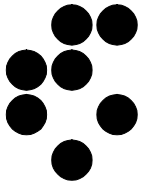
University of Ljubljana

J. Stefan Institute, Ljubljana, Slovenia



Report **DP-7298**

A new PI(D) tuning method based on the process reaction curve

Damir Vrančić

November 1995

Table of contents

1. Introduction

Today, the most applied tuning rules for PID types of controllers are ones, based on the process reaction curve (process step response). Also very popular, but not so widely used, are the tuning rules based on detection of one particular point on the process Nyquist curve (usually detecting the process ultimate frequency and amplitude). The best known rules are the Ziegler-Nichols tuning rules [Åström and Hägglund, 1995] [Hang and Cao, 1993] [Hang et al., 1991] [Thomas, 1991]. The reason of such popularity lies in their *simplicity*. In real plants it is usually quite easy to obtain a process step response and find the appropriate process lag and process rise times, which serves as a basis for calculation of PID parameters.

The drawbacks of these tuning methods arise from the lack of information based only on the process lag and process rise time. There are different processes existing with the same pair of lag and rise times, but which need different controller tuning. Different problems, arising when using the Ziegler-Nichols tuning rules, are reported in [Åström and Hägglund, 1995] [Hang and Sin, 1991] [Hang and Cao, 1993] [Hang et al., 1991] [Thomas, 1991] [Vrančić et al., 1993] [Vrančić et al., 1995]. Some authors suggested the introduction of so-called "set-point weighting" [Hang et al., 1991] and additional change of tuning rules [Åström and Hägglund, 1995]. The improved closed-loop response showed that they were very successful, but the rules were still based on the process lag and rise time, so their applicability to wide spectrum of possible processes still stays as an opened question.

Besides the very popular rules, based on the process step response or detecting the process ultimate point, as e.g. Ziegler-Nichols, Cohen-Coon, Chien-Hrones-Reswick rules, a more sophisticated approaches are also existing. They are usually based on the detecting more points on the process Nyquist curve [Åström and Hägglund, 1995] [Ho et al., 1993] [Leva, 1993] [Thomas, 1991] as to extract more information from the process.

Such tuning rules generally give better closed-loop performance at the cost of more extensive computations with longer and more sophisticated experiments on the testing plant. Sometimes, the plant must be driven into self-oscillations, that is frequently intolerable.

The scope of this work was to find such tuning rules for PI(D) controller that will use the positive sides of both approaches. Therefore the aim of this report is to show how the information from the process reaction curve can be successfully used for finding such controller parameters (K, T_i and indirectly also T_d) which will satisfy given frequency criterion (the frequency response method given by [Hanus, 1975]) for wide spectrum of chosen process types. Moreover, the simple tuning rules will give the exact result for the chosen processes.

2. Background materials

2.1. PI controller and process models

The PI controller is given in (1)

$$U = K\left(1 + \frac{1}{sT_i}\right)(W - Y), \tag{1}$$

where U, W and Y denote the Laplace transforms of the controller output u, set-point w and the process (plant) output y, respectively. The controller parameters K and T_i denote proportional gain and integral time constant, respectively. The PI controller in a closed-loop configuration with a process is shown in Fig. 1.

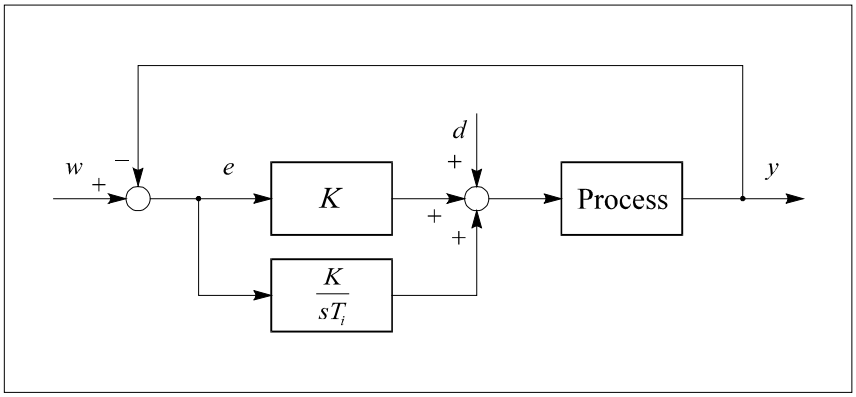


Fig. 1. The closed loop system with PI controller

The symbol *d* in Fig. 1 denotes the disturbance at the process input.

The so-called "frequency response method" was chosen for tuning the PI controller since

- it assures stability conditions for wide spectrum of process models, driven by PI(D) type of controller
- it gives the greatest proportional gain K which still assures the aperiodic process time response for the wide spectrum of process models
- the controller parameters can be simply readjusted such as to speed-up or slow-down the process closed-loop response

The method was tested by [Vrančić et al., 1995] and the experimental results showed that the method gave very good closed-loop responses for variety of tested process models which are given in Table 1.

Process	***Range of parameter***
$G_{P1}(s) = \frac{e^{-s}}{(1+sT)}$	$T = 0.1s \ldots 10s$
$G_{P2}(s) = \frac{e^{-s}}{(1+sT)^2}$	$T = 0.1s \ldots 10s$
$G_{P3}(s) = \frac{1}{(1+sT)(1+s)}$	$T = 0.1s \ldots 10s$
$G_{P4}(s) = \frac{1}{(1+sT)^2(1+s)^2}$	$T = 0.1s \ldots 10s$
$G_{P5}(s) = \frac{1}{(1+s)^n}$	$n = 2 \ldots 8$
$G_{P6}(s) = \frac{1}{(1+s)(1+sT)(1+sT^2)(1+sT^3)}$	$T = 0.2s \ldots 0.8s$
$G_{P7}(s) = \frac{(1-sT)}{(1+s)^3}$	$T = 0.1s \ldots 10s$
$G_{P8}(s) = \frac{e^{-s}(1+sT)}{(1+s)^2}$	$T = 0.1s \ldots 1s$
$G_{P9}(s) = \frac{1}{(1+s)(1+s(1+i\alpha))(1+s(1-i\alpha))}$	$\alpha = 0.1s \ldots 1s$

Table 1. The tested processes with parameter ranges

Moreover, the processes were not only tested in given range of parameters, as will be outlined later for processes $G_{P8}(s)$ and $G_{P9}(s)$.

2.2. Frequency response method

The main idea of the frequency response method (FRM) is to find such a controller $G_C(j\omega)$ for given process $G_P(j\omega)$ which will drive the frequency response of $G_P(j\omega)G_C(j\omega)$ toward the vertical line

$$\delta(\gamma) = -\frac{1}{2} + j\gamma \quad \gamma = [-\infty \ldots \infty], \tag{2}$$

such as

$$\mathbf{Re}\left\{G_P(j\omega)G_C(j\omega)\right\} \geq -\frac{1}{2}. \tag{3}$$

$$\mathbf{Re}\left\{G_P(0)G_C(0)\right\} = -\frac{1}{2} \tag{4}$$

The reason for such limitations lies in the fact that in such case the *closed-loop* response (see e.g. *M* circles in [DiStefano et al., 1990]) will have the *unchanged gain* (*M* circle is 1) at low frequencies, and the gain will *decrease* at higher frequencies (see Fig. 2).

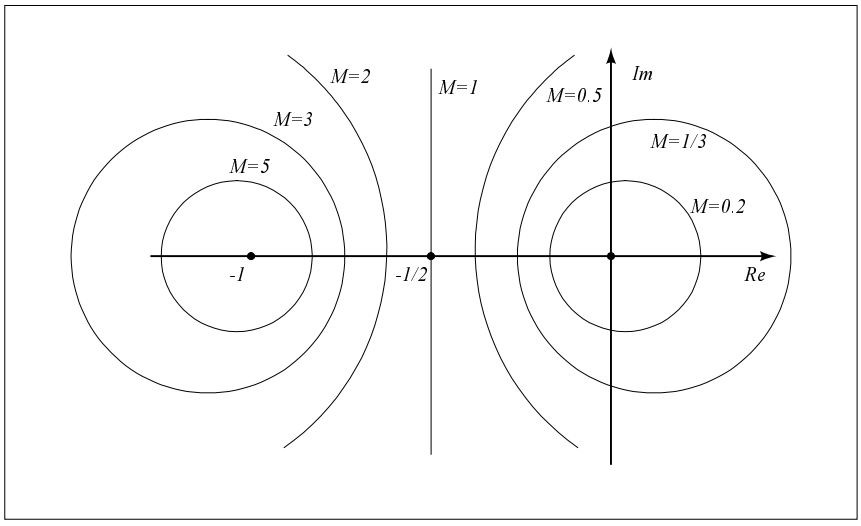


Fig. 2. M circles in the Polar plot

The open-loop Nyquist plot of the process curve, which follows the line $M=\beta$, will have the unchanged closed-loop gain $M=\beta$ at all frequencies. If the open-loop frequency response corresponds to conditions (3) and (4), the closed-loop amplitude response will go from $M=1$ at lower frequencies to $M<1$ at higher frequencies (see Fig. 2). Therefore, there will be no

resonance peak in closed-loop amplitude frequency response, so the system will be critically damped [DiStefano et al., 1990] [Boucher and Tanguy, 1976] and no oscillations will exist in closed-loop time response.

A typical process Nyquist curve $G_P(j\omega)$ is shown in Fig. 3.

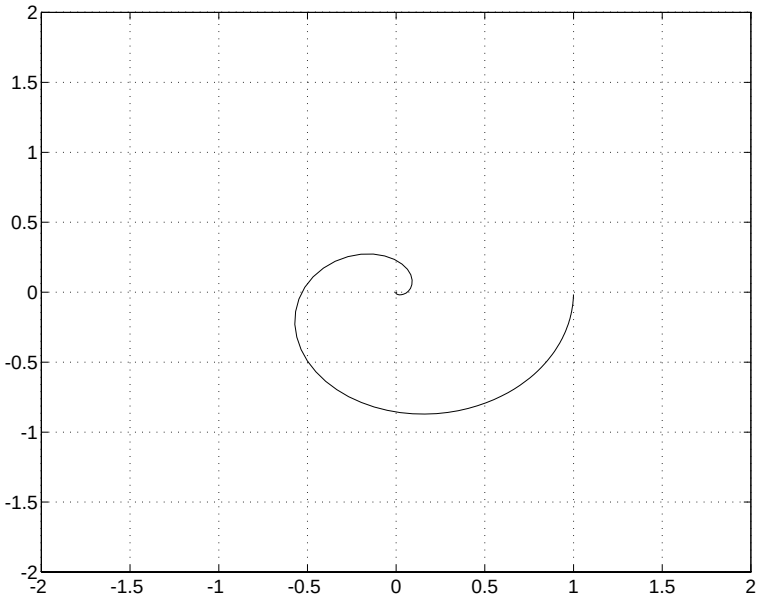


Fig. 3. Typical process Nyquist curve $G_p(j\omega)$

The frequency response method should therefore find such controller which will satisfy expressions (3) and (4), that the Nyquist curve will be such as shown in Fig. 4.

The frequency response method has another advantage. The given frequency limitations will also assure the system stability. The amplitude margin will always be greater than or equal to 2 ($A_m \geq 2$) and the phase margin greater than or equal to 60 degrees ($\varphi_m \geq 60°$) (see (3)).

Moreover, by changing the position of the vertical line $\delta(\gamma)$ from position $\mathbf{Re}\{\delta\}=-1/2$ toward higher or toward lower negative values, we can obtain faster or more sluggish process closed-loop time response, respectively.

Controller parameters can be adjusted in different ways. One of the possible concepts is using the optimisation and the second one is the analytical concept. While in this report we are interested in analytical results, we will develop the appropriate K and T_i analytically. In [Vrančić et al., 1995] the controller parameters, for given processes in Table 1, were obtained by the optimisation. Finally, the analytical solutions will be compared with the optimisation results.

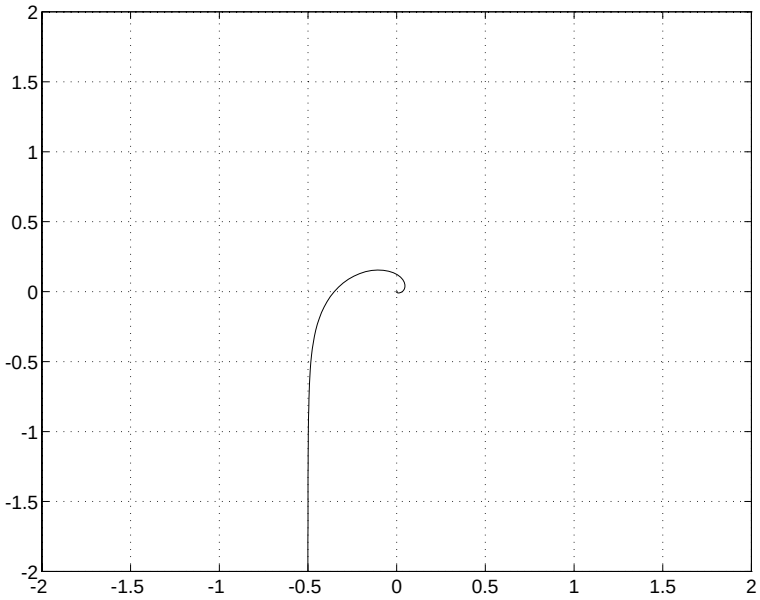


Fig. 4. The Nyquist curve of $G_p(j\omega)*G_c(j\omega)$

3. Theoretical evaluation

Let us have the following process transfer function:

$$G_P(s) = \frac{1 + b_1 s + b_2 s^2 + b_3 s^3 + \ldots + b_m s^m}{1 + a_1 s + a_2 s^2 + a_3 s^3 + \ldots + a_n s^n} \tag{5}$$

The static process gain is assumed to be equal to 1. In further evaluation will be suggested how to handle processes with the static gain which is different from 1.

When using the PI controller, given by (1), we gain the following open-loop system transfer function:

$$G_C(s)G_P(s) = \frac{d_0 + d_1 s + d_2 s^2 + d_3 s^3 + \ldots + d_q s^q}{c_0 s + c_1 s^2 + c_2 s^3 + c_3 s^4 + \ldots + c_p s^p}, \tag{6}$$

where $G_C(s)G_P(s)$ must be a strictly proper function. Constants c_i and d_i in (6) can be calculated by inserting (1) and (5) into (6):

$$\begin{aligned}
c_0 &= T_i \\
c_1 &= a_1 T_i \\
c_2 &= a_2 T_i \\
c_3 &= a_3 T_i \\
&\vdots \\
d_0 &= K \\
d_1 &= K(b_1 + T_i) \\
d_2 &= K(b_2 + b_1 T_i) \\
d_3 &= K(b_3 + b_2 T_i) \\
&\vdots
\end{aligned} \tag{7}$$

With the substitution of s by $j\omega$ in (6), the frequency response can be obtained:

$$G_C(j\omega)G_P(j\omega) = \frac{d_0 - d_2\omega^2 + \ldots + j\omega(d_1 - d_3\omega^2 + \ldots)}{-c_1\omega^2 + c_3\omega^4 - \ldots + j\omega(c_0 - c_2\omega^2 + \ldots)} \tag{8}$$

The real part of (8) comes to

$$\mathbf{Re}\left\{G_C(j\omega)G_P(j\omega)\right\} =$$
$$= \frac{\left(d_0 - d_2\omega^2 + \ldots\right)\left(-c_1\omega^2 + c_3\omega^4 - \ldots\right) + \omega^2\left(d_1 - d_3\omega^2 + \ldots\right)\left(c_0 - c_2\omega^2 + \ldots\right)}{\left(-c_1\omega^2 + c_3\omega^4 - \ldots\right)^2 + \omega^2\left(c_0 - c_2\omega^2 + \ldots\right)^2} = \tag{9}$$
$$= \frac{c_0 d_1 - c_1 d_0 + \omega^2\left(c_3 d_0 + c_1 d_2 - c_0 d_3 - c_2 d_1\right) + \ldots}{{c_0}^2 + \omega^2\left({c_1}^2 - 2c_0 c_2\right) - \ldots}$$

When inserting (4) into (9), we get:

$$\mathbf{Re}\left\{G_C(j0)G_P(j0)\right\} = \frac{c_0 d_1 - c_1 d_0}{{c_0}^2} = -\frac{1}{2} \tag{10}$$

It is somehow more tough task to satisfy condition (3). The solution could be in setting the next set of equations [Boucher and Tanguy, 1976]:

$$\begin{aligned}
&\left.\frac{\partial \mathbf{Re}\left\{G_C(j\omega)G_P(j\omega)\right\}}{\partial\omega}\right|_{\omega=0} = 0 \\
&\left.\frac{\partial^2 \mathbf{Re}\left\{G_C(j\omega)G_P(j\omega)\right\}}{\partial\omega^2}\right|_{\omega=0} = 0 \\
&\qquad\vdots \\
&\left.\frac{\partial^{n-1} \mathbf{Re}\left\{G_C(j\omega)G_P(j\omega)\right\}}{\partial\omega^{n-1}}\right|_{\omega=0} = 0
\end{aligned}, \tag{11}$$

where n determines the order of differential equations in (6). Let us now calculate the first derivative:

$$\frac{\partial \mathbf{Re}\left\{G_C(j\omega)G_P(j\omega)\right\}}{\partial\omega} =$$
$$= \frac{2\omega\left({c_0}^2\left[c_3 d_0 + c_1 d_2 - c_0 d_3 - c_2 d_1\right] - c_0 {c_1}^2 d_1 + {c_1}^3 d_0 + 2{c_0}^2 c_2 d_1 - 2c_0 c_1 c_2 d_0\right)}{\left[{c_0}^2 + \omega^2\left({c_1}^2 - 2c_0 c_2\right)\right]^2} \tag{12}$$

From expression (12), it can be seen that the first expression in (11) is already satisfied, when inserting $\omega=0$. The second derivative equals to:

$$\frac{\partial^2 \mathbf{Re}\{G_C(j\omega)G_P(j\omega)\}}{\partial\omega^2} =$$
$$= \frac{2\left(c_0^2\left[c_3d_0 + c_1d_2 - c_0d_3 - c_2d_1\right] - c_0c_1^2d_1 + c_1^3d_0 + 2c_0^2c_2d_1 - 2c_0c_1c_2d_0\right)\left[c_0^2 + \omega^2\left(c_1^2 - 2c_0c_2\right)\right]^2}{\left[c_0^2 + \omega^2\left(c_1^2 - 2c_0c_2\right)\right]^4} -$$
$$- \frac{4\omega^2\left[c_0^2 + \omega^2\left(c_1^2 - 2c_0c_2\right)\right]\left(c_1^2 - 2c_0c_2\right)\left(c_0^2\left[c_3d_0 + c_1d_2 - c_0d_3 - c_2d_1\right] - c_0c_1^2d_1 + c_1^3d_0 + 2c_0^2c_2d_1 - 2c_0c_1c_2d_0\right)}{\left[c_0^2 + \omega^2\left(c_1^2 - 2c_0c_2\right)\right]^4} = \tag{13}$$
$$= \frac{2\left(c_0^2\left[c_3d_0 + c_1d_2 - c_0d_3 - c_2d_1\right] - c_0c_1^2d_1 + c_1^3d_0 + 2c_0^2c_2d_1 - 2c_0c_1c_2d_0\right)}{\left[c_0^2 + \omega^2\left(c_1^2 - 2c_0c_2\right)\right]^4} \cdot$$
$$\cdot\left(\left[c_0^2 + \omega^2\left(c_1^2 - 2c_0c_2\right)\right]^2 - 4\omega^2\left[c_0^2 + \omega^2\left(c_1^2 - 2c_0c_2\right)\right]\left(c_1^2 - 2c_0c_2\right)\right)$$

It can be seen that expression (13) becomes zero, when

$$c_0^2\left[c_3d_0 + c_1d_2 - c_0d_3 - c_2d_1\right] - c_0c_1^2d_1 + c_1^3d_0 + 2c_0^2c_2d_1 - 2c_0c_1c_2d_0 = 0\,. \tag{14}$$

If (14) is satisfied, then the higher derivatives in (11) also equal to zero, so (11) is completely fulfilled.

To satisfy conditions (10) and (14), the whole process model has to be known (at least elements c_0 to c_3 and d_0 to d_3). As it is usually quite difficult to obtain whole process model, we tried to find the solution by using the concept of multiple integration method [Strejc, 1959].

Consider the process step response as shown in Fig. 5.

