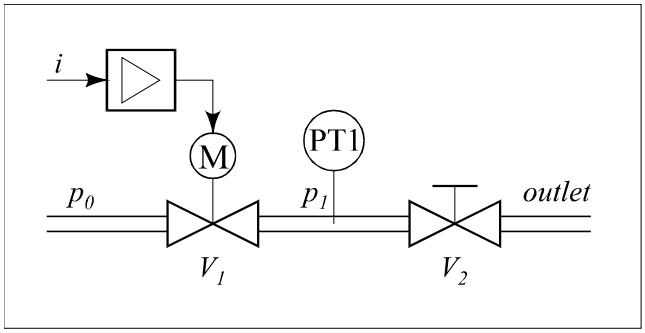

Fig. 110. Pneumatic laboratory set-up.

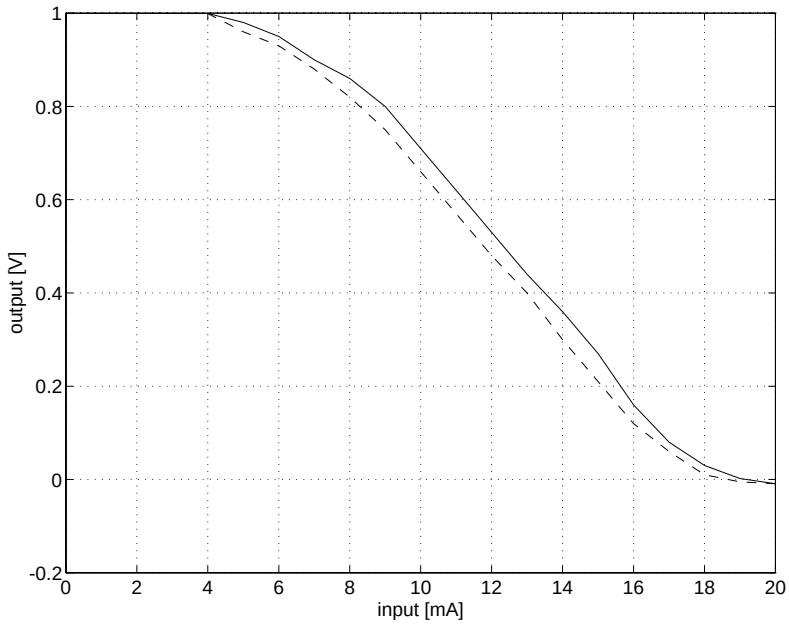

Fig. 111. The static characteristics of the pneumatic set-up (output voltage); __ increasing the input current, -- decreasing the input current.

The system time response, when driven by the auto-tuning algorithm, is shown in Fig. 112.

The following values of the process gain, areas, and factors α and α_D are obtained:

K_{PR}	A_1	A_2	A_3	A_4	A_5	α	α_D
-0.089	$-2.203\cdot10^{-2}$	$-3.723\cdot10^{-3}$	$-5.359\cdot10^{-4}$	$-6.857\cdot10^{-5}$	$-7.85\cdot10^{-6}$	0.715	0.0472

Note that the PID controller parameters (159) are modified so as to comply with the restriction $\alpha_D \geq \alpha/4$.

The following PI and PID controller parameters were obtained from (160), (162), (163), (159), (180) and (181):

	K	T_i	T_d
PI	-7.835	0.1439	
PID (T_d=0.2T_i)	-16.39	0.184	0.0368
PID (159) - (modified α_D)	-31.34	0.2094	0.0529

The system response (see Fig. 112) is very good when using all three controllers. It is obvious that the closed-loop response becomes faster when controllers with higher proportional gain are used. Different closed-loop transients at low and high reference levels again indicate the non-linear characteristics of the plant. The higher process delay time is particularly noticeable when decreasing the pressure, rather than when increasing the pressure, as shown in Fig. 113.

The closed-loop responses in Fig. 112 are shown in Fig. 113.