Let us define a function $y_1(t)$ as:

$$y_1(t) = \int_0^t \left(1 - y(t)\right)dt \tag{15}$$

A variable A_1 denotes the surface between process input u and process output y, if process steady-state gain equals to 1:

$$A_1 = y_1(\infty) = \int_0^\infty \left(1 - y(t)\right)dt = \lim_{s\to 0}\frac{1}{s}\left(1 - G_P(s)\right) = a_1 - b_1 \tag{16}$$

Some additional information about the process transfer function can be obtained by calculating the surface A_2 (17), shown in Fig. 6.

$$A_2 = y_2(\infty) = \int_0^\infty \left(A_1 - y_1(t)\right)dt = \lim_{s\to 0}\frac{1}{s}\left(A_1 - \frac{1}{s}\left(1 - G_P(s)\right)\right) = b_2 - a_2 + A_1a_1\,, \tag{17}$$

where $y_2(t)$ equals to

$$y_2(t) = \int_0^t \left(A_1 - y_1(t) \right) dt \tag{18}$$

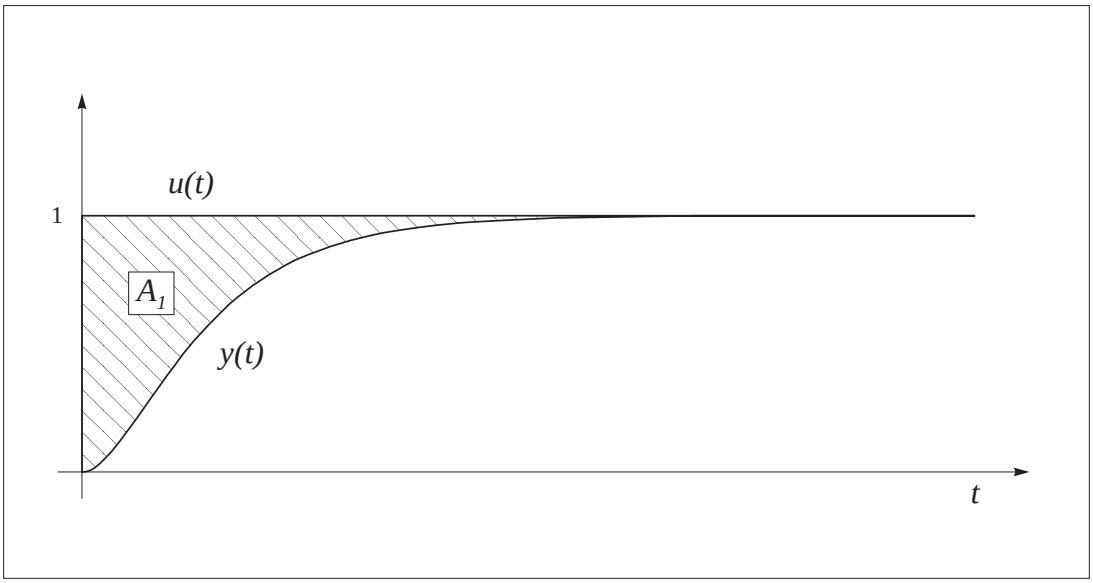


Fig. 5. Typical process response on a unity-gain step function u(t). The area A_1 denotes the surface between u(t) and y(t)

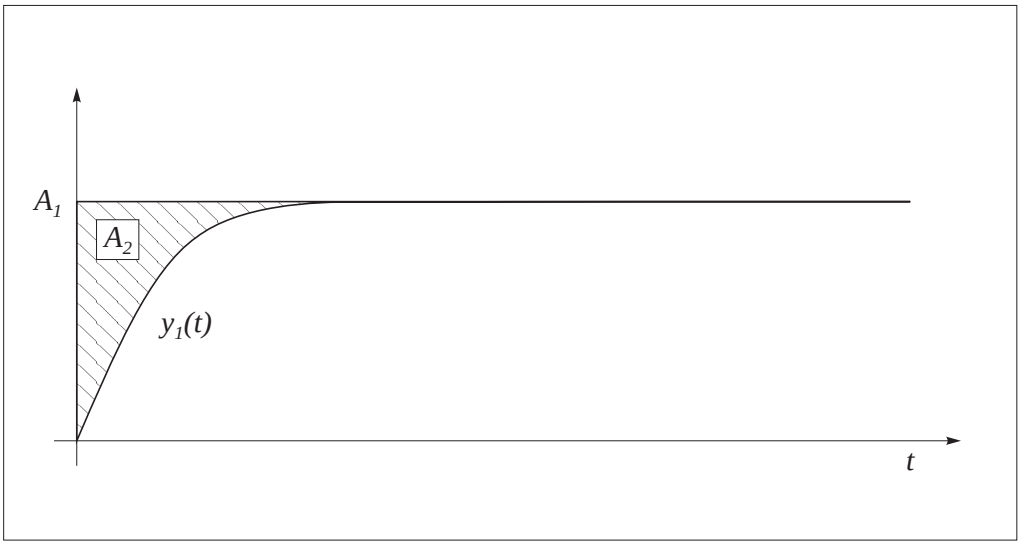


Fig. 6. The area between the A_1 and the $y_1(t)$

The similar can be derived for the third integral:

$$y_3(t) = \int_0^t \left(A_2 - y_2(t) \right) dt \tag{19}$$

$$A_3 = y_3(\infty) = \int_0^\infty \left(A_2 - y_2(t) \right) dt = \lim_{s \to 0} \frac{1}{s} \left(A_2 - \frac{1}{s} \left(A_1 - \frac{1}{s} \left(1 - G_P(s) \right) \right) \right) = a_3 - b_3 + A_2 a_1 - A_1 a_2 , \tag{20}$$

which is shown in Fig. 7.

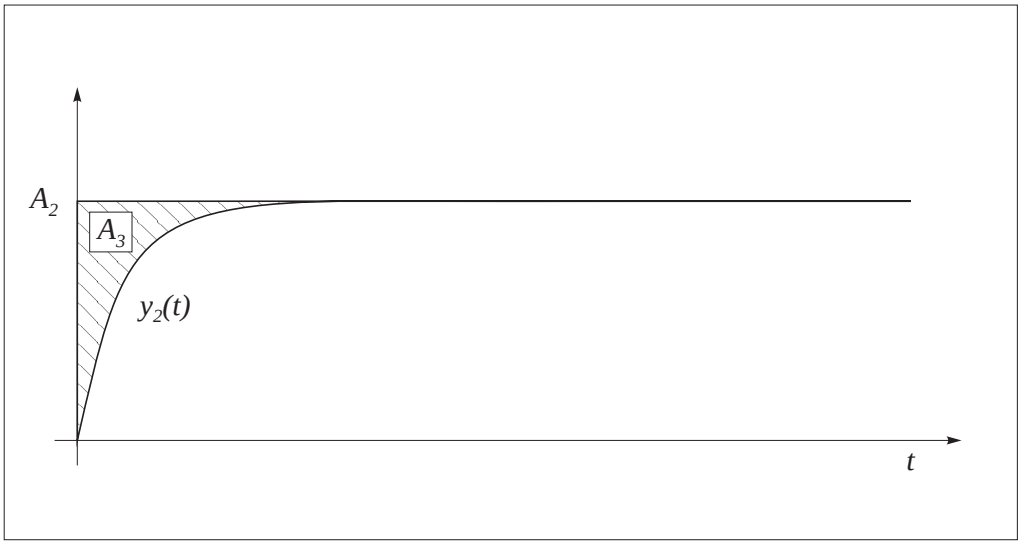


Fig. 7. The area between the A_2 and the $y_2(t)$

Here, we have to be aware of the fact that the values A_1, A_2 and A_3 can be obtained quite *easily* from the process step response. Now, it will be shown that these values can be used in controller parameters calculation. When inserting (7) into (10), we can see that

$$T_i = \frac{a_1 - b_1}{1 + \frac{1}{2K}} . \tag{21}$$

Inserting (16) into (21) lead us to the following expression:

$$T_i = \frac{A_1}{1 + \frac{1}{2K}} . \tag{22}$$

Therefore, controller constants K and T_i are related through the surface A_1 if (4) is satisfied. Note that also the first derivative in (11) equals to 0, as given in (12).

Expression (22) is very important due to its interesting behaviour. If we choose one of the controller constants (e.g. K), the other (T_i) can be calculated by considering expression (22). It will be explained later how process response can be significantly improved by using such approach (e.g. undershoots vanish, etc.).

To calculate both parameters for the PI controller, expression (7) is inserted into (14):

$$KT_i^3\left[a_3 + a_1\left(b_2 + b_1T_i\right) - b_3 - b_2T_i - a_1^2\left(b_1 + T_i\right) + a_1^3 + a_2^2\left(b_1 + T_i\right) - 2a_1a_2\right] = 0 \tag{23}$$

The expression inside the brackets in (23) must equal to 0. The controller integral time constant can be calculated from (23) as

$$\begin{aligned} T_i &= \frac{-a_3 - a_1b_2 + b_3 + a_1^2b_1 - a_1^3 - a_2b_1 + 2a_1a_2}{a_1b_1 - b_2 - a_1^2 + a_2} = \\ &= \frac{-\left[a_3 - b_3 + a_1\left(b_2 - a_2 + a_1\left(a_1 - b_1\right)\right) - a_2\left(a_1 - b_1\right)\right]}{-\left[b_2 - a_2 + a_1\left(a_1 - b_1\right)\right]} \end{aligned} \tag{24}$$

When comparing (24) to (16), (17) and (20), we can see, that the controller integral constant becomes equal to the ratio of two surfaces (integrals):

$$T_i = \frac{A_3}{A_2} \tag{25}$$

Expression (25) therefore gives the solution for the integral time constant T_i, while K can be calculated from (22).

Now, let us imagine that T_i would be tuned from (25) and K would be 1 [Vrančić et al., 1995]. Then

$$\mathbf{Re}\left\{G_C(0)G_P(0)\right\} = -\alpha\,. \tag{26}$$

To scale the Nyquist curve such to satisfy the condition (4), the proportional gain K has to be

$$\boxed{K = \frac{0.5}{\alpha}} \tag{27}$$

Inserting (27) into (22) gives

$$\boxed{T_i = \frac{A_1}{1+\alpha}}. \tag{28}$$

A factor α can be obtained by comparing (25) and (28):

$$\boxed{\alpha = \frac{A_1 A_2}{A_3} - 1} \tag{29}$$

The PI controller tuning procedure can therefore go on as follows:

- measure a process step response
- calculate the surfaces A_1, A_2 and A_3
- calculate factor α from (29)
- calculate controller parameters K and T_i from (27) and (28)

The above procedure is valid if process steady-state gain equals to 1. If this assumption is not true, then the process step response can be rescaled as to give the steady-state gain equal to 1. The actual steady-state gain is saved in variable A_0. The procedure is then the same in all points, except that the controller gain K must be calculated by expression (30) instead of (27).

$$K = \frac{0.5}{\alpha A_0} \tag{30}$$

Note that the introduction of the factor α has only "hystorical" reasons. This factor was used during the optimisation procedure by which controller parameters K and T_i were obtained (see [Vrančić et al., 1995]). Therefore, controller parameters can also be calculated directly from (22) and (25).

4. Experimental results

The tuning procedure was tested on 9 different processes (see Table 1), where one particular process parameter was chosen for each process:

$$G_{P1}(s) = \frac{e^{-s}}{(1+s)} \tag{31}$$

$$G_{P2}(s) = \frac{e^{-s}}{(1+s)^2} \tag{32}$$

$$G_{P3}(s) = \frac{1}{(1+s)^2} \tag{33}$$

$$G_{P4}(s) = \frac{1}{(1+s)^4} \tag{34}$$

$$G_{P5}(s) = \frac{1}{(1+s)^8} \tag{35}$$

$$G_{P6}(s) = \frac{1}{(1+s)(1+0.5s)(1+0.25s)(1+0.125s)} \tag{36}$$

$$G_{P7}(s) = \frac{(1-s)}{(1+s)^3} \tag{37}$$

$$G_{P8}(s) = \frac{e^{-s}(1+0.4s)}{(1+s)^2} \tag{38}$$

$$G_{P9}(s) = \frac{1}{(1+s)(1+s(1+i))(1+s(1-i))} \tag{39}$$

The tuning method was also tested on noise, wrongly estimated end of experiment (integration) time and on process non-linearity.

The used simulation scheme in program package SIMULINK is shown in Fig. 8. A white noise was filtered by the first-order filter with time constant T_{filt}=0.1s at various constants K_{filt} (Fig. 8).

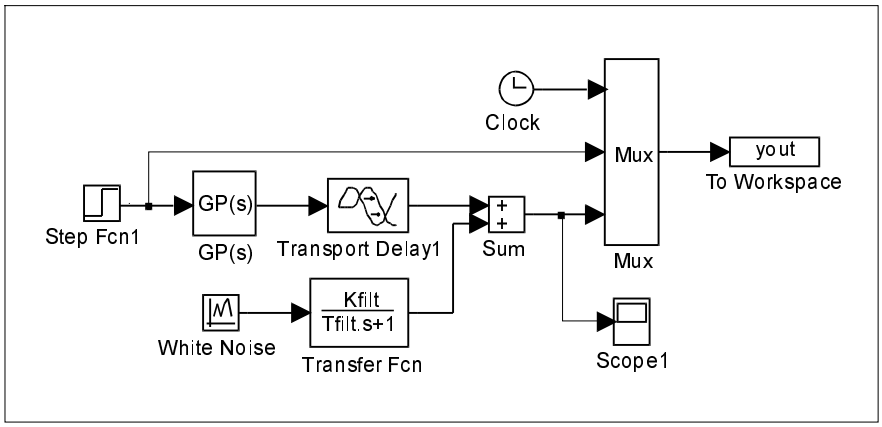


Fig. 8. The simulation scheme in SIMULINK

The MATLAB function STEPTUN.M for calculating areas A_1, A_2 and A_3, and controller parameters K and T_i is given in appendix.

The experimental results for processes $G_{P1}(s)$ to $G_{P9}(s)$ with and without present noise are given in Tables 2 to 10, where K_{filt} was changed from value 0 to 0.3.

K_{filt}	A_0	A_1	A_2	A_3	α	K	T_i
0	1.000	1.999	2.502	2.674	0.871	0.574	1.069
0.1	1.002	2.017	2.601	3.080	0.703	0.710	1.184
0.2	1.005	2.026	2.663	3.342	0.615	0.810	1.255
0.3	1.007	2.039	2.79	3.951	0.44	1.129	1.416

Table 2. Calculated values of areas and controller parameters vs. noise amplitude K_{filt} for the process $G_{P1}(s)$

K_{filt}	A_0	A_1	A_2	A_3	α	K	T_i
0	1.000	3.00	5.504	8.191	1.016	0.492	1.488
0.1	0.998	2.969	5.234	6.954	1.235	0.406	1.329
0.2	0.996	2.952	5.047	6.218	1.397	0.360	1.232
0.3	0.994	2.919	4.665	4.349	2.131	0.236	0.932

Table 3. Calculated values of areas and controller parameters vs. noise amplitude K_{filt} for the process $G_{P2}(s)$

K_{filt}	A_0	A_1	A_2	A_3	α	K	T_i
0	1.00	1.998	2.992	3.963	0.509	0.983	1.325
0.1	1.002	2.015	3.083	4.336	0.433	1.153	1.406
0.2	1.005	2.027	3.172	4.69	0.371	1.34	1.478
0.3	1.007	2.048	3.306	5.304	0.277	1.795	1.604

Table 4. Calculated values of areas and controller parameters vs. noise amplitude K_{filt} for the process $G_{P3}(s)$

K_{filt}	A_0	A_1	A_2	A_3	α	K	T_i
0	1.00	3.997	9.98	19.9	1.005	0.498	1.994
0.1	1.00	3.993	9.869	19.31	1.04	0.481	1.957
0.2	1.001	3.997	9.776	18.72	1.087	0.460	1.915
0.3	1.001	4.013	9.925	19.73	1.018	0.491	1.988

Table 5. Calculated values of areas and controller parameters vs. noise amplitude K_{filt} for the process $G_{P4}(s)$

K_{filt}	A_0	A_1	A_2	A_3	α	K	T_i
0	1.0	7.999	36.0	120.1	1.397	0.358	3.336
0.1	0.999	7.958	35.17	113.2	1.473	0.34	3.218
0.2	0.997	7.948	34.98	112.5	1.472	0.341	3.216
0.3	0.996	7.873	33.52	100.4	1.63	0.308	2.994

Table 6. Calculated values of areas and controller parameters vs. noise amplitude K_{filt} for the process $G_{P5}(s)$

K_{filt}	A_0	A_1	A_2	A_3	α	K	T_i
0	1.0	1.875	2.424	2.729	0.665	0.752	1.126
0.1	1.002	1.891	2.521	3.132	0.522	0.955	1.242
0.2	1.005	1.902	2.611	3.514	0.413	1.204	1.346
0.3	1.007	1.912	2.695	3.894	0.324	1.534	1.445

Table 7. Calculated values of areas and controller parameters vs. noise amplitude K_{filt} for the process $G_{P6}(s)$

K_{filt}	A_0	A_1	A_2	A_3	α	K	T_i
0	1.00	3.998	8.99	15.95	1.253	0.399	1.775
0.1	0.998	3.973	8.713	14.66	1.361	0.368	1.683
0.2	0.996	3.954	8.538	13.96	1.418	0.354	1.635
0.3	0.994	3.915	8.176	12.30	1.603	0.314	1.504

Table 8. Calculated values of areas and controller parameters vs. noise amplitude K_{filt} for the process $G_{P7}(s)$

K_{filt}	A_0	A_1	A_2	A_3	α	K	T_i
0	1.00	2.60	4.301	5.976	0.871	0.574	1.389
0.1	0.998	2.583	4.123	5.17	1.059	0.473	1.254
0.2	0.996	2.578	4.075	5.052	1.079	0.465	1.24
0.3	0.994	2.552	3.838	4.04	1.424	0.353	1.053

Table 9. Calculated values of areas and controller parameters vs. noise amplitude K_{filt} for the process $G_{P8}(s)$

K_{filt}	A_0	A_1	A_2	A_3	α	K	T_i
0	1.00	3.00	5.012	5.072	1.964	0.255	1.012
0.1	1.004	3.053	5.433	7.318	1.267	0.393	1.347
0.2	1.007	3.096	5.865	10.41	0.744	0.667	1.776
0.3	1.011	3.129	5.968	10.37	0.80	0.618	1.738

Table 10. Calculated values of areas and controller parameters vs. noise amplitude K_{filt} for the process $G_{P9}(s)$

It can be seen that some processes are more and some are less sensitive to noise. The most sensitive to noise are faster processes, as $G_{P1}(s)$, $G_{P2}(s)$, $G_{P3}(s)$, $G_{P6}(s)$ and $G_{P9}(s)$. As it will be outlined later, the resulted calculated parameters at the highest level of noise are resulting in still quite good closed-loop time response. Moreover, in next chapter it will be shown that much better estimations of K and T_i can be obtained when using another approach for noisy processes.

The next we tested the new algorithm sensitivity on the wrongly estimated end of experiment time. The results are shown in Tables 11 to 19. The first row denotes the original step-response experimental time and other rows denotes results when the experimental time is shorten. T_{fin} denotes the experimental time in seconds. It can be seen that the new algorithm is not too sensitive on the improper estimation of the experiment time. Moreover, if process is noisy, it is recommended to decrease the measurement time of step response, what will be performed in next chapter.

T_{fin}	A_0	A_1	A_2	A_3	α	K	T_i
12	1.000	1.999	2.502	2.674	0.871	0.574	1.069
8	0.999	1.99	2.45	2.51	0.941	0.532	1.025
6	0.991	1.955	2.302	2.17	1.073	0.470	0.943
4	0.939	1.813	1.863	1.433	1.358	0.392	0.769

Table 11. Calculated values of areas and controller parameters vs. experimental time T_{fin} [s] for the process $G_{P1}(s)$

T_{fin}	A_0	A_1	A_2	A_3	α	K	T_i
15	1.000	3.00	5.504	8.191	1.016	0.492	1.488
12	1.00	3.00	5.475	8.06	1.04	0.483	1.472
10	0.998	2.984	5.391	7.75	1.076	0.466	1.437
8	0.99	2.936	5.126	6.942	1.168	0.433	1.354

Table 12. Calculated values of areas and controller parameters vs. experimental time T_{fin} [s] for the process $G_{P2}(s)$

T_{fin}	A_0	A_1	A_2	A_3	α	K	T_i
12	1.00	1.998	2.992	3.963	0.509	0.983	1.325
10	0.999	1.993	2.955	3.833	0.537	0.933	1.297
8	0.996	1.969	2.837	3.484	0.603	0.832	1.228
6	0.977	1.885	2.497	2.703	0.741	0.69	1.082

Table 13. Calculated values of areas and controller parameters vs. experimental time T_{fin} [s] for the process $G_{P3}(s)$

T_{fin}	A_0	A_1	A_2	A_3	α	K	T_i
16	1.00	3.997	9.98	19.9	1.005	0.498	1.994
14	0.999	3.99	9.912	19.54	1.024	0.489	1.972
12	0.996	3.965	9.711	18.65	1.064	0.472	1.921
10	0.985	3.889	9.188	16.68	1.142	0.445	1.815

Table 14. Calculated values of areas and controller parameters vs. experimental time T_{fin} [s] for the process $G_{P4}(s)$

T_{fin}	A_0	A_1	A_2	A_3	α	K	T_i
25	1.0	7.999	36.0	120.1	1.397	0.358	3.336
20	0.998	7.977	35.68	117.6	1.421	0.353	3.295
18	0.995	7.936	35.15	114.0	1.448	0.347	3.243
15	0.972	7.75	33.06	101.6	1.523	0.338	3.072

Table 15. Calculated values of areas and controller parameters vs. experimental time T_{fin} [s] for the process $G_{P5}(s)$

T_{fin}	A_0	A_1	A_2	A_3	α	K	T_i
12	1.0	1.875	2.424	2.729	0.665	0.752	1.126
8	0.998	1.864	2.365	2.549	0.729	0.687	1.078
6	0.990	1.823	2.199	2.169	0.848	0.596	0.986
5	0.974	1.767	2.011	1.82	0.952	0.540	0.905

Table 16. Calculated values of areas and controller parameters vs. experimental time T_{fin} [s] for the process $G_{P6}(s)$

T_{fin}	A_0	A_1	A_2	A_3	α	K	T_i
15	1.00	3.998	8.99	15.95	1.253	0.399	1.775
12	0.998	3.985	8.875	15.43	1.292	0.388	1.739
10	0.992	3.945	8.593	14.36	1.362	0.371	1.671
8	0.966	3.823	7.865	12.07	1.492	0.347	1.534

Table 17. Calculated values of areas and controller parameters vs. experimental time T_{fin} [s] for the process $G_{P7}(s)$

T_{fin}	A_0	A_1	A_2	A_3	α	K	T_i
14	1.00	2.60	4.301	5.976	0.871	0.574	1.389
10	0.999	2.589	4.23	5.701	0.921	0.544	1.348
8	0.993	2.556	4.05	5.16	1.006	0.501	1.274
7	0.984	2.514	3.857	4.666	1.078	0.471	1.210

Table 18. Calculated values of areas and controller parameters vs. experimental time T_{fin} [s] for the process $G_{P8}(s)$

T_{fin}	A_0	A_1	A_2	A_3	α	K	T_i
20	1.00	3.00	5.012	5.072	1.964	0.255	1.012
16	0.999	2.987	4.882	4.335	2.364	0.212	0.888
14	0.999	2.989	4.905	4.465	2.284	0.219	0.910
12	1.004	3.034	5.24	5.902	1.694	0.294	1.126

Table 19. Calculated values of areas and controller parameters vs. experimental time T_{fin} [s] for the process $G_{P9}(s)$

Finally, the chosen processes were also tested on the non-linearity. The non-linearity:

$$f(u) = u - K_{nl}\,\mathbf{sin}(\pi u) \tag{40}$$

was added on the process output. The variable *u* denotes input value and *f(u)* is the output of the non-linearity. The values from K_{nl}=0.05 to K_{nl}=0.15 were tested. The non-linearity (40) at K_{nl}=0.15 is shown in Fig. 9.

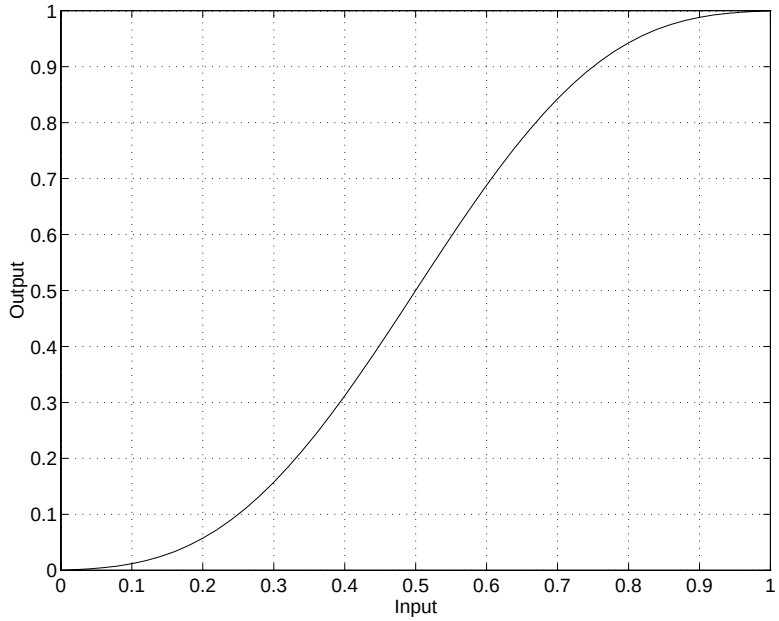


Fig. 9. The non-linear function (40) @ K_{nl}=0.15

The following scheme in SIMULINK was used for testing the process non-linearity:

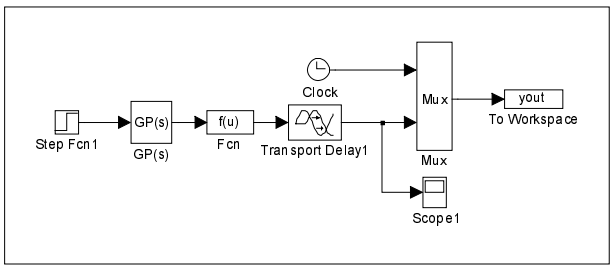


Fig. 10. The simulation scheme in SIMULINK

The function *f(u)* denotes the process non-linearity (40). Tables 20 to 28 gives us the calculated values of areas and controller parameters vs. parameter K_{nl}. It is shown, that the results differs from ones obtained by the linear process, but the differences are not so crucial, that will be proved later from the closed-loop time response results.

K_{nl}	A_0	A_1	A_2	A_3	α	K	T_i
0	1.000	1.999	2.502	2.674	0.871	0.574	1.069
0.05	1.00	1.929	2.242	2.19	0.974	0.513	0.977
0.10	1.00	1.858	1.982	1.706	1.158	0.432	0.861
0.15	1.00	1.787	1.722	1.222	1.518	0.33	0.71

Table 20. Calculated values of areas and controller parameters vs. non-linearity K_{nl} for the process $G_{P1}(s)$

K_{nl}	A_0	A_1	A_2	A_3	α	K	T_i
0	1.000	3.00	5.504	8.191	1.016	0.492	1.488
0.05	1.00	2.925	5.05	6.94	1.127	0.444	1.376
0.10	1.00	2.851	4.588	5.689	1.299	0.385	1.240
0.15	1.00	2.776	4.129	4.438	1.583	0.316	1.075

Table 21. Calculated values of areas and controller parameters vs. non-linearity K_{nl} for the process $G_{P2}(s)$

K_{nl}	A_0	A_1	A_2	A_3	α	K	T_i
0	1.00	1.998	2.992	3.963	0.509	0.983	1.325
0.05	1.00	1.924	2.611	3.145	0.598	0.837	1.205
0.10	1.00	1.85	2.23	2.327	0.773	0.647	1.044
0.15	1.00	1.776	1.849	1.51	1.175	0.425	0.816

Table 22. Calculated values of areas and controller parameters vs. non-linearity K_{nl} for the process $G_{P3}(s)$

K_{nl}	A_0	A_1	A_2	A_3	α	K	T_i
0	1.00	3.997	9.98	19.9	1.005	0.498	1.994
0.05	1.00	3.925	9.217	17.03	1.124	0.445	1.848
0.10	1.00	3.846	8.45	14.17	1.293	0.387	1.677
0.15	1.00	3.771	7.685	11.31	1.562	0.32	1.472

Table 23. Calculated values of areas and controller parameters vs. non-linearity K_{nl} for the process $G_{P4}(s)$

K_{nl}	A_0	A_1	A_2	A_3	α	K	T_i
0	1.0	7.999	36.0	120.1	1.397	0.358	3.336
0.05	1.00	7.922	34.45	109.4	1.496	0.334	3.174
0.10	1.00	7.845	32.91	98.62	1.618	0.309	2.997
0.15	1.00	7.769	31.36	87.89	1.772	0.282	2.802

Table 24. Calculated values of areas and controller parameters vs. non-linearity K_{nl} for the process $G_{P5}(s)$

K_{nl}	A_0	A_1	A_2	A_3	α	K	T_i
0	1.0	1.875	2.424	2.729	0.665	0.752	1.126
0.05	1.00	1.815	2.153	2.207	0.771	0.649	1.025
0.10	1.00	1.755	1.883	1.686	0.961	0.520	0.895
0.15	1.00	1.696	1.613	1.165	1.349	0.371	0.722

Table 25. Calculated values of areas and controller parameters vs. non-linearity K_{nl} for the process $G_{P6}(s)$

K_{nl}	A_0	A_1	A_2	A_3	α	K	T_i
0	1.00	3.998	8.99	15.95	1.253	0.399	1.775
0.05	1.00	3.88	8.32	13.80	1.339	0.374	1.659
0.10	1.00	3.762	7.653	11.66	1.469	0.340	1.524
0.15	1.00	3.644	6.986	9.517	1.675	0.299	1.362

Table 26. Calculated values of areas and controller parameters vs. non-linearity K_{nl} for the process $G_{P7}(s)$

K_{nl}	A_0	A_1	A_2	A_3	α	K	T_i
0	1.00	2.60	4.301	5.976	0.871	0.574	1.389
0.05	1.00	2.519	3.879	4.945	0.977	0.512	1.275
0.10	1.00	2.439	3.458	3.913	1.155	0.433	1.132
0.15	1.00	2.359	3.036	2.882	1.485	0.337	0.949

Table 27. Calculated values of areas and controller parameters vs. non-linearity K_{nl} for the process $G_{P8}(s)$

K_{nl}	A_0	A_1	A_2	A_3	α	K	T_i
0	1.00	3.00	5.012	5.072	1.964	0.255	1.012
0.05	1.00	3.005	4.96	5.214	1.858	0.269	1.051
0.10	1.00	3.010	4.91	5.363	1.755	0.285	1.092
0.15	1.00	3.015	4.859	5.512	1.658	0.302	1.134

Table 28. Calculated values of areas and controller parameters vs. non-linearity K_{nl} for the process $G_{P9}(s)$

Now, the open-loop and closed-loop time responses will be shown for all 9 processes ((31) to (39)). Controller settings will be based on the original step responses, the noisy step responses (at K_{filt}=0.3), the shorten step responses (the shortest in Tables 11 to 19) and the non-linear step responses (at K_{nl}=0.15). The shown time-responses are those marked by the shadowed rows in Tables 2 to 28.

Several closed-loop experiments were performed, where tracking and control performances were being tested. The set-point changed from 0 to 1 at time origin, and the disturbance unity step change at process input appeared at the half-time of each experiment (see Fig. 1).

Fig. 11 and 12 show simulation schemes, built in the program package SIMULINK, adapted for testing the closed-loop operation for the linear and noiseless processes, processes with the present noise, and the non-linear processes.

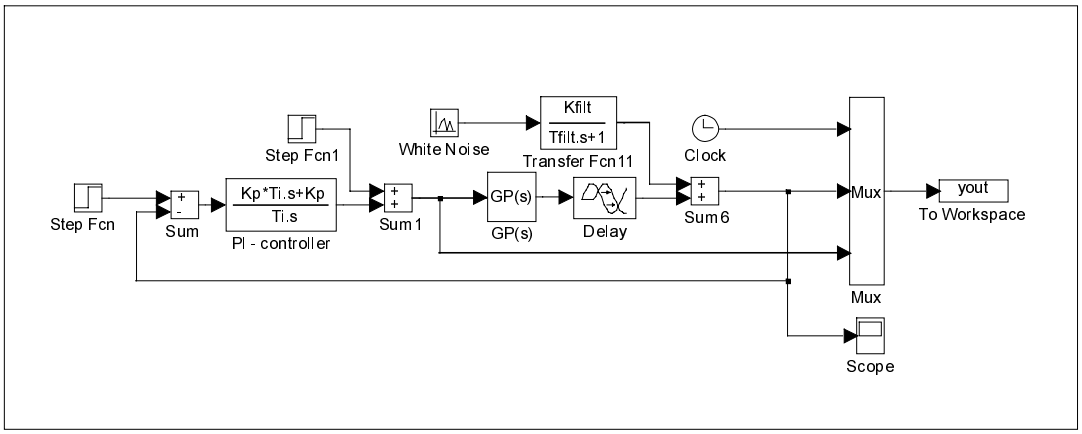


Fig. 11. The closed-loop scheme in SIMULINK for testing the processes with and without noise

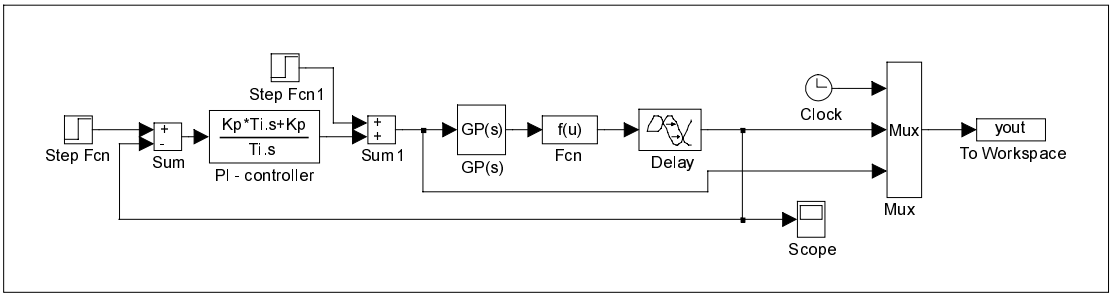


Fig. 12. The closed-loop scheme in SIMULINK for testing the non-linear processes

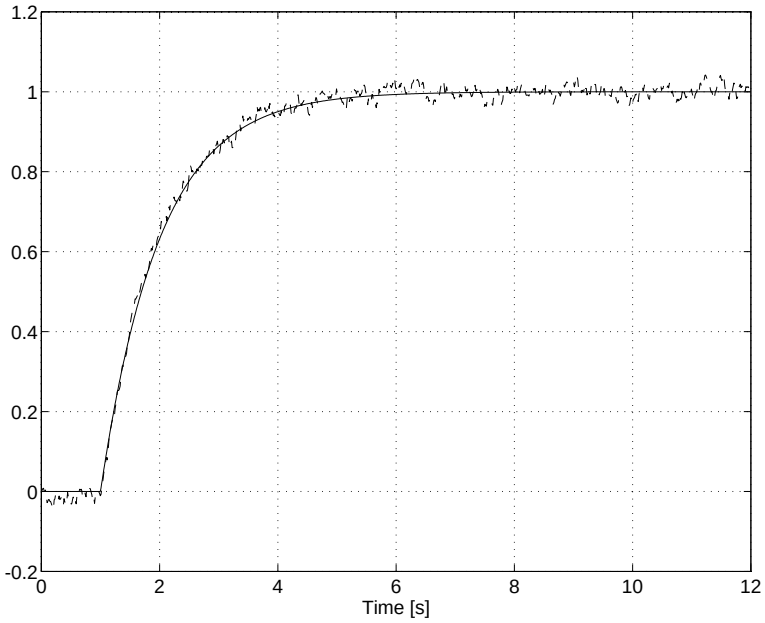


Fig. 13. The open-loop response of the process $G_{P1}(s)$;
__ noiseless response, -- response with added noise (K_{filt} = 0.3)

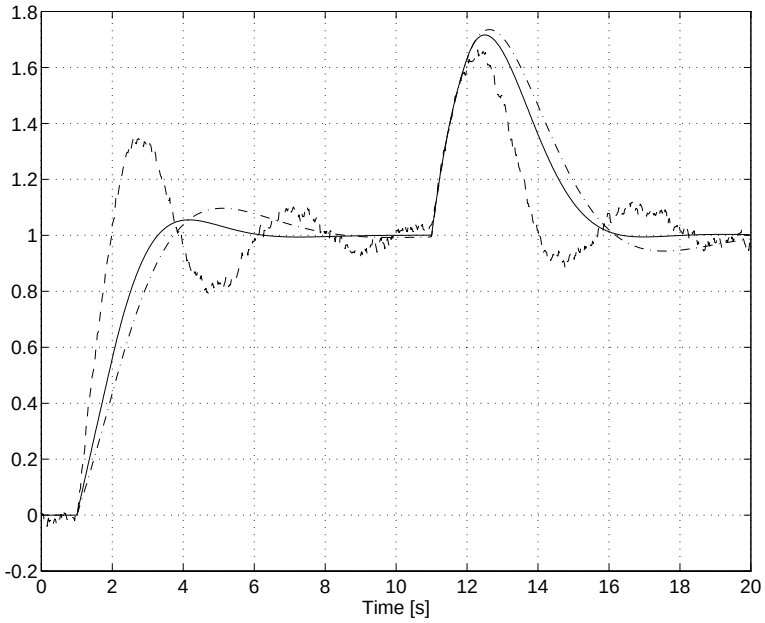


Fig. 14. The closed-loop response of the process $G_{P1}(s)$;
__ controller parameters calculation based on the noiseless response,
-- controller parameters calculation based on response with added noise (K_{filt} = 0.3),
-.- controller parameters calculation based on the shorten step response (T_{fin} = 4s)

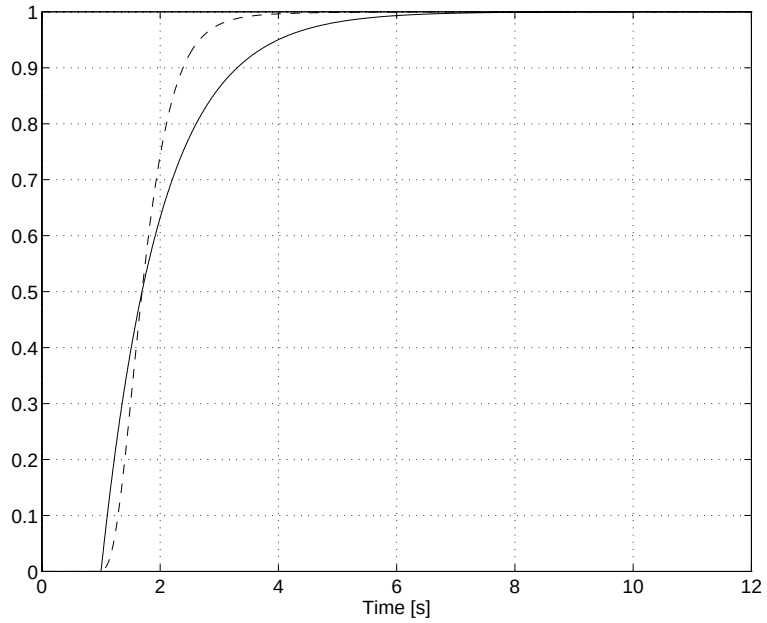


Fig. 15. The open-loop response of the process $G_{P1}(s)$;
__ linear process (K_{nl} = 0), -- non-linear process (K_{nl} = 0.15)

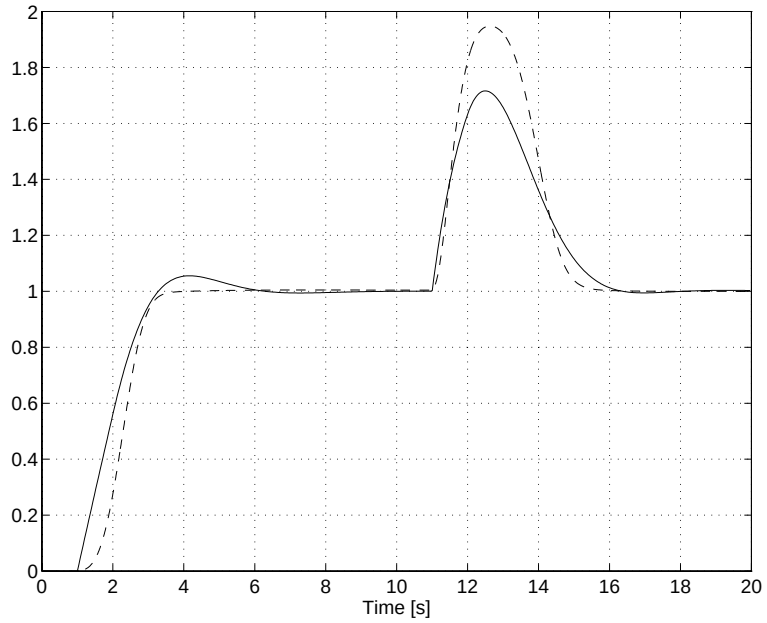


Fig. 16. The closed-loop response of the process $G_{P1}(s)$;
__ controller parameters calculation based on the linear response,
-- controller parameters calculation based on the non-linear response (K_{nl} = 0.15)

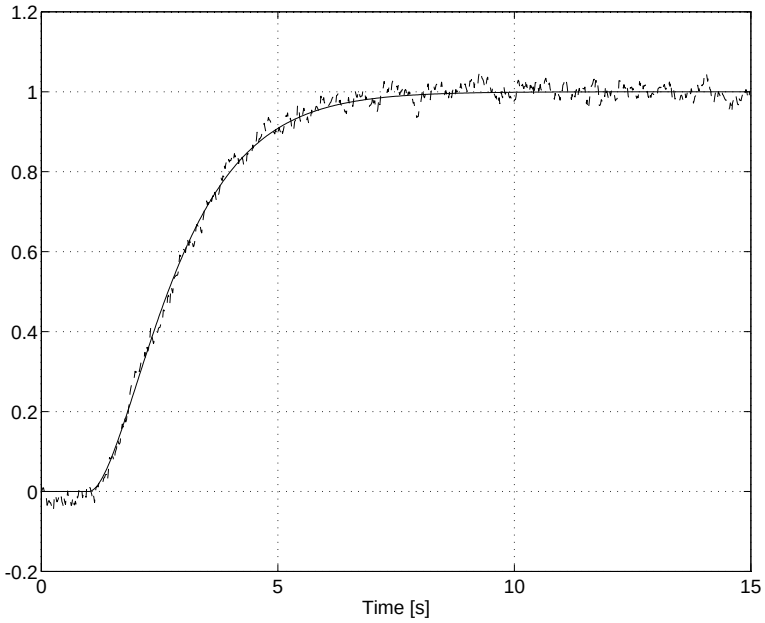


Fig. 17. The open-loop response of the process $G_{P2}(s)$;
__ noiseless response, -- response with added noise (K_{filt} = 0.3)

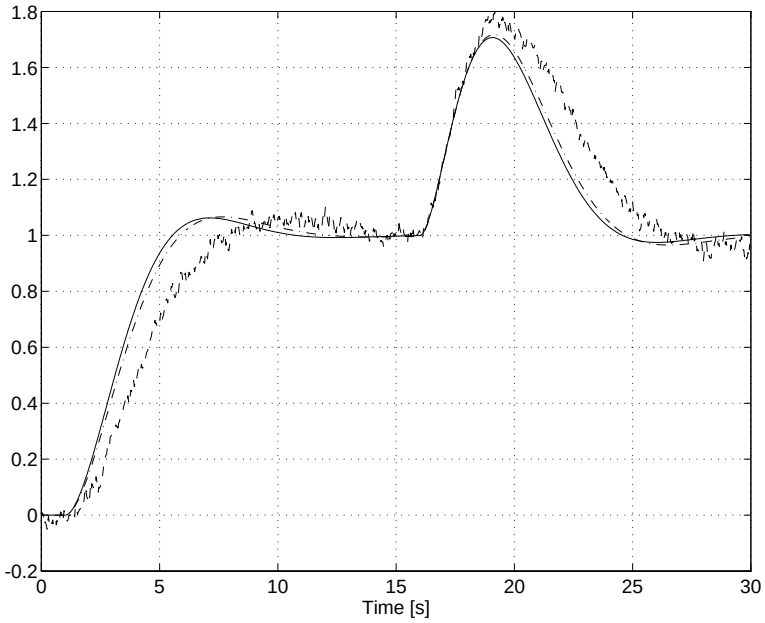


Fig. 18. The closed-loop response of the process $G_{P2}(s)$;
__ controller parameters calculation based on the noiseless response,
-- controller parameters calculation based on response with added noise (K_{filt} = 0.3),
-.- controller parameters calculation based on the shorten step response (T_{fin} = 8s)

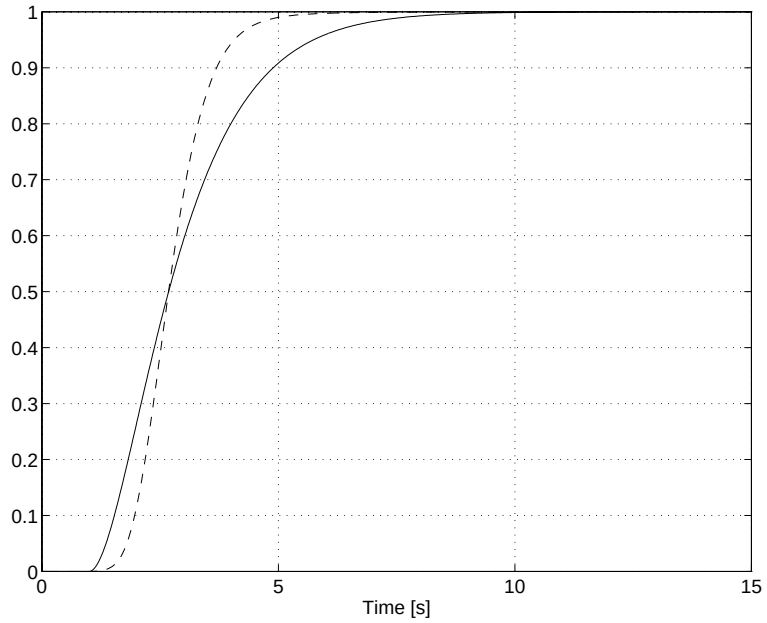


Fig. 19. The open-loop response of the process $G_{P2}(s)$;
__ linear process (K_{nl} = 0), -- non-linear process (K_{nl} = 0.15)

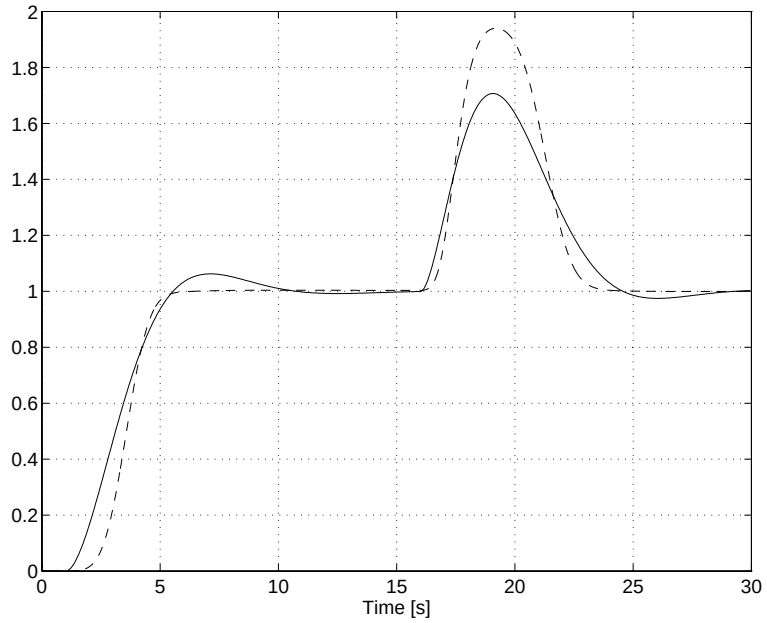


Fig. 20. The closed-loop response of the process $G_{P2}(s)$;
__ controller parameters calculation based on the linear response,
-- controller parameters calculation based on the non-linear response (K_{nl} = 0.15)