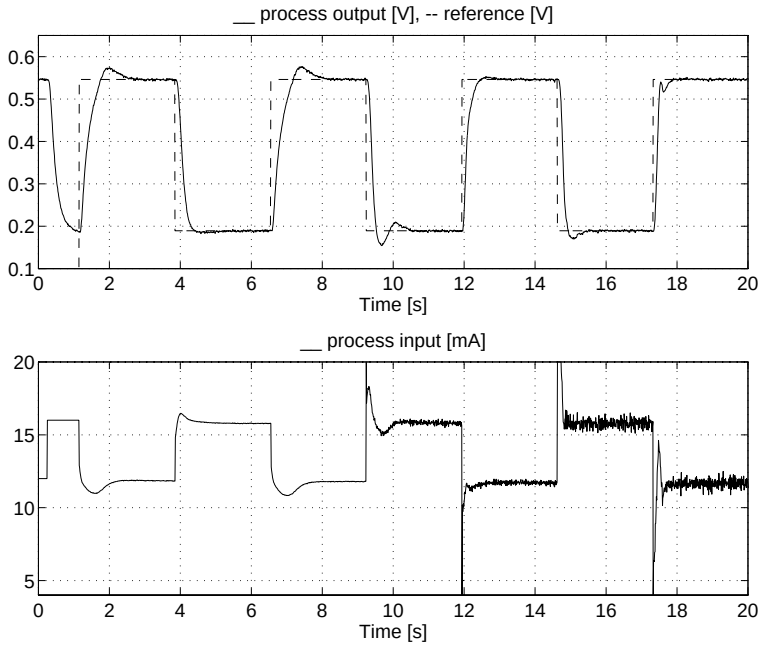

Fig. 112. The system responses under the auto-tuning algorithm for the pneumatic set-up

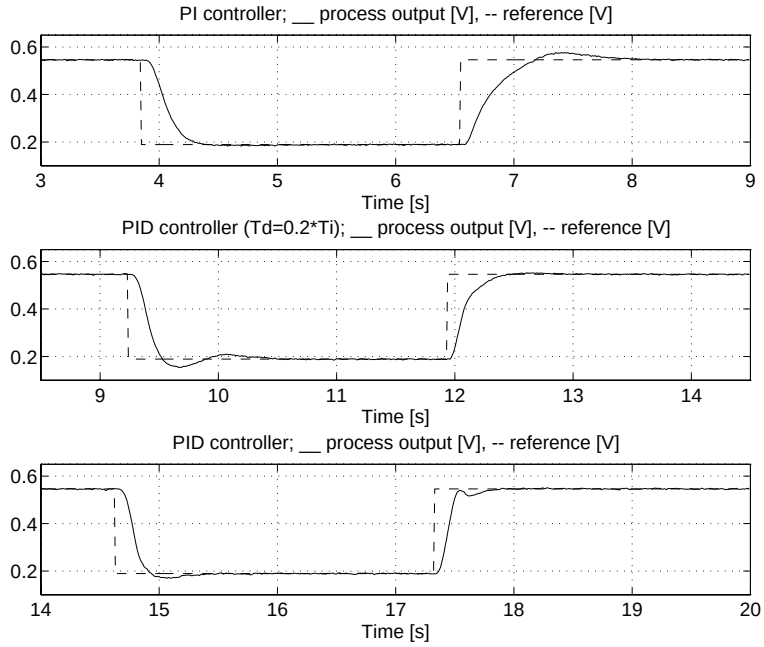

Fig. 113. The closed-loop responses under the real-time auto-tuning algorithm (the detailed view of Fig. 112)

The fourth experiment was made on a three-water-columns laboratory set-up (Vrančić et al., 1993d), which is shown in Fig. 114.