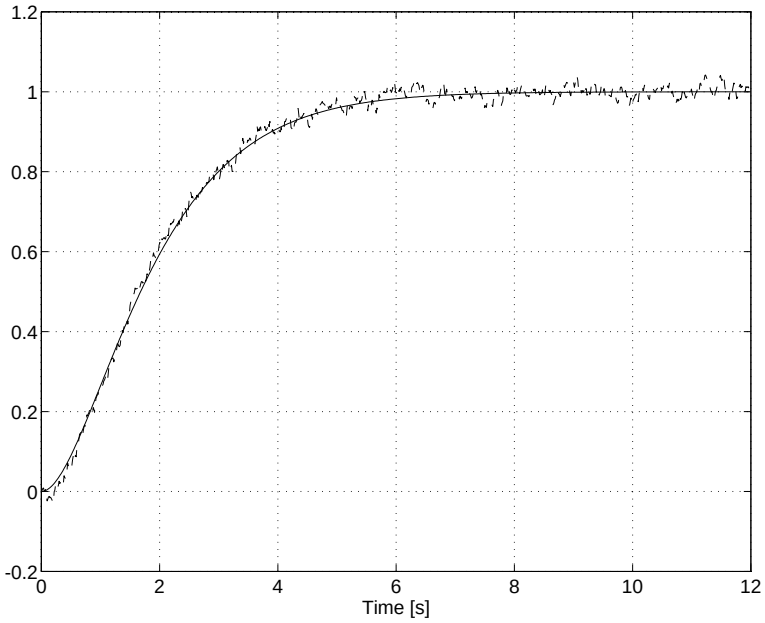


Fig. 21. The open-loop response of the process $G_{P3}(s)$;
__ noiseless response, -- response with added noise (K_{filt} = 0.3)

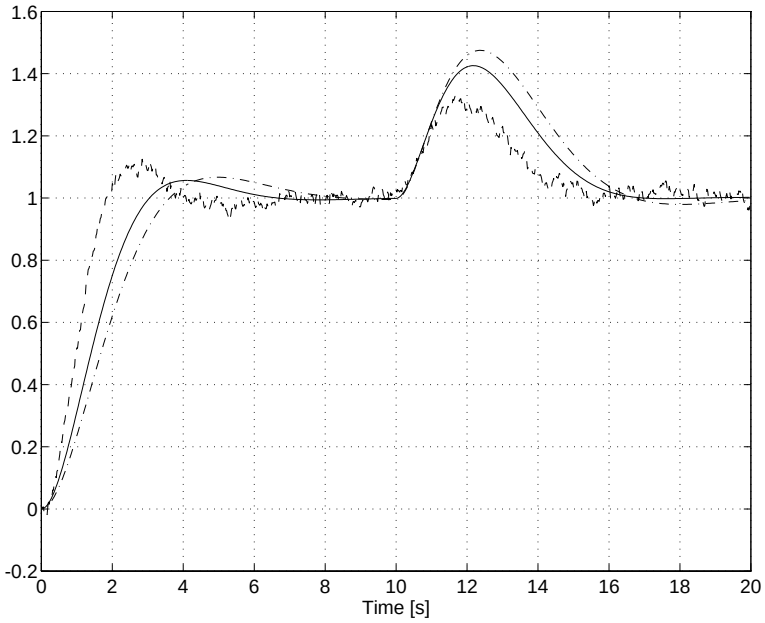


Fig. 22. The closed-loop response of the process $G_{P3}(s)$;
__ controller parameters calculation based on the noiseless response,
-- controller parameters calculation based on response with added noise (K_{filt} = 0.3),
-.- controller parameters calculation based on the shorten step response (T_{fin} = 6s)

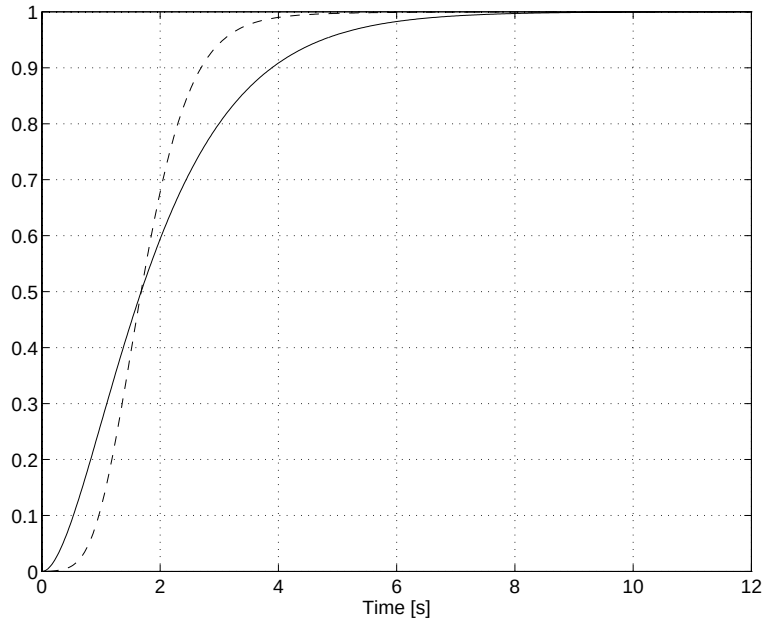


Fig. 23. The open-loop response of the process $G_{P3}(s)$;
__ linear process (K_{nl} = 0), -- non-linear process (K_{nl} = 0.15)

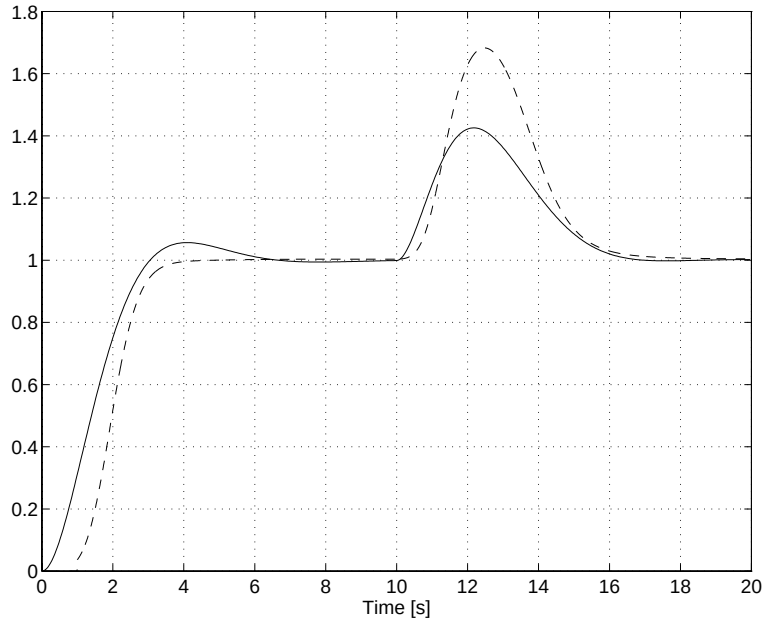


Fig. 24. The closed-loop response of the process $G_{P3}(s)$;
__ controller parameters calculation based on the linear response,
-- controller parameters calculation based on the non-linear response (K_{nl} = 0.15)

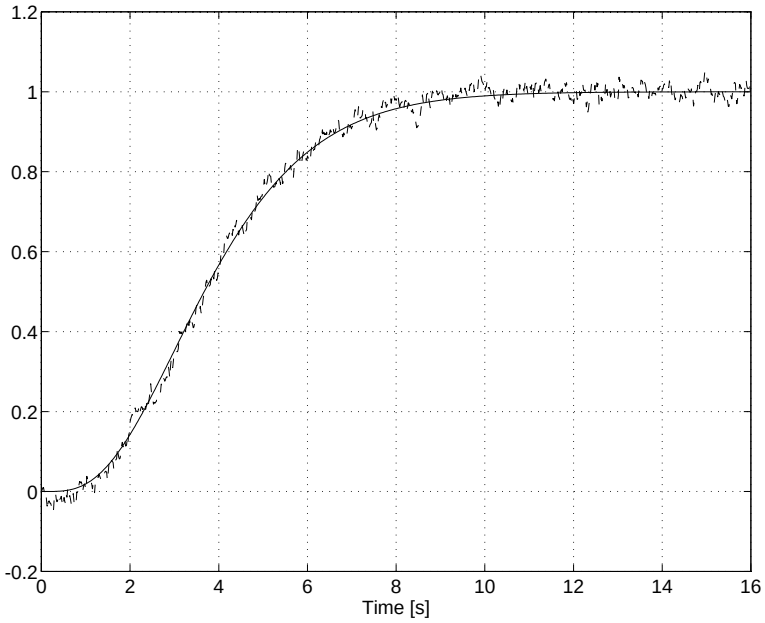


Fig. 25. The open-loop response of the process $G_{P4}(s)$;
__ noiseless response, -- response with added noise (K_{filt} = 0.3)

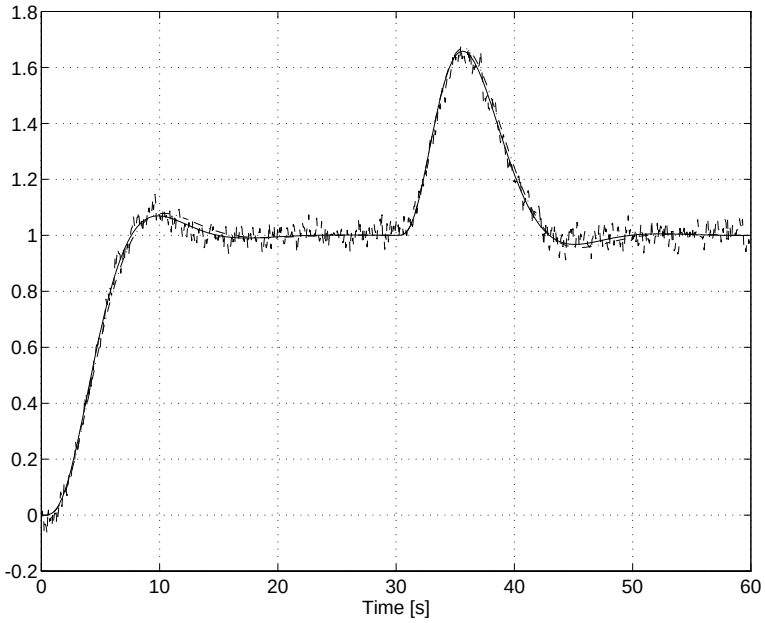


Fig. 26. The closed-loop response of the process $G_{P4}(s)$;
__ controller parameters calculation based on the noiseless response,
-- controller parameters calculation based on response with added noise (K_{filt} = 0.3),
-.- controller parameters calculation based on the shorten step response (T_{fin} = 10s)

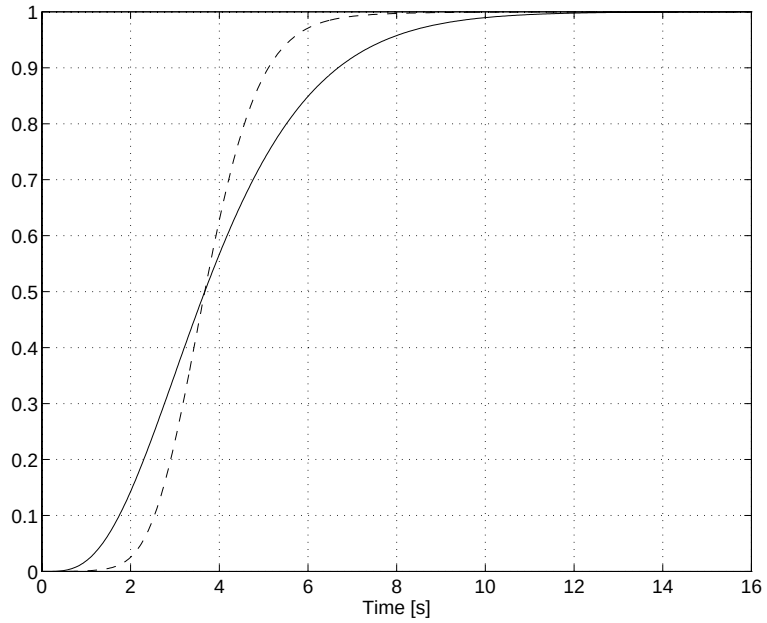


Fig. 27. The open-loop response of the process $G_{P4}(s)$;
__ linear process (K_{nl} = 0), -- non-linear process (K_{nl} = 0.15)

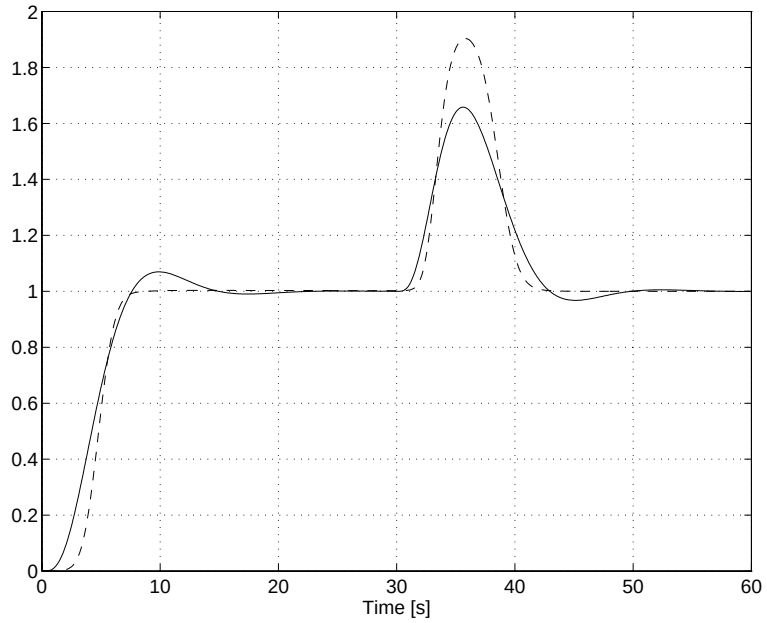


Fig. 28. The closed-loop response of the process $G_{P4}(s)$;
__ controller parameters calculation based on the linear response,
-- controller parameters calculation based on the non-linear response (K_{nl} = 0.15)

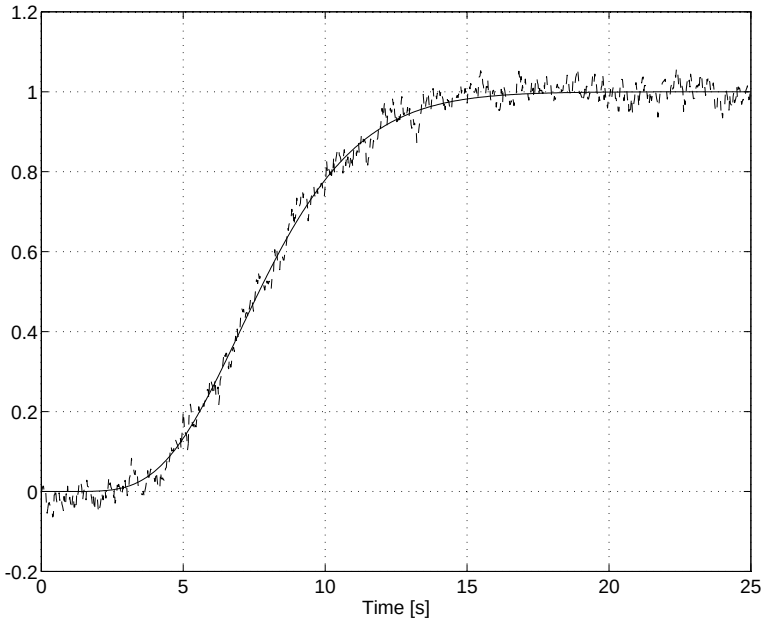


Fig. 29. The open-loop response of the process $G_{P5}(s)$;
__ noiseless response, -- response with added noise (K_{filt} = 0.3)

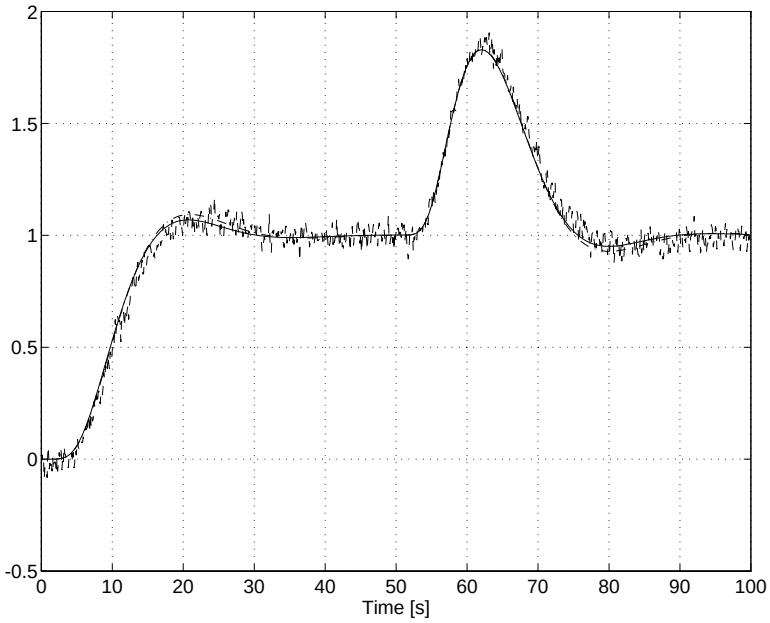


Fig. 30. The closed-loop response of the process $G_{P5}(s)$;
__ controller parameters calculation based on the noiseless response,
-- controller parameters calculation based on response with added noise (K_{filt} = 0.3),
-.- controller parameters calculation based on the shorten step response (T_{fin} = 15s)

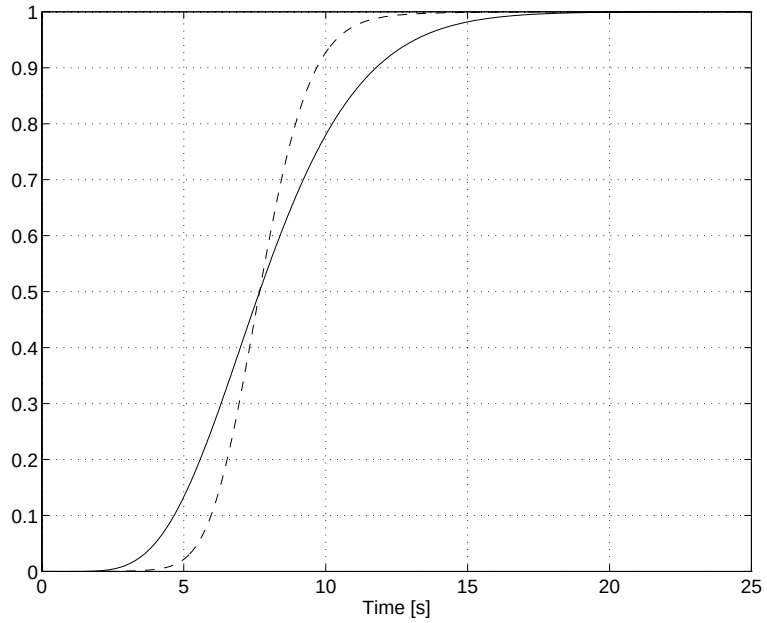


Fig. 31. The open-loop response of the process $G_{P5}(s)$;
__ linear process (K_{nl} = 0), -- non-linear process (K_{nl} = 0.15)

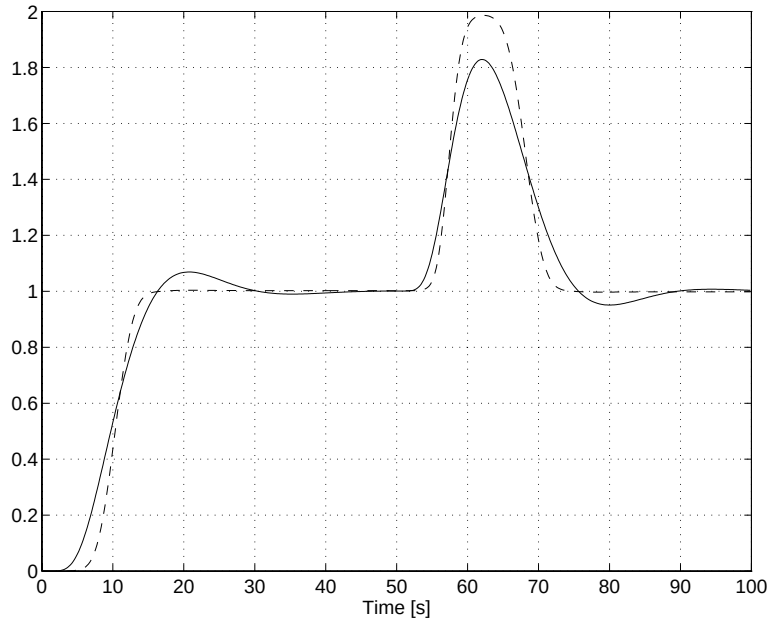


Fig. 32. The closed-loop response of the process $G_{P5}(s)$;
__ controller parameters calculation based on the linear response,
-- controller parameters calculation based on the non-linear response (K_{nl} = 0.15)

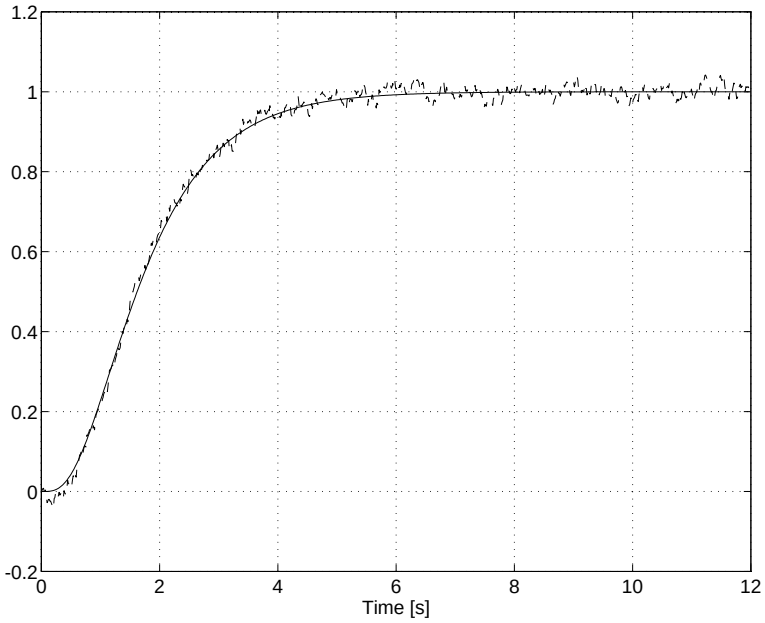


Fig. 33. The open-loop response of the process $G_{P6}(s)$;
__ noiseless response, -- response with added noise (K_{filt} = 0.3)

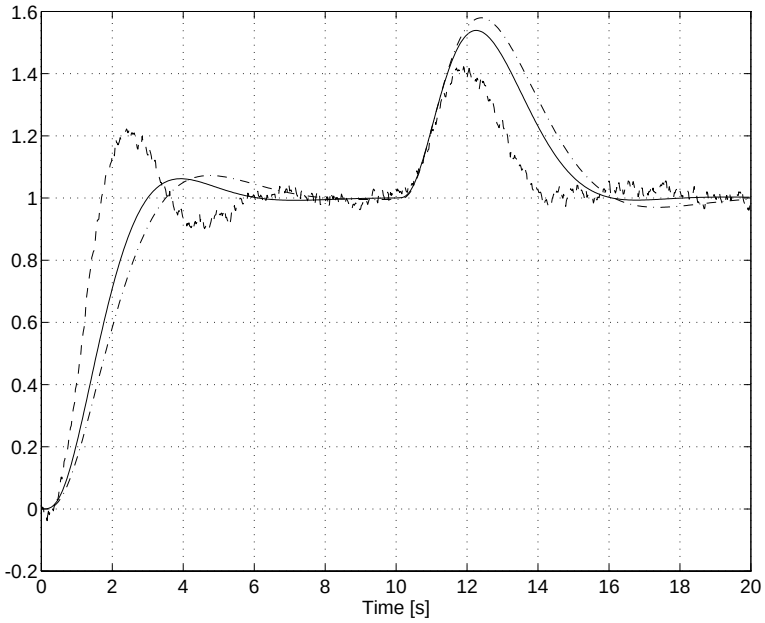


Fig. 34. The closed-loop response of the process $G_{P6}(s)$;
__ controller parameters calculation based on the noiseless response,
-- controller parameters calculation based on response with added noise (K_{filt} = 0.3),
-.- controller parameters calculation based on the shorten step response (T_{fin} = 5s)

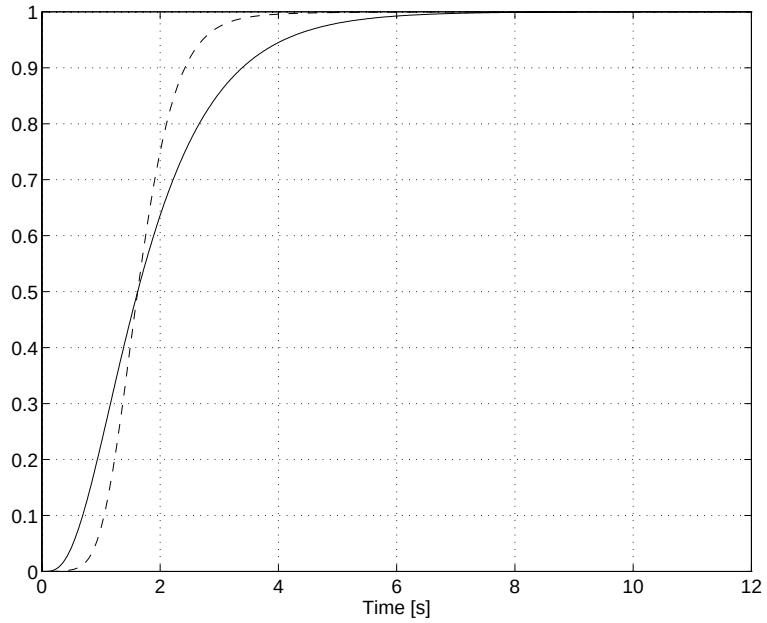


Fig. 35. The open-loop response of the process $G_{P6}(s)$;
__ linear process (K_{nl} = 0), -- non-linear process (K_{nl} = 0.15)

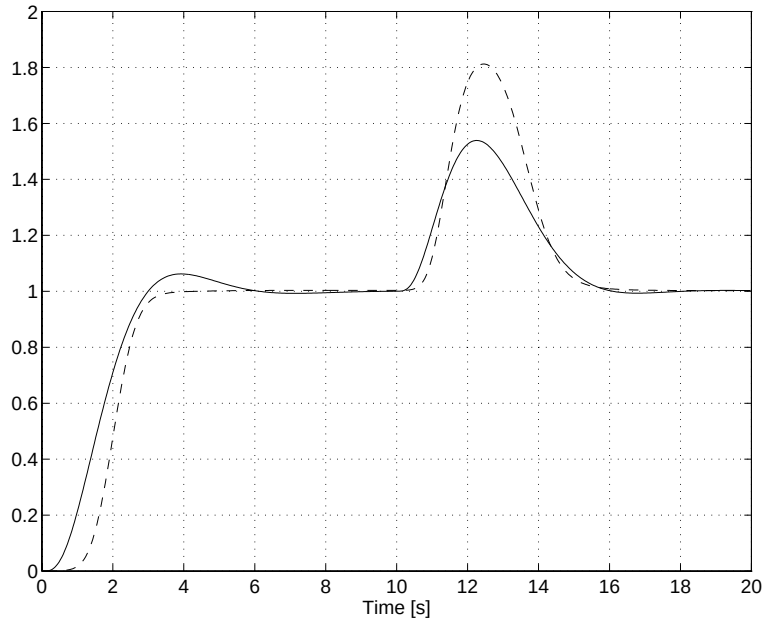


Fig. 36. The closed-loop response of the process $G_{P6}(s)$;
__ controller parameters calculation based on the linear response,
-- controller parameters calculation based on the non-linear response (K_{nl} = 0.15)

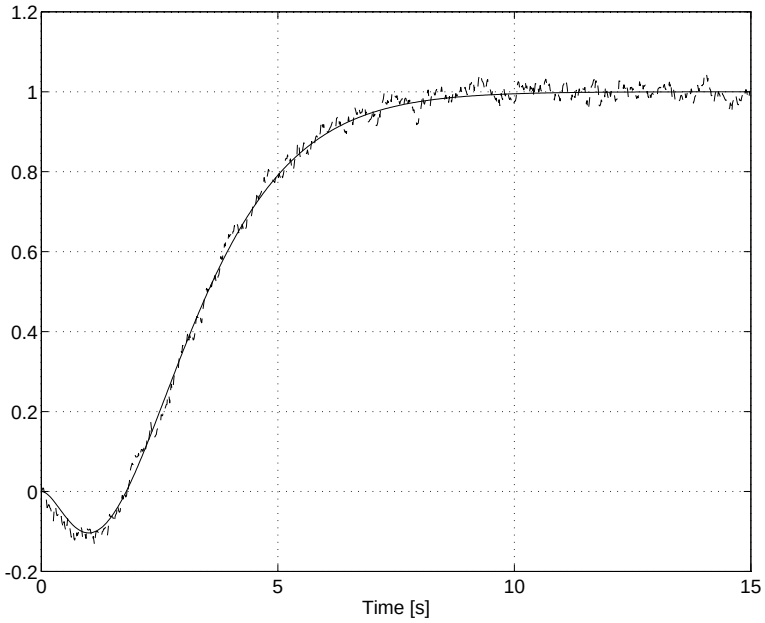


Fig. 37. The open-loop response of the process $G_{P7}(s)$;
__ noiseless response, -- response with added noise (K_{filt} = 0.3)

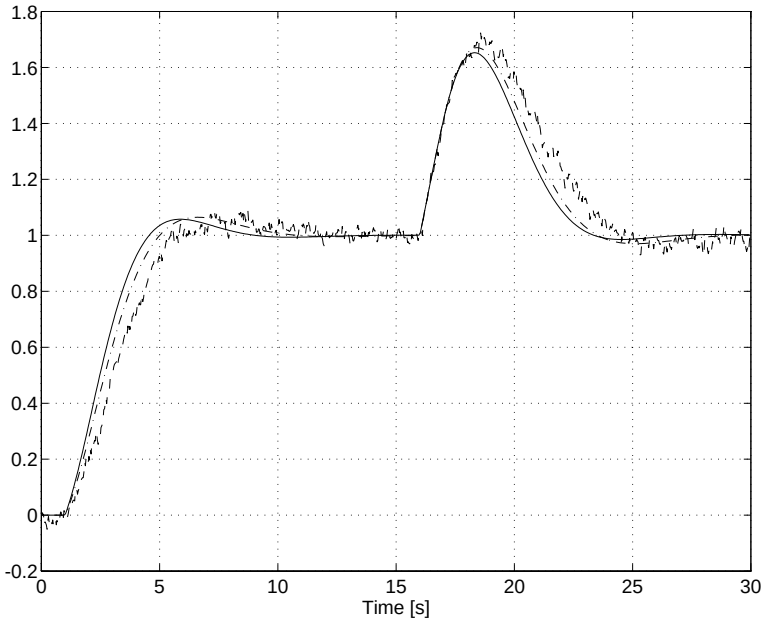


Fig. 38. The closed-loop response of the process $G_{P7}(s)$;
__ controller parameters calculation based on the noiseless response,
-- controller parameters calculation based on response with added noise (K_{filt} = 0.3),
-.- controller parameters calculation based on the shorten step response (T_{fin} = 8s)

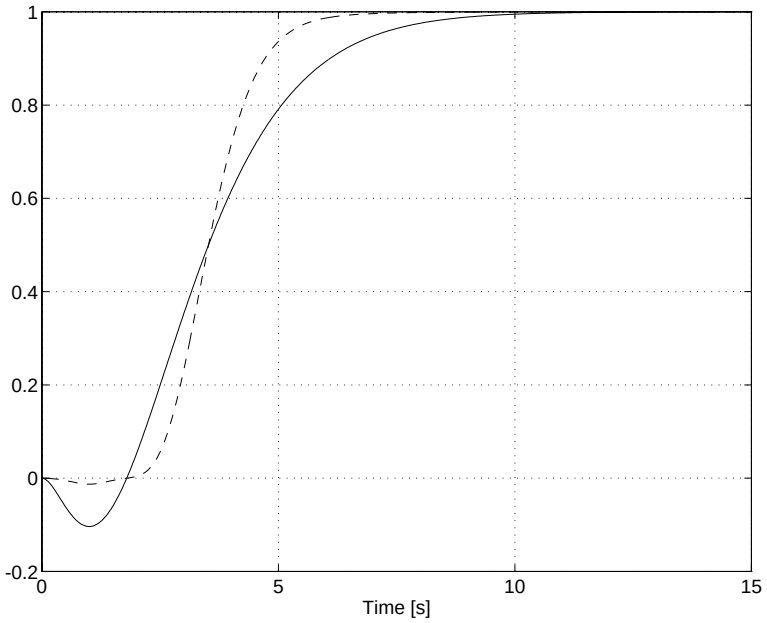


Fig. 39. The open-loop response of the process $G_{P7}(s)$;
__ linear process (K_{nl} = 0), -- non-linear process (K_{nl} = 0.15)

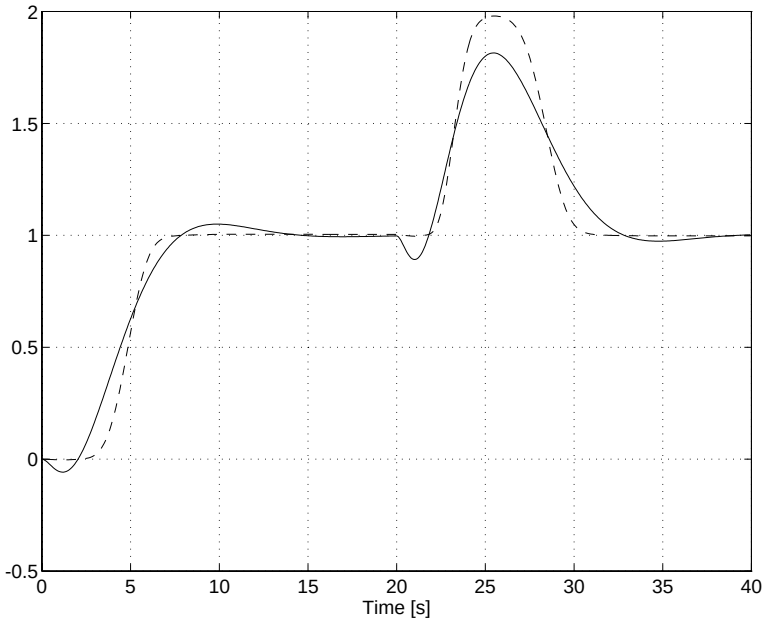


Fig. 40. The closed-loop response of the process $G_{P7}(s)$;
__ controller parameters calculation based on the linear response,
-- controller parameters calculation based on the non-linear response (K_{nl} = 0.15)

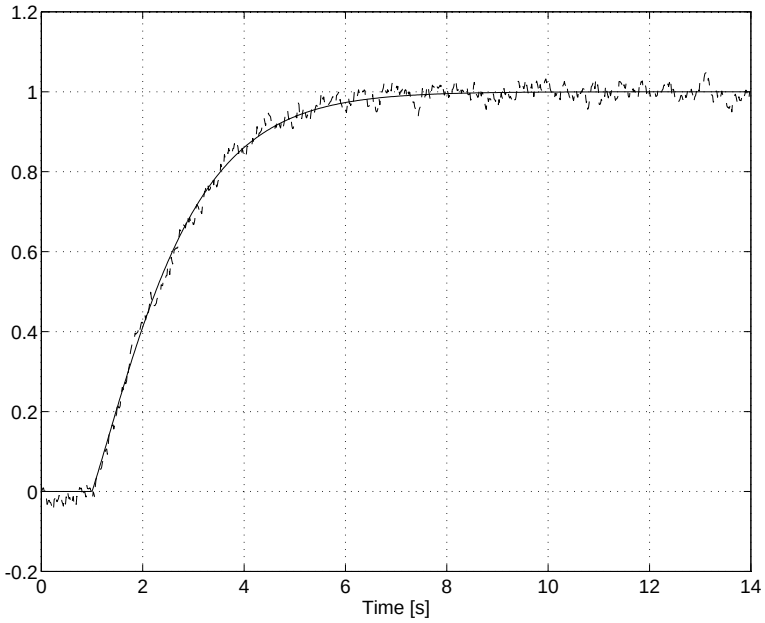


Fig. 41. The open-loop response of the process $G_{P8}(s)$;
__ noiseless response, -- response with added noise (K_{filt} = 0.3)

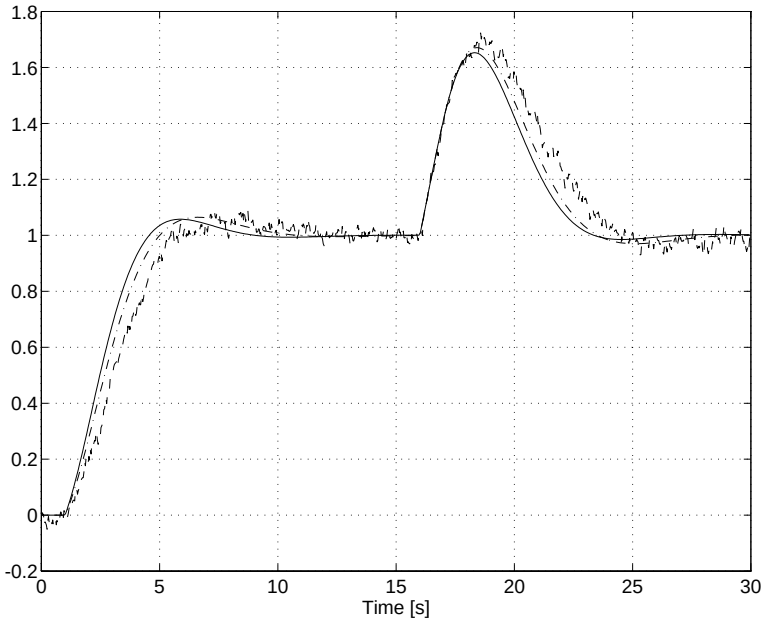


Fig. 42. The closed-loop response of the process $G_{P8}(s)$;
__ controller parameters calculation based on the noiseless response,
-- controller parameters calculation based on response with added noise (K_{filt} = 0.3),
-.- controller parameters calculation based on the shorten step response (T_{fin} = 7s)

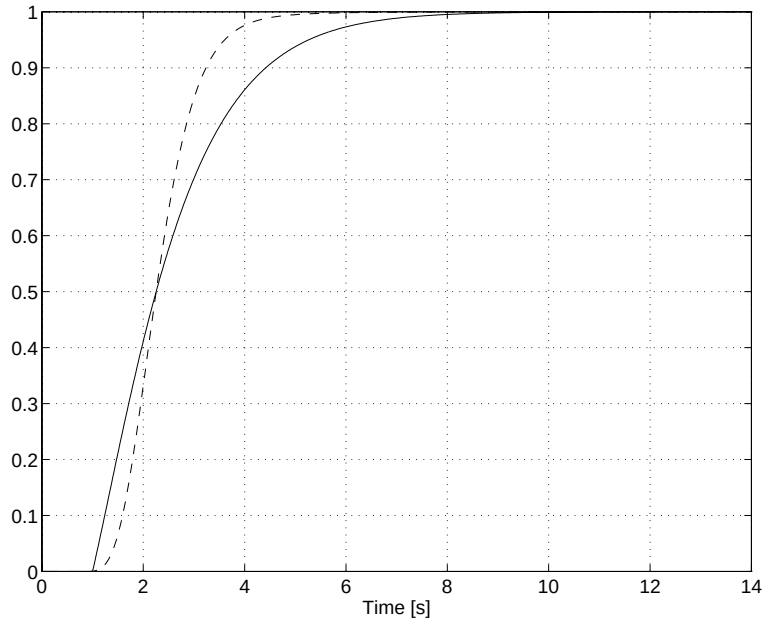


Fig. 43. The open-loop response of the process $G_{P8}(s)$;
__ linear process (K_{nl} = 0), -- non-linear process (K_{nl} = 0.15)

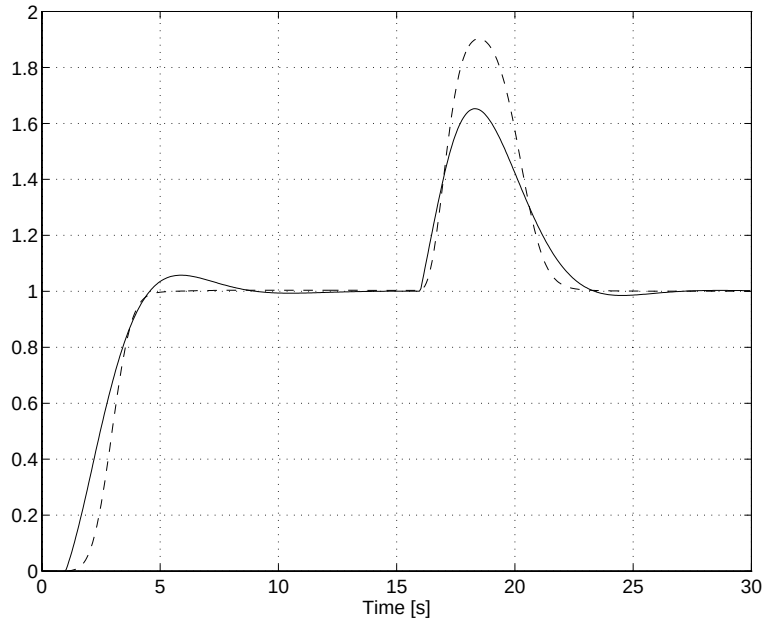


Fig. 44. The closed-loop response of the process $G_{P8}(s)$;
__ controller parameters calculation based on the linear response,
-- controller parameters calculation based on the non-linear response (K_{nl} = 0.15)

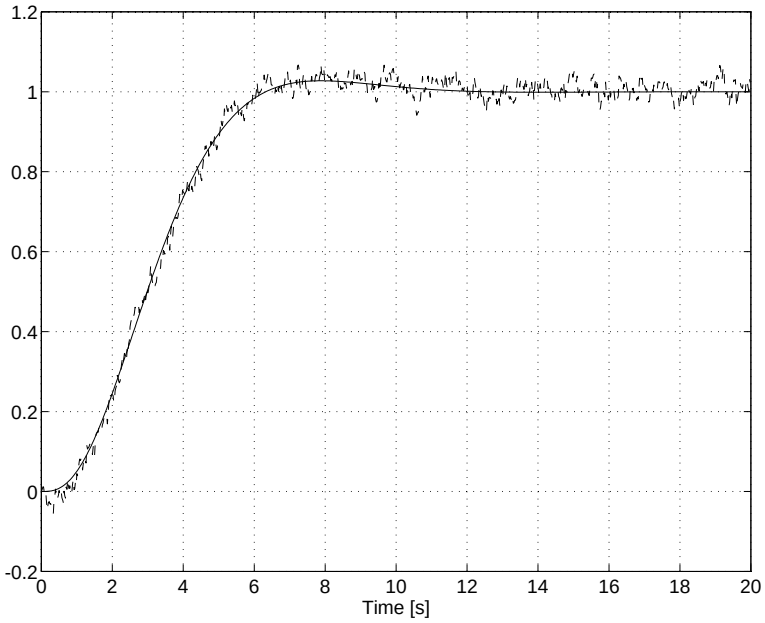


Fig. 45. The open-loop response of the process $G_{P9}(s)$;
__ noiseless response, -- response with added noise (K_{filt} = 0.3)

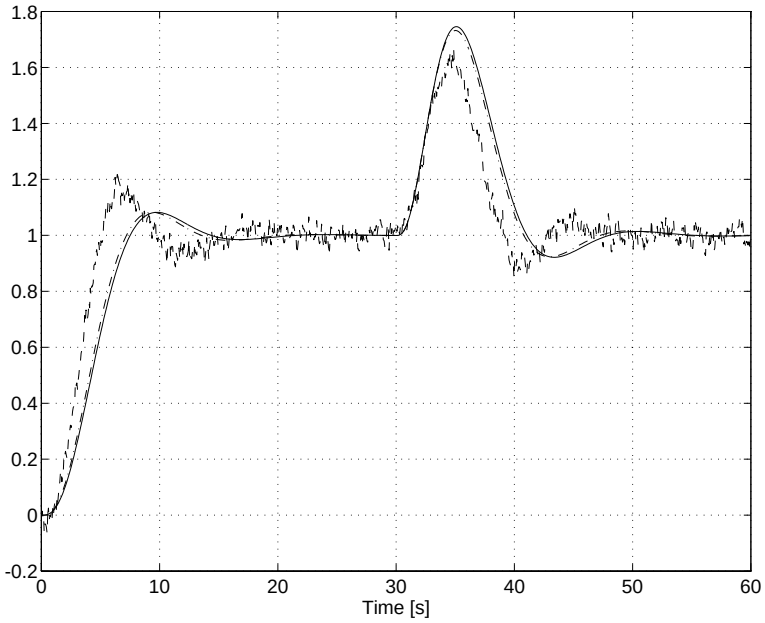


Fig. 46. The closed-loop response of the process $G_{P9}(s)$;
__ controller parameters calculation based on the noiseless response,
-- controller parameters calculation based on response with added noise (K_{filt} = 0.3),
-.- controller parameters calculation based on the shorten step response (T_{fin} = 12s)