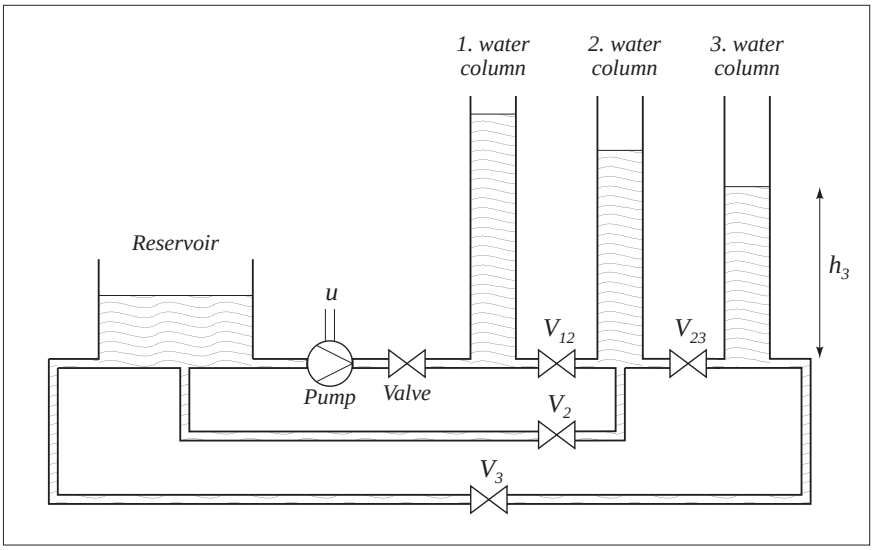

Fig. 114. Three-water-columns laboratory set-up.

The water comes from the reservoir through the pump to water column 1. It then flows through the valve V_{12} to water column 2 and then through V_2 and V_{23} after which it flows back to the reservoir and into the third water column. The water flows from column 3 through V_3 back to the reservoir. The process input is the voltage on the pump (u), and the process output is the water level in the third column (h_3). The input-output characteristic of the process is non-linear (in the steady-state $h_3 \propto u^2$).

The system time response, when driven by the auto-tuning algorithm, is shown in Fig. 115.

The following values of the process gain, areas, and factors α and α_D are obtained:

K_{PR}	A_1	A_2	A_3	A_4	A_5	α	α_D
1.0605	197.22	$2.7274 \cdot 10^4$	$3.2409 \cdot 10^6$	$3.3652 \cdot 10^8$	$3.0693 \cdot 10^{10}$	0.565	-0.0796

It can be seen that the calculated α_D is negative. The reason for this could be due to the highly non-linear and dominantly second-order behaviour of the process, variations in the speed of the pump, and the non-linearity of the level sensor. The auto-tuning algorithm corrects the value α_D to $\alpha_D=\alpha/4=0.141$. It is therefore clearly shown that the multiple integrations up to area A_5 cannot be successfully used for relatively non-linear and disturbed processes.

The following PI and PID controller parameters were obtained from (160), (159), (162), (163), (180) and (181), by taking into account the restrictions of the auto-tuning algorithm ($\alpha_D \geq \alpha/4$):

	K	T_i	T_d
PI	0.834	37.5	
PID ($T_d=0.2T_i$)	2.143	152.4	30.49
PID (159) - (modified α_D)	3.338	163.0	37.45

All of the closed-loop responses are once again very good. As in the previous examples, it is obvious that the closed-loop process response becomes faster when controllers with higher proportional gain are used, without increasing the process overshoot. On the other hand, the process input becomes noisier when both PID controllers are used because of the quantisation of the A/D converter. The process non-linear behaviour can be clearly seen from the closed-loop responses, where no overshoots are visible when decreasing the reference signal, whilst overshoots are noticable when the reference signal is increased.

The closed-loop responses in Fig. 115 are shown in Fig. 116.

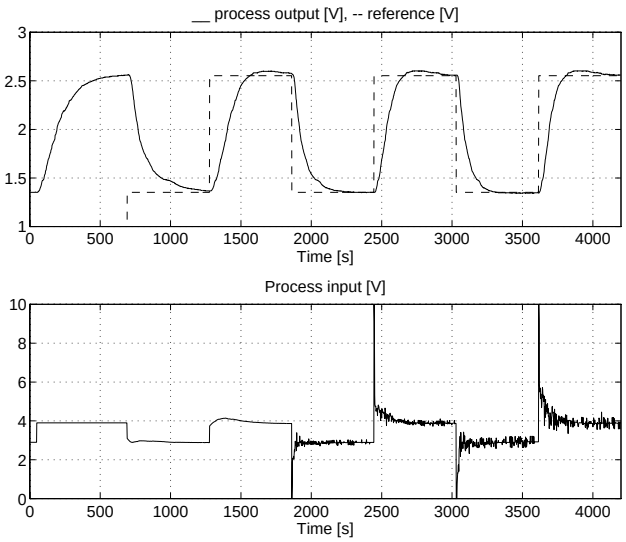

Fig. 115. The system responses under the auto-tuning algorithm for the three-water-columns process