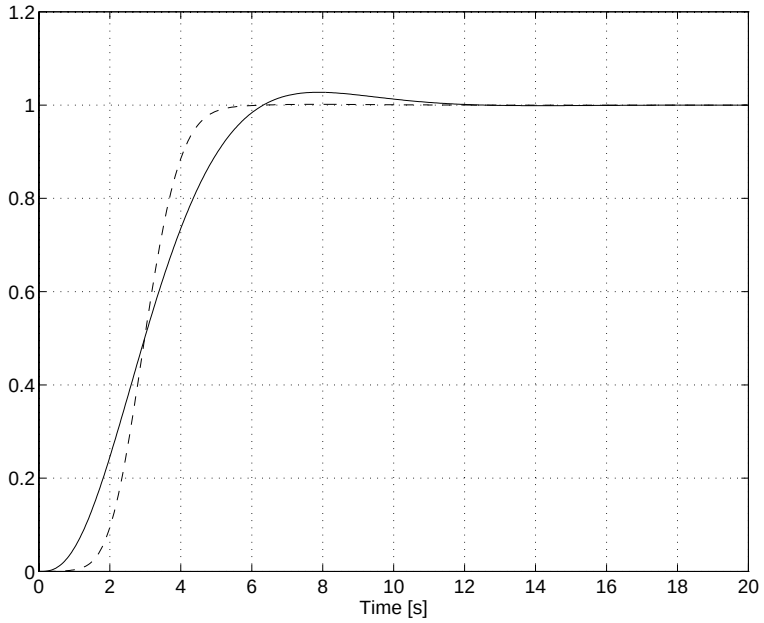


Fig. 47. The open-loop response of the process $G_{P9}(s)$;
__ linear process (K_{nl} = 0), -- non-linear process (K_{nl} = 0.15)

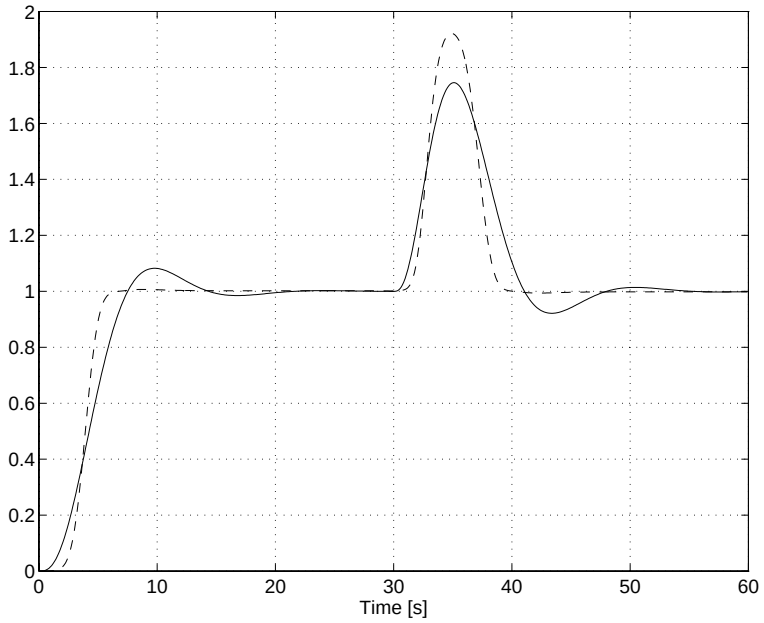


Fig. 48. The closed-loop response of the process $G_{P9}(s)$;
__ controller parameters calculation based on the linear response,
-- controller parameters calculation based on the non-linear response (K_{nl} = 0.15)

The simulation results in Figs. 13 to 48 show that the new tuning method, based on the process reaction curve, gives quite good results when testing the system closed-loop time response. Even when quite the high level of noise is present in the system (K_{filt}=0.3), the new method still gives stable results for tested processes. It can be seen that, as mentioned before, the biggest difference between the noiseless and a process with present noise can be seen for the processes G_{P1}, G_{P2}, G_{P3}, G_{P6} and G_{P9}. In next chapter it will be outlined that the obtained results for the processes with noise can be considerably improved by shortening the integration time.

Closed-loop time responses also show that the new method is not too sensitive to badly estimated and of experiment time (T_{fin}), so it can turn out as a quite robust method for real applications.

The results obtained with non-linear processes showed that the new method is also quite robust on process non-linearity. It can be seen that even when the non-linearity in the process is quite distinctive (K_{nl}=0.15), what can be clearly seen from the open-loop time responses, the closed-loop behaviour remains acceptable for all tested processes.

5. Discussions and limit cases

5.1. Practical multiple integration

The new tuning method is based on the multiple integrations, as derived in previous chapter. It will be outlined that it is very important how to perform such integrations. Moreover, the results might be completely unusable in the presence of noise.

The original expressions for calculating the areas A_1 to A_3 are given by expressions (15) to (20). Instead of successive calculating the results (16), (17) and (20), one can insert (15) into (18) and (15) and (18) into (19) and get the following results:

$$y_1(t) = t - \int_0^t y(t)dt \tag{41}$$

$$y_2(t) = A_1 t - \frac{t^2}{2} + \int_0^t \left[\int_0^t y(t)dt \right] dt \tag{42}$$

$$y_3(t) = A_2 t - A_1 \frac{t^2}{2} + \frac{t^3}{6} - \int_0^t \left[\int_0^t \left[\int_0^t y(t)dt \right] dt \right] dt \tag{43}$$

The area A_1 is calculated from (41) when inserting $t=T_{fin}$ (the end of experiment time). The resulted A_1 is then inserted into (42) and A_2 is determined by asserting $t=T_{fin}$. The area A_2 is then used in (43) and A_3 is calculated by using $t=T_{fin}$.

The MATLAB procedure AREATEST.M, which performs such calculations, is given in appendix. The results, when using the noise filter K_{filt}=0.3 for each tested process, are shown in Table 29. When comparing to appropriate values in Tables 2 to 10 (for K_{filt}=0.3), the significant difference can seen. The calculated controller gain K becomes even negative for processes G_{P6} and G_{P9}.

Process	A_0	A_1	A_2	A_3	α	K	T_i
$G_{P1}(s)$	1.023	2.198	3.855	8.246	0.028	17.77	2.139
$G_{P2}(s)$	0.99	2.882	4.366	2.651	3.746	0.135	0.607
$G_{P3}(s)$	1.023	2.193	4.272	9.11	0.029	17.27	2.132
$G_{P4}(s)$	1.002	3.982	9.696	18.62	1.073	0.465	1.92
$G_{P5}(s)$	1.031	8.551	44.74	201.2	0.901	0.538	4.497
$G_{P6}(s)$	1.023	2.087	3.874	8.774	-0.079	-6.21	2.265
$G_{P7}(s)$	0.99	3.897	7.952	10.95	1.829	0.276	1.378
$G_{P8}(s)$	0.99	2.497	3.362	1.711	3.908	0.129	0.509
$G_{P9}(s)$	1.028	3.418	9.268	32.07	-0.012	-39.8	3.461

Table 29. The resulted areas and controller constants when using equations (41) to (43) @ $K_{filt}=0.3$

The example showed that we must be careful when calculating areas A_1 to A_3. The reason for such bad results lies in the fact that the errors were accumulated at the end of experiments, when process step responses were actually close to the final values.

Figures 49 to 52 depict above statements. For the process $G_{P2}(s)$, we were studying what is happening to the calculated values $y_1(t)$, $y_2(t)$ and $y_3(t)$ (see (15), (18) and (19)). The filter $K_{filt}=0.3$ was used and the simulation time was $T_{fin}=15$s. The open-loop time response is given in Fig. 49. Fig. 50 shows the first integral $y_1(t)$. It can be seen that $y_1(t)$, for the system with noise, tracks the noiseless response quite well at the beginning, but when process approaches the new steady-state value, the response of the process with the noise starts to drift apart from the noiseless response. This drift is afterwards accumulated by additional integrations as shown in Figs. 51 and 52. From Fig. 52 can be seen that the difference between noiseless response $y_3(t)$ and the response of the process with noise becomes quite significant.

The accuracy of the new method can be therefore significantly improved by shortening the simulation time. It is shown in Tables 30 to 33, where the processes, which were the most sensitive to the added noise (G_{P1}, G_{P2}, G_{P6} and G_{P9}), were tested. When comparing these results with the original ones from Tables 2, 3, 7 and 10, we can see quite improved results.

The experiment on the process G_{P2} was made to show improvements made by using the shorter simulation time, where $T_{fin}=10$s was used. The results can be seen in Figures 53 to 56. Comparing the functions $y_1(t)$, $y_2(t)$ and $y_3(t)$ to those obtained by using $T_{fin}=15$s (Figs. 49 to 52), show obvious improvements.

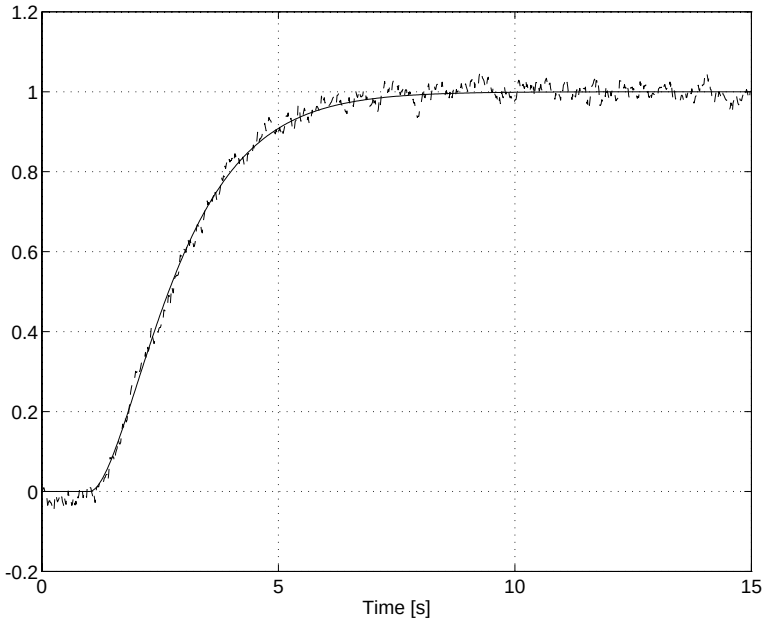


Fig. 49. The open-loop response of the process $G_{P2}(s)$; Simulation time T_{fin}=15s; __ noiseless response, -- response with added noise (K_{filt} = 0.3)

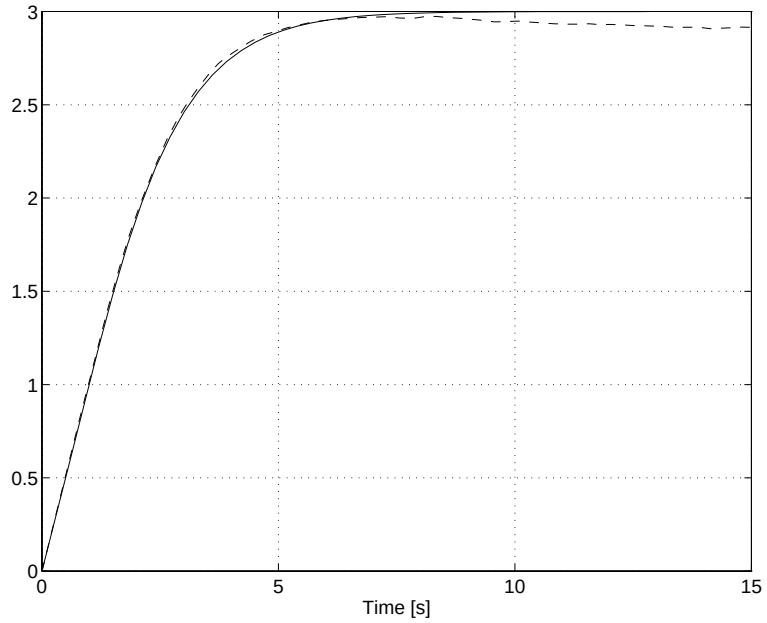


Fig. 50. The first integral $y_1(t)$ of the process $G_{P2}(s)$; Simulation time T_{fin}=15s; __ noiseless response, -- response with added noise (K_{filt} = 0.3)

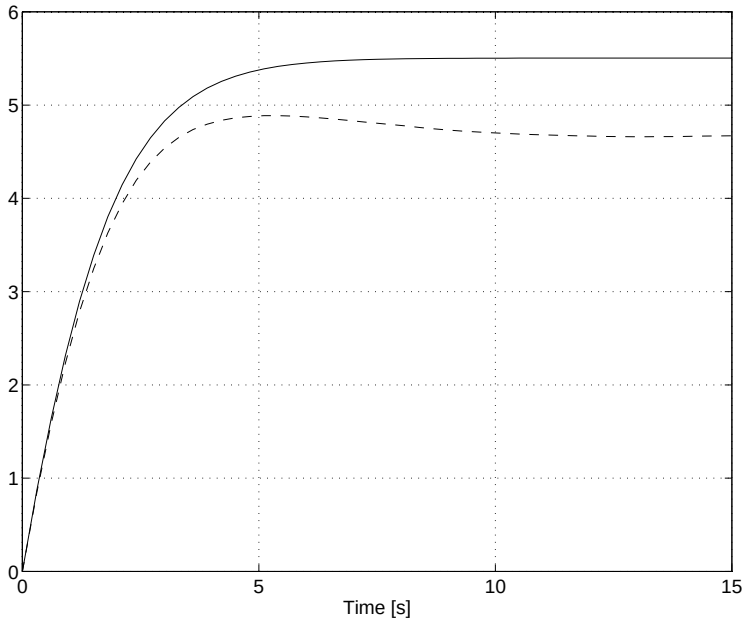


Fig. 51. The second integral $y_2(t)$ of the process $G_{P2}(s)$; Simulation time T_{fin}=15s; __ noiseless response, -- response with added noise (K_{filt} = 0.3)

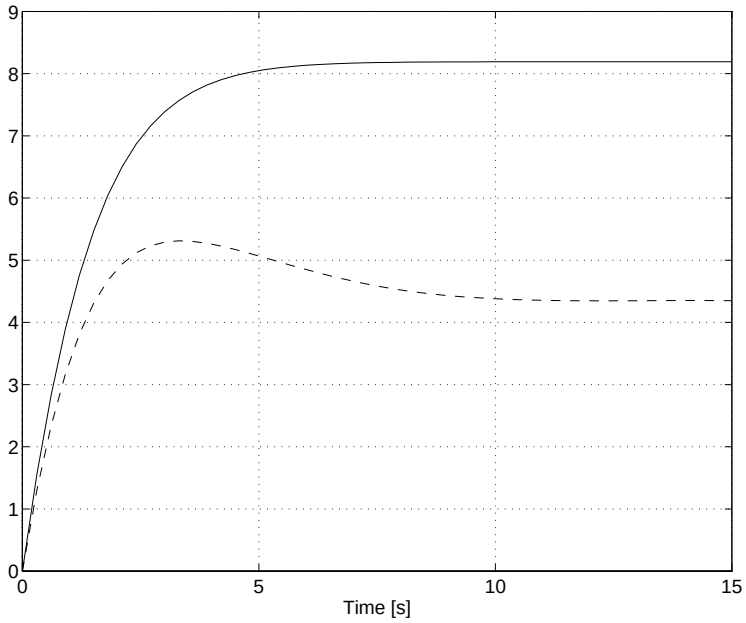


Fig. 52. The third integral $y_3(t)$ of the process $G_{P2}(s)$; Simulation time T_{fin}=15s; __ noiseless response, -- response with added noise (K_{filt} = 0.3)

K_{filt}	A_0	A_1	A_2	A_3	α	K	T_i
0	1.000	1.999	2.502	2.674	0.871	0.574	1.069
0.1	1.001	1.994	2.463	2.548	0.928	0.539	1.035
0.2	1.003	2.00	2.487	2.615	0.902	0.553	1.051
0.3	1.005	2.005	2.495	2.635	0.898	0.554	1.056

Table 30. Calculated values of areas and controller parameters vs. noise amplitude K_{filt} for the process $G_{P1}(s)$ when experiment time is shorten to T_{fin}=8s.

K_{filt}	A_0	A_1	A_2	A_3	α	K	T_i
0	1.000	3.00	5.504	8.191	1.016	0.492	1.488
0.1	1.00	3.001	5.453	7.943	1.06	0.472	1.457
0.2	1.002	3.019	5.503	8.096	1.052	0.474	1.471
0.3	1.005	3.033	5.548	8.268	1.035	0.481	1.49

Table 31. Calculated values of areas and controller parameters vs. noise amplitude K_{filt} for the process $G_{P2}(s)$ when experiment time is shorten to T_{fin}=10s.

K_{filt}	A_0	A_1	A_2	A_3	α	K	T_i
0	1.0	1.875	2.424	2.729	0.665	0.752	1.126
0.1	1.00	1.87	2.387	2.611	0.71	0.704	1.094
0.2	1.002	1.873	2.39	2.61	0.717	0.696	1.091
0.3	1.004	1.878	2.415	2.688	0.687	0.724	1.113

Table 32. Calculated values of areas and controller parameters vs. noise amplitude K_{filt} for the process $G_{P6}(s)$ when experiment time is shorten to T_{fin}=8s.

K_{filt}	A_0	A_1	A_2	A_3	α	K	T_i
0	1.00	3.00	5.012	5.072	1.964	0.255	1.012
0.1	1.002	3.03	5.186	5.731	1.742	0.286	1.105
0.2	1.006	3.057	5.349	6.543	1.5	0.332	1.223
0.3	1.009	3.11	5.649	7.767	1.262	0.393	1.375

Table 33. Calculated values of areas and controller parameters vs. noise amplitude K_{filt} for the process $G_{P9}(s)$ when experiment time is shorten to T_{fin}=16s.

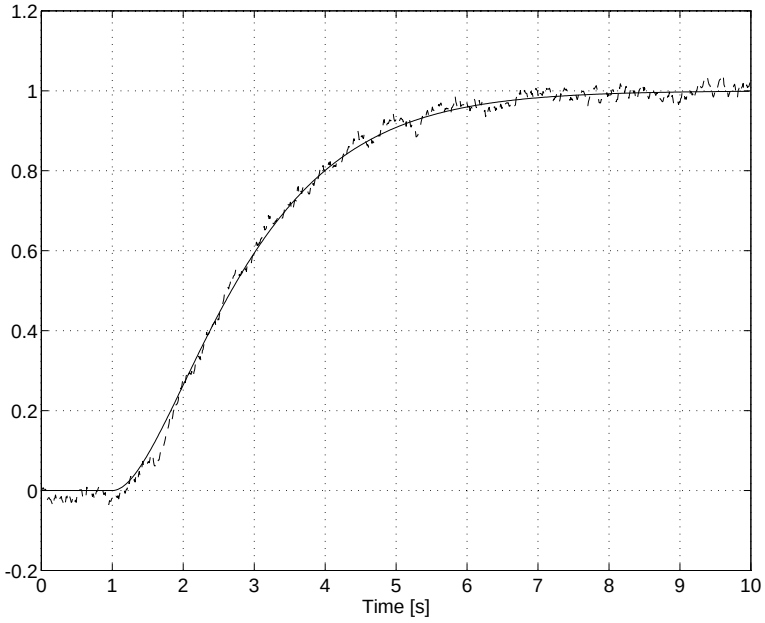


Fig. 53. The open-loop response of the process $G_{P2}(s)$; Simulation time T_{fin}=10s; __ noiseless response, -- response with added noise (K_{filt} = 0.3)

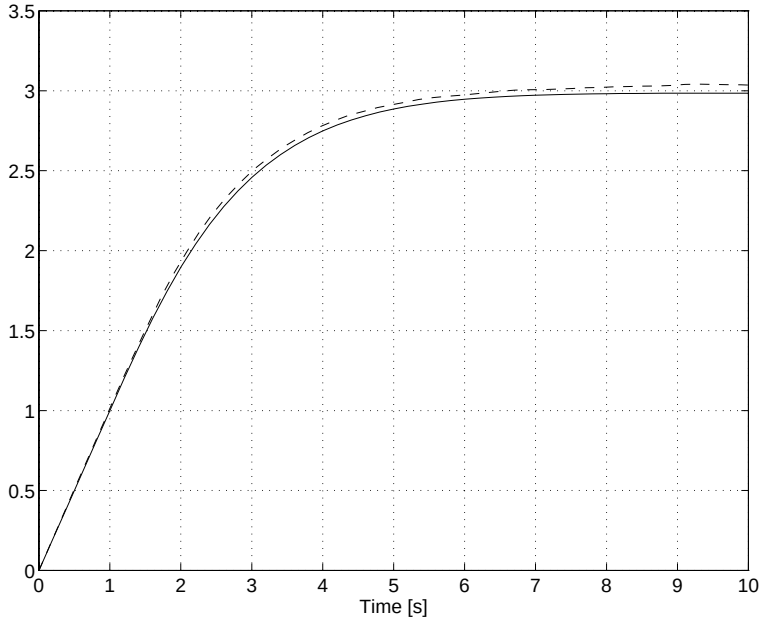


Fig. 54. The first integral $y_1(t)$ of the process $G_{P2}(s)$; Simulation time T_{fin}=10s; __ noiseless response, -- response with added noise (K_{filt} = 0.3)

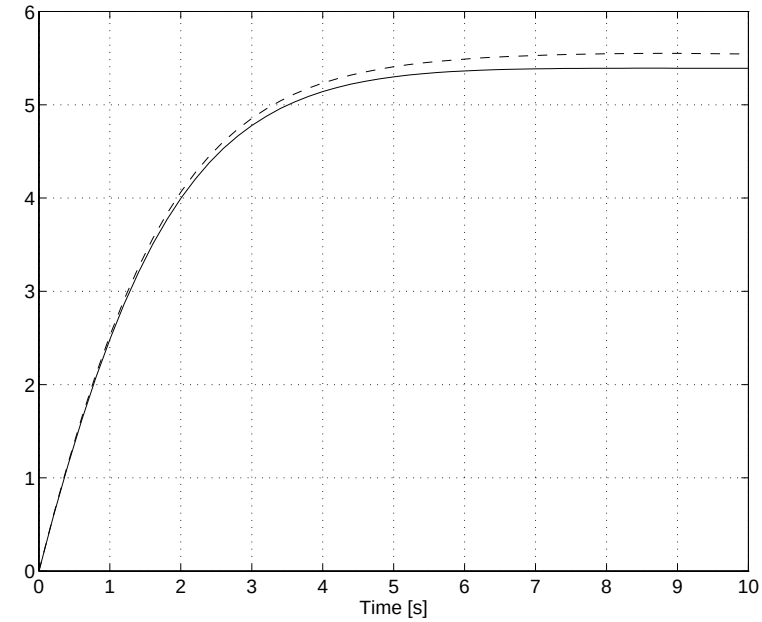


Fig. 55. The second integral $y_2(t)$ of the process $G_{P2}(s)$; Simulation time T_{fin}=10s; __ noiseless response, -- response with added noise (K_{filt} = 0.3)

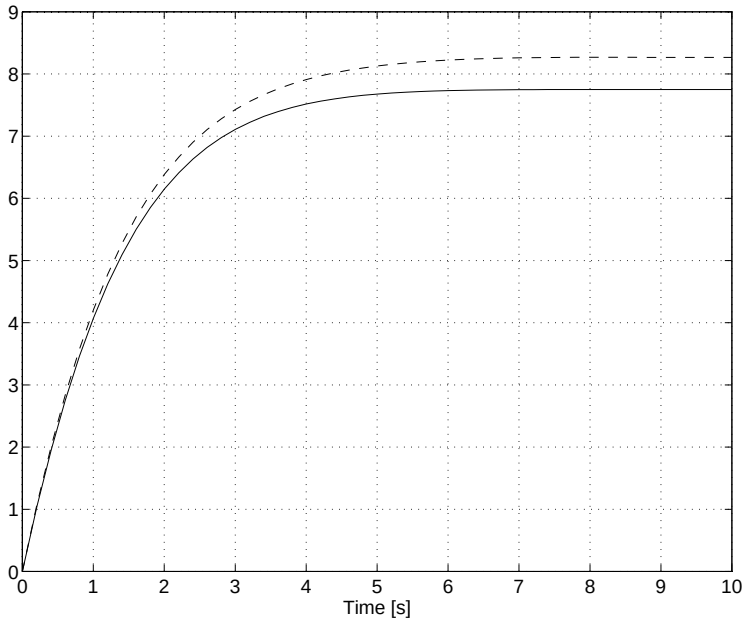


Fig. 56. The third integral $y_3(t)$ of the process $G_{P2}(s)$; Simulation time T_{fin}=10s; __ noiseless response, -- response with added noise (K_{filt} = 0.3)

To improve the accuracy of the new method even more, we decided to separate the experiment time from the integration time. The experimental results from the end of integration time (T_{int}) to the end of experiment time (T_{fin}) were used for determination the process gain (A_0) and the process response during the integration time was used for calculating surfaces A_1 to A_3 (see Fig. 57).

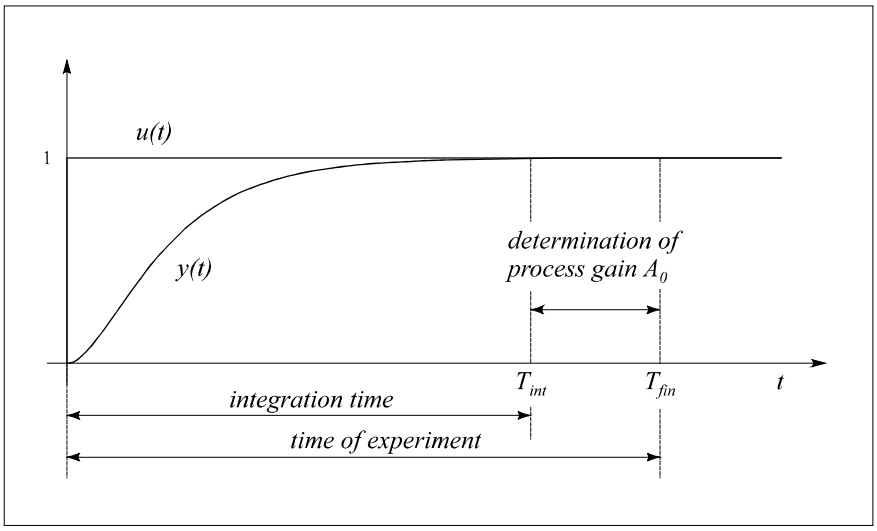


Fig. 57. A new approach for determination of A_0 to A_3

Different experiments were made to show the improvements made by the new approach. For the process G_{P9}, which previously gave the worse results, a new values of areas A_1 to A_3 and the values of controller parameters were obtained which are given in Table 34. Comparing these values to ones given in Table 33 and Table 10, we can see significant improvements of the new approach.

Tfin	***Tint***	A_0	A_1	A_2	A_3	α	***K***	T_i
20s	14s	1.007	3.058	5.21	5.741	1.775	0.28	1.102

Table 34. Calculated values of areas and controller parameters @ noise amplitude K_{filt}=0.3 for process $G_{P9}(s)$ when using the separated integration and experimental time.

The modified MATLAB procedure STEPTUN1.M is given in appendix.

5.2. Limit cases

In this chapter the applicability of the new tuning method will be tested on several other processes and process constants than those given in Table 1.

Firstly, some process models which give ***negative value of factor*** α will be treated. The first representative is the process similar to the process $G_{P9}(s)$ but with the following parameters:

$$G_P(s) = \frac{1}{(1+s)(1+s(1+2j))(1+s(1-2j))} \tag{44}$$

The open-loop process step response is shown in Fig. 58 and the process frequency response is given by the Bode plot in Fig. 59. The frequency peak in Fig. 59 causes the overshoot in time domain, as shown in Fig. 58.

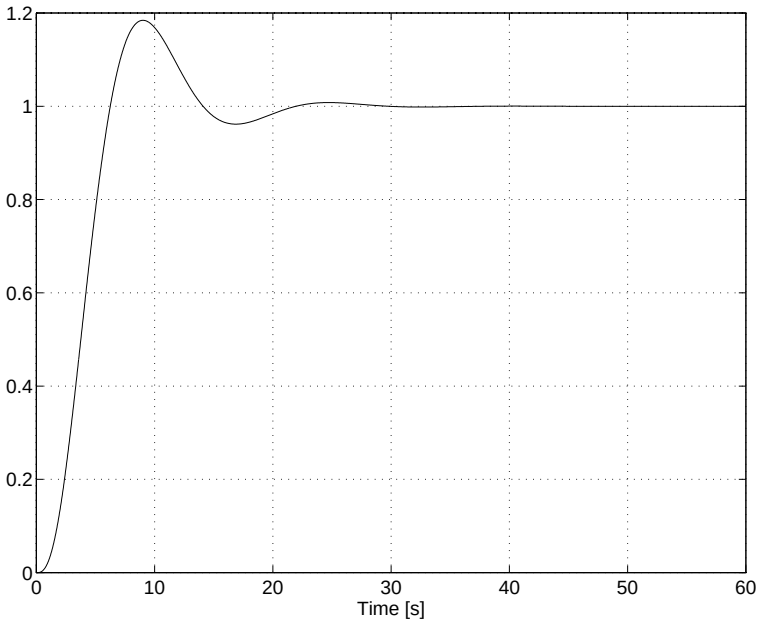


Fig. 58. The process $G_P(s)$ step response

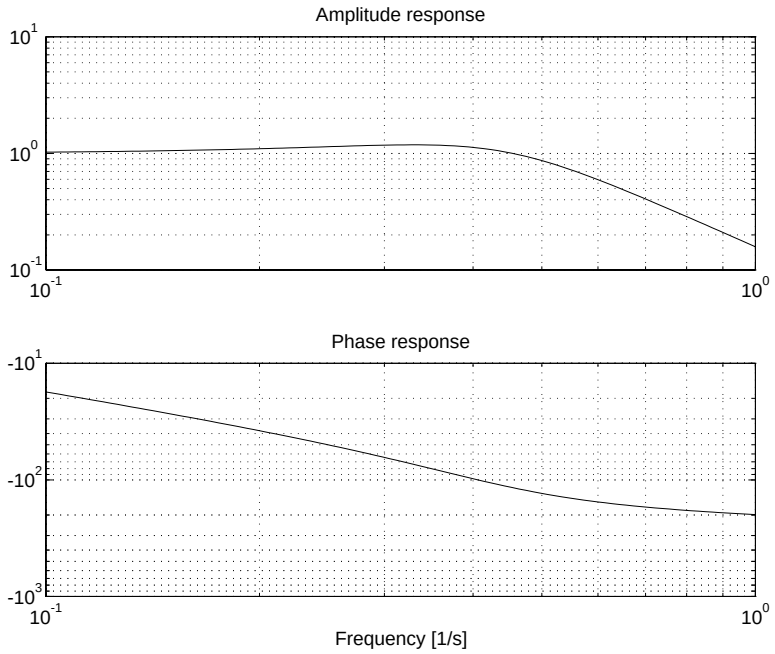


Fig. 59. The Bode plot of the process $G_P(s)$

The values of calculated areas and the controller parameters are given in Table 35. It can be noticed that factor α is negative. Although the resulted controller parameters K and T_i are negative, it will be proved that the resulted closed-loop behaviour is acceptable.

A_0	A_1	A_2	A_3	α	K	T_i
1.00	3.00	2.134	-9.044	-1.708	-0.293	-4.239

Table 35. Calculated values of areas and controller parameters for $G_P(s)$

The Nyquist diagram of the process (44) and the controller given in Table 35, is shown in Fig. 60. It is obvious that the shape of the Nyquist curve is the one defined by the frequency response method (FRM). The real part equals to -1/2 at small frequencies and the value is greater for all higher frequencies. The closed-loop time response is given in Fig. 61. It can be seen that the response behaves like the one for the phase non-minimal process, but no oscillations can be noticed on time response. The system is critically damped.

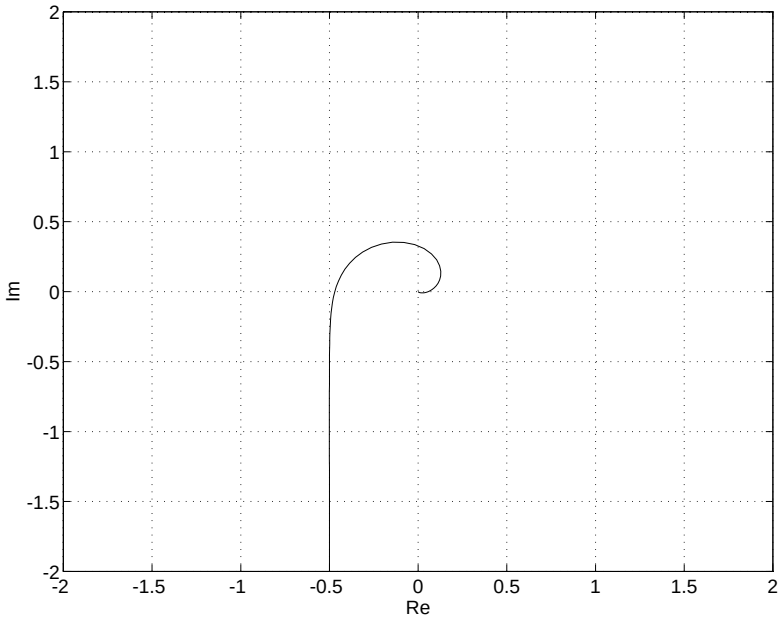


Fig. 60. The Nyquist frequency response of the process and controller ($G_P(j\omega)G_C(j\omega)$)

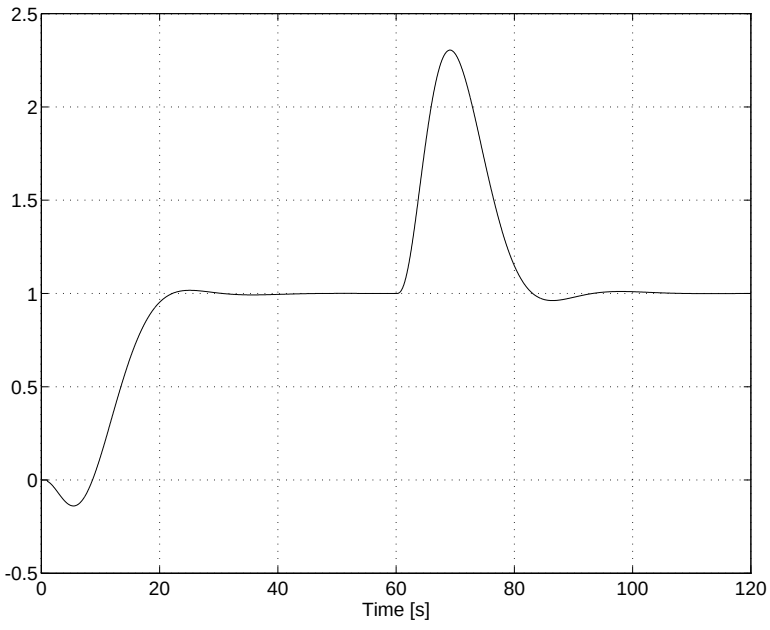


Fig. 61. The closed-loop time response of $G_P(s)$ and the controller given in Table 35

The second example is performed with a process with relatively strong time constant of the zero (The process transfer function is similar to the $G_{P8}(s)$):

$$G_P(s) = \frac{e^{-s}(1+2s)}{(1+s)^2} \tag{45}$$

The open-loop process step response is shown in Fig. 62 and the process frequency response is given by the Bode frequency response in Fig. 63. The frequency peak in Fig. 63 causes the overshoot in time domain, as shown in Fig. 62.

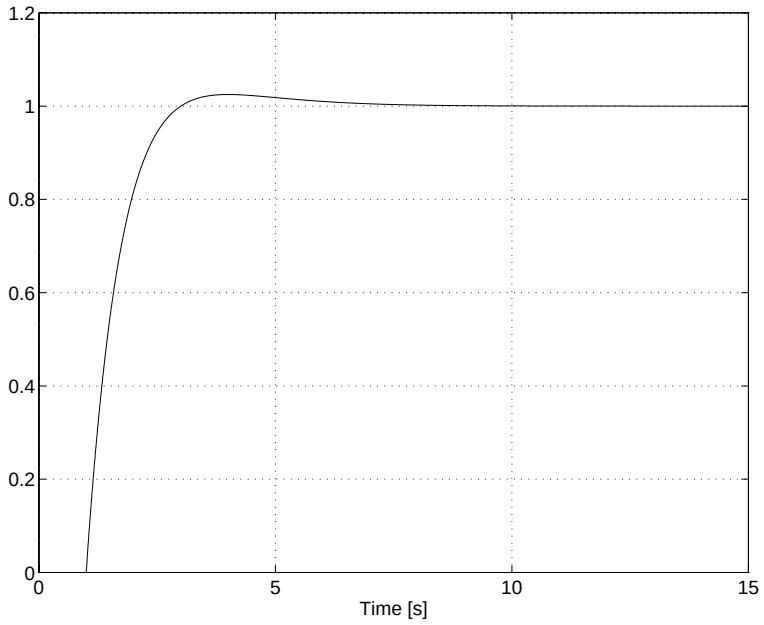


Fig. 62. The process $G_P(s)$ step response

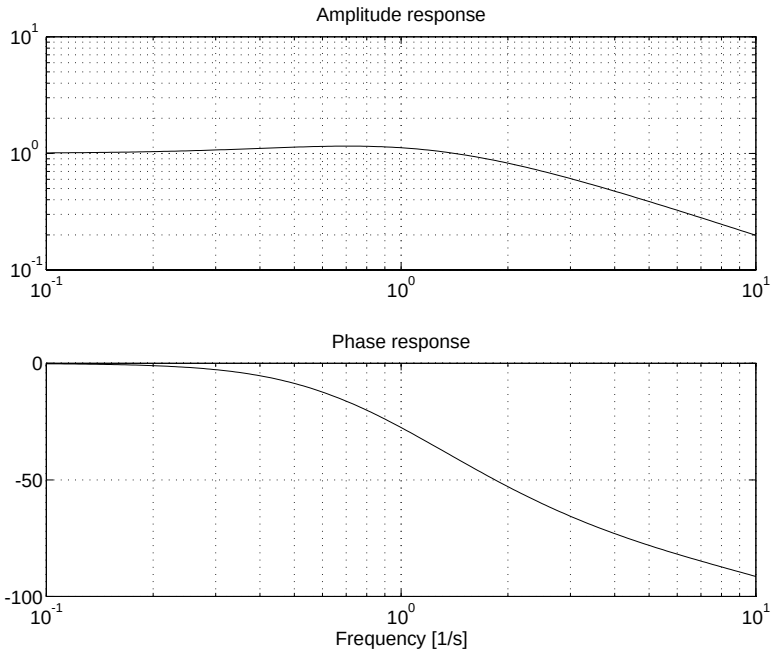


Fig. 63. The Bode plot of the process $G_P(s)$ frequency response

The calculated values of the areas with controller parameters are given in Table 36. It can be noticed that factor α has again negative sign. The same as in the previous case will be proved that the resulted closed-loop behaviour is the one desired.

A_0	A_1	A_2	A_3	α	K	T_i
1.00	3.00	2.134	-9.044	-1.708	-0.293	-4.239

Table 36. Calculated values of areas and controller parameters for process $G_P(s)$

The Nyquist diagram of the process (45) and the controller given in Table 36 is shown in Fig. 64. It is obvious that the shape of the Nyquist curve is as defined by the frequency response method (FRM). The real part equals to -1/2 at small frequencies and is greater for all higher frequencies. The closed-loop time response is given in Fig. 65. It can be seen that the response is quite good. The system is critically damped again.

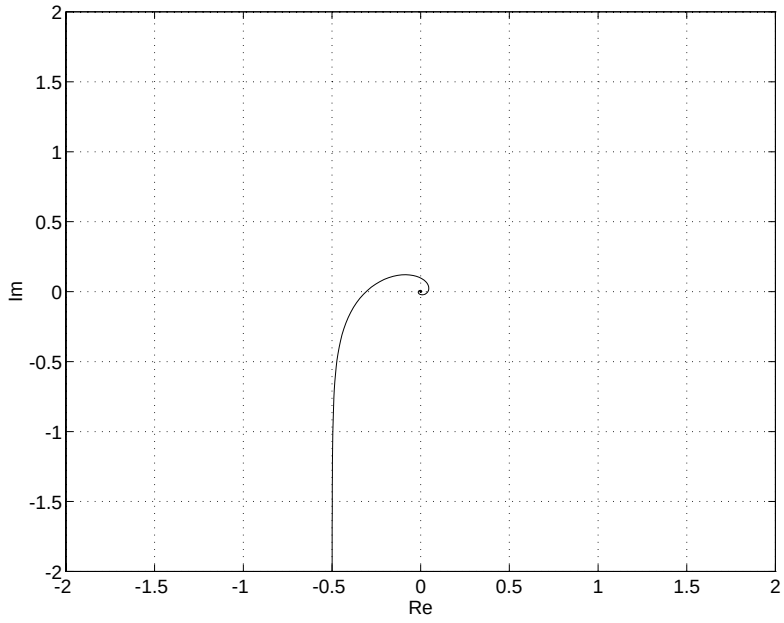


Fig. 64. The Nyquist frequency response of the process and controller ($GP(j\omega)GC(j\omega)$)

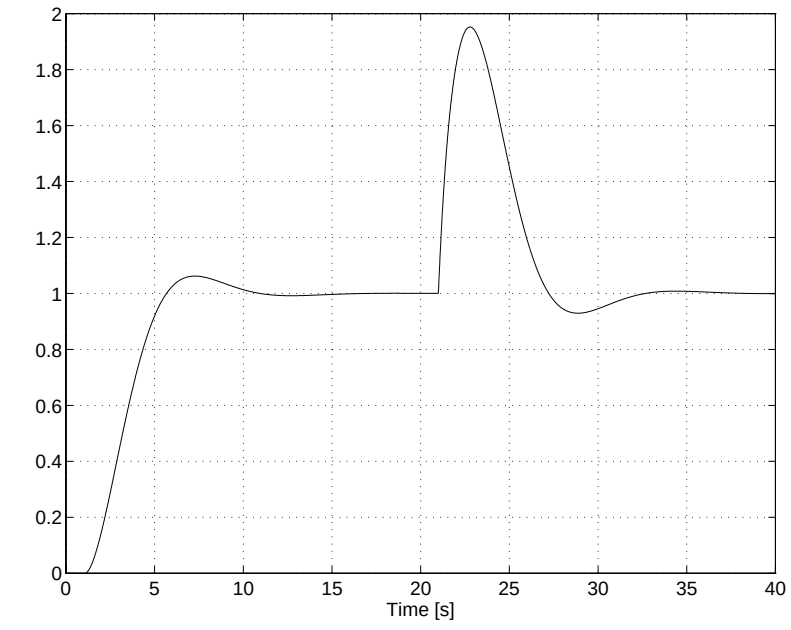


Fig. 65. The closed-loop time response of $G_P(s)$ and controller given in Table 34

If the time constant of a zero in (45) would not be 2s, but would change from 1.4s to 1.5s, the signs of K and T_i would alter, such as shown in table 37.

T_{zero}	A_0	A_1	A_2	A_3	α	K	T_i
1.4	1.00	1.601	1.313	0.52	3.041	0.164	0.396
1.5	1.00	1.501	1.016	-0.018	-84.26	-0.006	-0.018

Table 37. Calculated values of areas and controller parameters for process $G_P(s)$ with different zero time constant

In both cases it can be noticed that the controller proportional gain is relatively small, but the integral term proportional factor:

$$K_i = \frac{K}{T_i} \tag{46}$$

remains almost unchanged. This implies that the integral character of the controller, for some processes with T_{zero} between 1.4s and 1.5s, becomes dominant, the proportional gain is neglected and the PI controller becomes and can be realised by the I controller:

$$G_C(s) = \frac{K_i}{s} \tag{47}$$

In both previous cases (44) and (45) we got the negative values of K and T_i. The question which remains is what happens when K and T_i have *different* signs. Now, such case will be investigated, in which the process:

$$G_P(s) = \frac{(1+s)}{(1+2s)(1+0.1s)} \tag{48}$$

was tested. Fig. 66 shows the process step response and Fig. 67 gives the Bode frequency response.