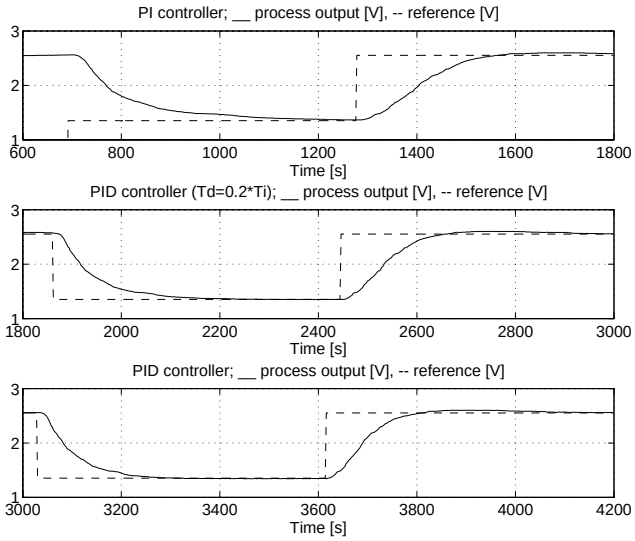

Fig. 116. The closed-loop responses under the real-time auto-tuning algorithm (detailed view of Fig. 115)

10. Discussion

The aim of this part of the thesis was to find a simple tuning method for the PI(D) controller which would be suitable for a large set of processes. A frequency domain tuning criterion (magnitude optimum) was chosen. It was shown that the parameters of the PI controller can be calculated from the step response by the multiple integration method in a very simple way, using the areas A_1, A_2, and A_3. This also holds when calculating the PID controller parameters, if areas A_4 and A_5 can be successfully obtained. The derivative term time constant can also be obtained by fixing the ratio $\rho=T_d/T_i$. This, however, does not result in the optimal closed-loop time response, according to MO criterion.

Tests on the laboratory plants also showed that the method is relatively robust to process noise and non-linearity. The main point, however, is that in spite of the considerably *demanding frequency criterion* - and the fact that the calculation of the controller parameters is based on the *complete* process model - implementation using the time domain approach is very simple and straightforward.

The method also resulted in excellent PI controller tuning performances for high-order processes, highly non-minimal phase processes and for those processes with higher time delays (Vrančić, 1995b).

The advantages of the new tuning approach are:

- There is no need to detect the process inflection point and the slope from the process step response (which is difficult in a noisy environment). The method is based on multiple integrations (summations) and is therefore suitable for use in the auto-tuning algorithms.
- The controller parameters are calculated *exactly*, according to the given MO criterion, for a wide spectrum of process models.
- The calculation of areas A_1 to A_3 is neither too sensitive to process noise nor to non-linearity (Vrančić, 1995b).

The drawback of such an approach is that the method requires a *stable* open-loop process response in order to determine the appropriate controller parameters. However, the closed-loop stability is not guaranteed. Only the necessary condition for stability is derived.

However, the new tuning method was tested in Vrančić et al. (1995a), and the experimental results showed very good closed-loop time responses for all the process models given in Appendix E. In Vrančić (1995b), a few additional processes were also tested.

The new method can be successfully used for improving the classical tuning rules. It can also be used in a pre-tuning stage when some other control goal is to be met. The extension of the new tuning method (using the multiple integration approach) to other types of controllers, two-degrees-of freedom PID controllers, as well as MIMO PI controllers (Vrančić et al., 1997a) is also straightforward.