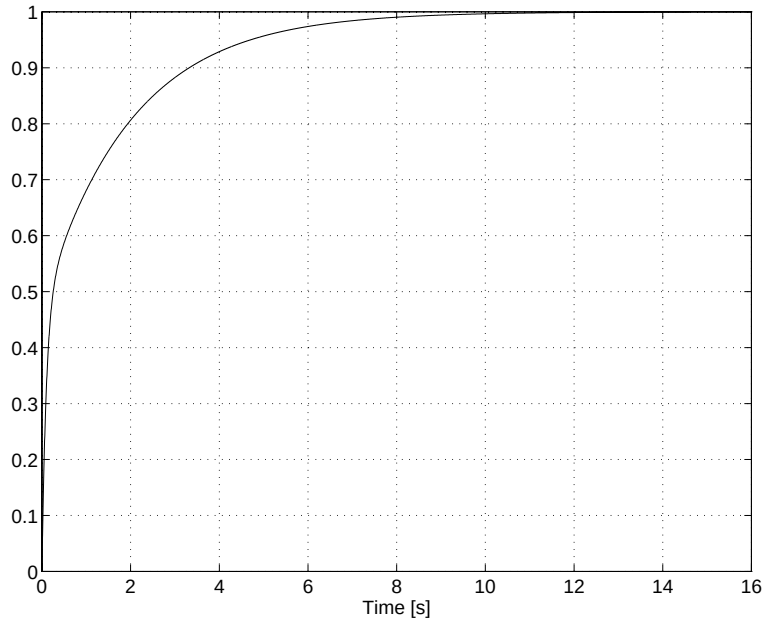


Fig. 66. The process $G_P(s)$ step response

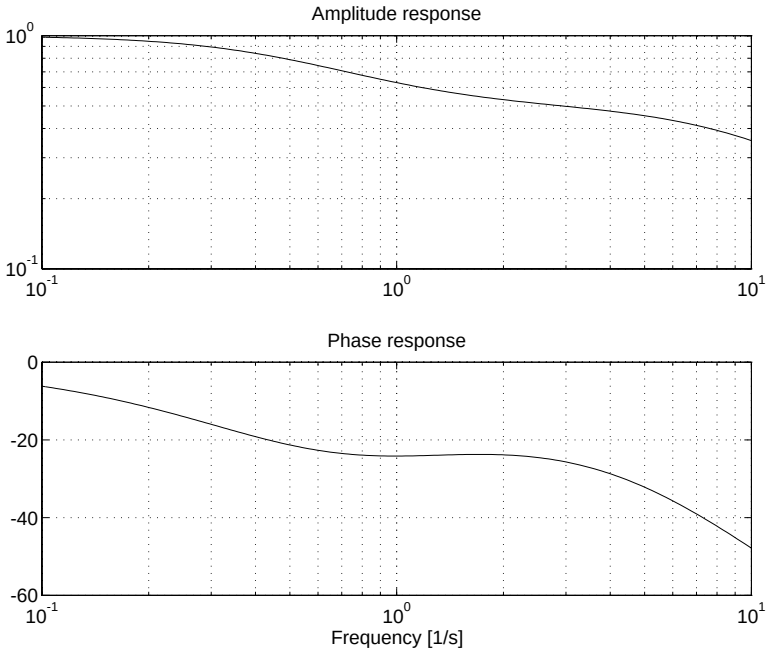


Fig. 67. The Bode plot of the process $G_P(s)$ frequency response

The new tuning algorithm gives the next values of the surfaces and controller parameters:

A_0	A_1	A_2	A_3	α	K	T_i
1.00	1.097	2.076	3.977	-0.427	-1.17	1.916

Table 38. Calculated values of areas and controller parameters for process $G_P(s)$

Such controller parameters give the unstable closed-loop response. As a rule of thumb in such cases is to change the sign of α (from negative to positive value or vice versa) and recalculate values of K (30) and T_i (28). In particular case, the modified controller parameters for α=0.427 are K=1.17 and T_i=0.769s.

Fig. 68 gives the Nyquist diagram of the process (47) and the modified controller. It is obvious that the shape of the Nyquist curve is as defined by the frequency response method (FRM), where the real part equals to -1/2 at small frequencies and the value is greater for all higher frequencies.

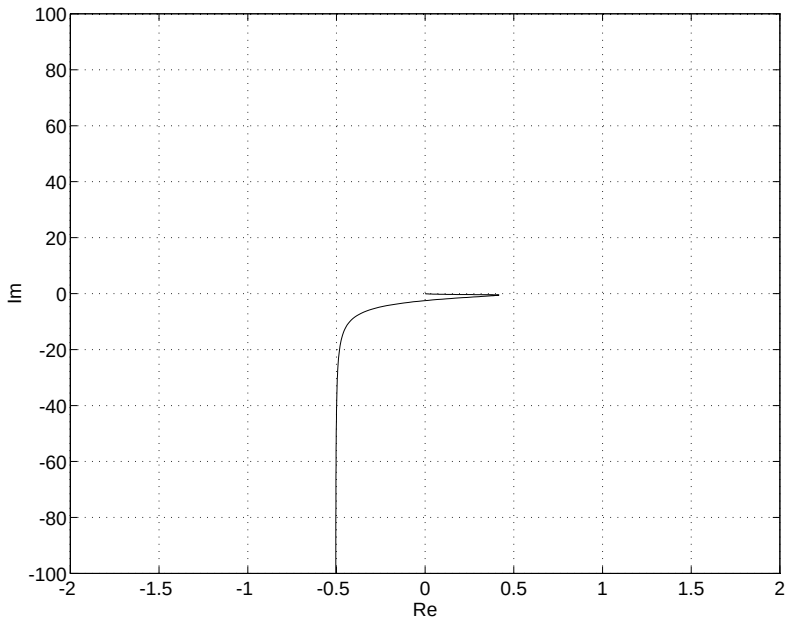


Fig. 68. The Nyquist frequency response of the process and modified controller ($G_P(j\omega)G_C(j\omega)$)

The closed-loop time response of the process with a modified controller is given in Fig. 69. The response is again critically damped.

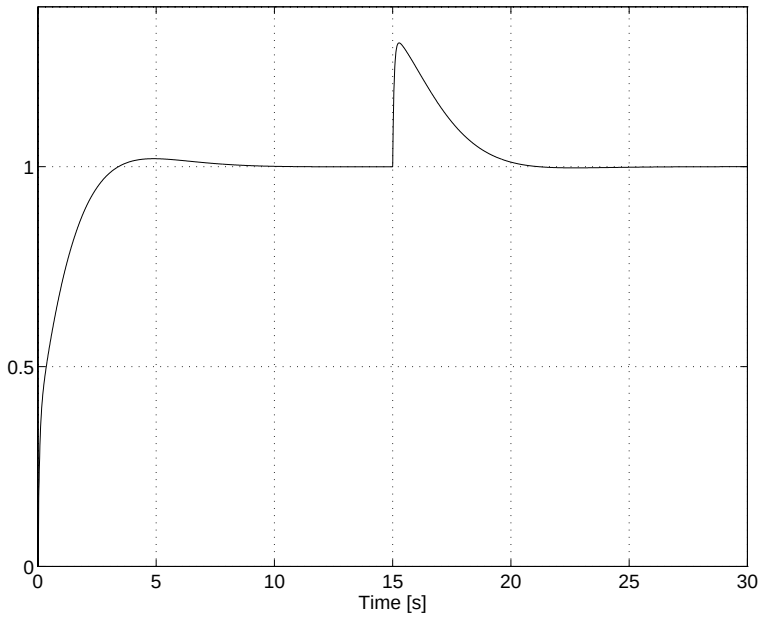


Fig. 69. The closed-loop time response of the process (47) and the modified controller

Another limit case is when factor α has *low values*. In such case, the proportional gain K will have great values (27), (30). If the controller gain K must be limited (what is the case in all practical implementations (see [Peng et al.]), then T_i (28) can be recalculated from (30) as:

$$T_i = \frac{A_1}{1 + \frac{0.5}{A_0 K}} \tag{49}$$

This can happen when dealing with the low-order processes, e.g. the first order process. The theoretical value of the factor α for such process is α=0. The small values of the factor α also appear for the second or higher order processes where one (the main) time constant is dominant and all the rest of constants can be neglected (they are much smaller than the main one). The controller parameter calculation for such processes is also more sensitive to noise, where factor α can easily become smaller than 0. What to do in such cases was already given in this chapter.

5.3. Derivative (D) part of a PID controller

The next question which appears is whether it is possible to calculate the derivative part of the PID controller using the new method approach. Here we will try to use only the already available information about the surfaces A_1, A_2 and A_3. It is assumed that the three successive integrations are enough (maximum) in real processes and that additional integrations would not give so exact information more. In our further work the latter assumption will be tested.

We try to find the relation between process constants by using the "schoolbook" PID controller:

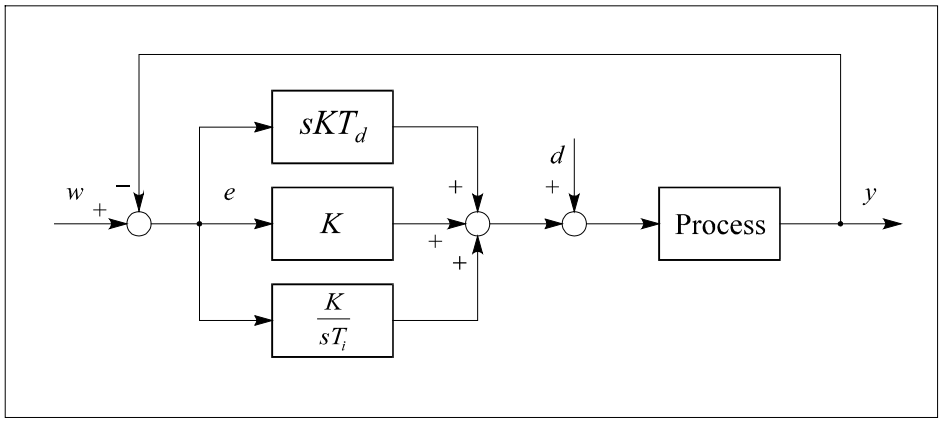


Fig. 70. The "schoolbook" PID controller

The controller transfer function is therefore:

$$G_C(s) = K\frac{1 + sT_i + s^2T_iT_d}{sT_i} \tag{50}$$

When applying such controller to the process (5), and when expressing the coefficient c_0 to c_3 and d_0 to d_3 in (6), we get the following result:

$$
\begin{aligned}
c_0 &= T_i \\
c_1 &= a_1 T_i \\
c_2 &= a_2 T_i \\
c_3 &= a_3 T_i \\
&\vdots \\
d_0 &= K \\
d_1 &= K\left(b_1 + T_i\right) \\
d_2 &= K\left(b_2 + b_1 T_i + T_d T_i\right) \\
d_3 &= K\left(b_3 + b_2 T_i + b_1 T_d T_i\right) \\
&\vdots
\end{aligned}
\tag{51}
$$

Inserting the changed constants (51) into (10) and (14), the changed value of the constant α can be expressed:

$$
\alpha = \frac{A_1 A_2}{A_3} - 1 - T_d \frac{A_1^{\,2}}{A_3} \tag{52}
$$

From (52) is obvious that α depends on the chosen derivative time constant T_d, but there are given no strict guidelines how to choose parameter T_d. It can be seen that the factor α becomes zero when T_d becomes equal to:

$$
T_{d\,\mathbf{max}} = \frac{A_1 A_2 - A_3}{A_1^{\,2}} \tag{53}
$$

So, the actual T_d should be smaller than T_{dmax}. In fact, T_d must be such smaller than T_{dmax}, that the system will remain stable.

The example was performed to depict above statements. The following process was used:

$$
G_P\left(s\right) = \frac{1}{\left(1+s\right)^3} \tag{54}
$$

Table 39 gives the values of areas, which were calculated from the process step response:

A_0	A_1	A_2	A_3
1.00	3.00	6.00	10.0

Table 39. Calculated values of areas for the process $G_P(s)$

The process was tested with 4 different differential time constants: T_d=0s, T_d=0.3s, T_d=0.6s and T_d=0.8s. The values of α (52) and corresponding controller parameters K (27) and T_i (28) are given in Table 40.

T_d	α	K	T_i
0	0.8	0.625	1.667
0.3	0.53	0.943	1.961
0.6	0.26	1.923	2.381
0.8	0.08	6.25	2.778

Table 40. Calculated values of α and controller parameters for the process $G_P(s)$

The adequate closed-loop time responses are given in Fig. 71. It is shown that all four time responses are stable. Increasing the value of T_d from 0s to 0.6s improves the closed-loop response, while using the T_d=0.8s causes damped oscillations (although it gives the fastest response). So, there exists the optimal solution for the controller constant T_d. The optimal value of T_d is the greatest T_d for which the Nyquist curve of the process and controller will still follow the frequency response method recommendations. This optimal value of T_d will be specially investigated in our further work.

The Nyquist curves are given in Fig. 72. It can be seen that the solutions for T_d=0s to T_d=0.6s give the desired shape of the Nyquist curve, given by the frequency response method, while the solution for T_d=0.8s crosses the vertical line $\mathbf{Re}\{G_C(j\omega)G_P(j\omega)\}$=-1/2 and approaches the unstable point -1+j0. Fig. 73 gives the detailed scheme of the Fig. 72.

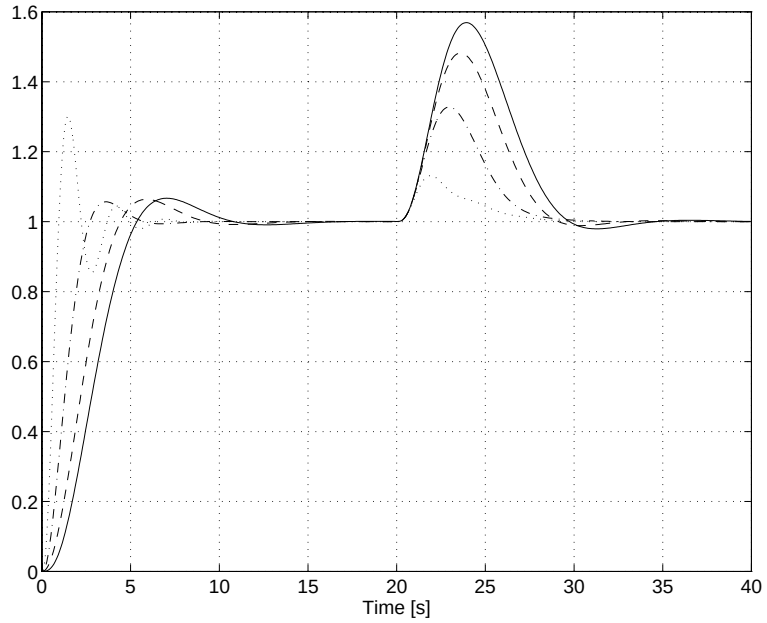


Fig. 71. The closed-loop time response of the process $G_P(s)$ at different differential time constants T_d; __ T_d=0s, -- T_d=0.3s, -.- T_d=0.6s, ... T_d=0.8s

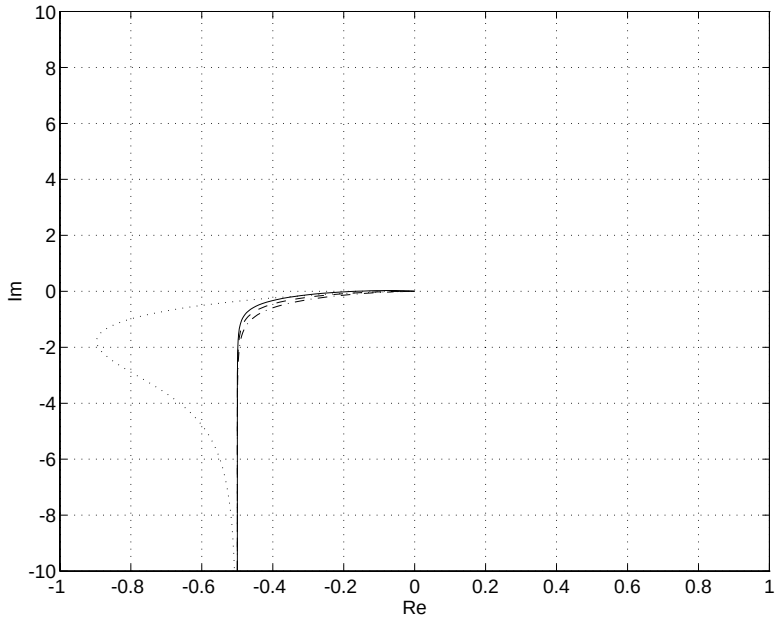


Fig. 72. The Nyquist frequency response of the process and controller ($G_P(j\omega)G_C(j\omega)$)

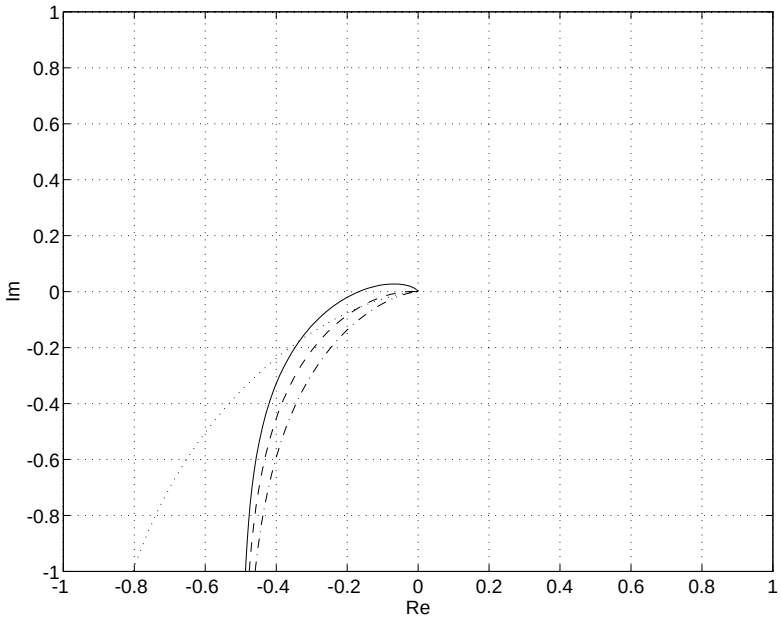


Fig. 73. The Nyquist frequency response of the process and controller ($G_P(j\omega)G_C(j\omega)$) - detailed scheme

5.4. Improving the classical tuning rules

The main drawback of the classical tuning rules, based on the process reaction curve is that they usually use only the limited information about the process (e.g. only process lag and process rise time). Therefore they can only be successfully used in a limited range of process models.

Here, the Ziegler-Nichols and Cohen-Coon tuning rules will be used for calculating controller proportional gain, while expression (49) will be used for calculating the integral time constant T_i. At first, the Ziegler-Nichols tuning rules for PI controller (see [Vrančić et al., 1995]) were used to calculate the controller parameters for the following process:

$$G_P(s) = \frac{e^{-s}}{(1+s)} \tag{55}$$

The calculated controller parameters were (see [Vrančić et al., 1995]):

$$\begin{aligned} K &= 0.9 \\ T_i &= 3.3s \end{aligned} \tag{56}$$

The closed-loop time response is given with the dashed line in Fig. 74.

A quite sluggish response can be observed. To improve the response, the controller integral time constant was recalculated using the expression (49). The new value was T_i=1.29s. The changed closed-loop response is given with the full line in Fig. 74, where much better response can be noticed.

Similarly, the Cohen-Coon tuning rules were applied to the process:

$$G_P(s) = \frac{1}{(1+s)^2}, \tag{57}$$

where the calculated controller parameters were (see [Vrančić et al., 1995]):

$$\begin{aligned} K &= 8.71 \\ T_i &= 0.774s \end{aligned} \tag{58}$$

The dashed line in Fig. 75 shows the closed-loop time response. Again, as in previous case, the integral time constant was recalculated, using the expression (49). The new value was T_i=1.89s. A full line in Fig. 75 shows the closed-loop response when using the changed controller. It can be seen, that the controller becomes more stable.

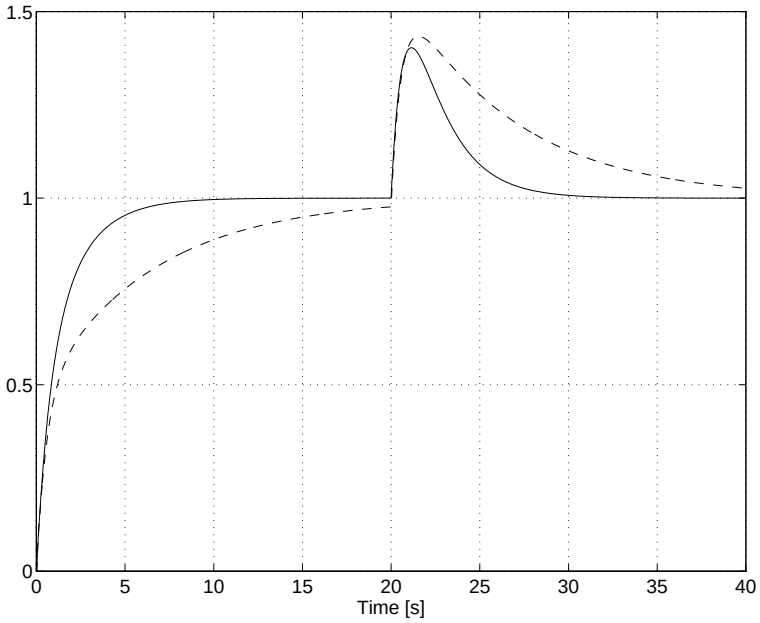


Fig. 74. The closed-loop response for the process $G_P(s)$ @ K=0.9; __ T_i=1.29s, -- T_i=3.3s

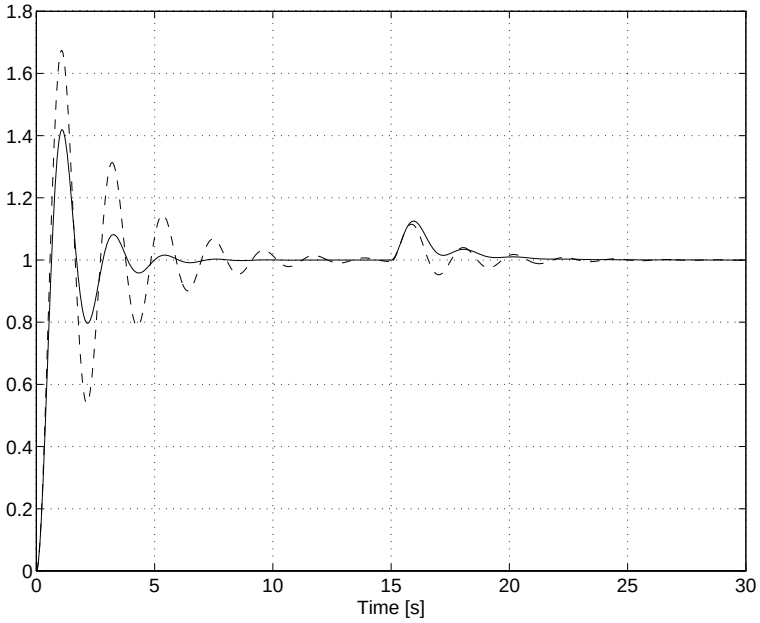


Fig. 75. The closed-loop response for the process $G_P(s)$ @ K=8.71; __ T_i=1.89s, -- T_i=0.774s

6. Conclusions

The improved controller tuning results can be seen when using the whole process reaction curve instead of detecting the process lag and rise time. In the same time, the tuning procedure is simple and can be implemented in praxis.

The only thing which should be done by the operator is to determine the experiment and integration time (see chapter 5.1). Then the procedure calculates the appropriate areas and controller constants by itself.

The advantages of such approach are:

- There is no need to detect the process reflection point (difficult for noisy processes)
- The controller parameters are exactly calculated, according to given frequency criterion, for wide spectrum of process models
- The method is not too sensitive to noise (see chapter 5.1), non-linearity and badly estimated time of experiment
- The process overshoot can be controlled by moving the vertical axis in Nyquist plot from position -1/2 to the left or to the right
- The method gives stable closed-loop responses

The drawback of such approach is that the disturbance rejection is not always the optimal one. Moving the "vertical axis" from the position -1/2 is not always the best concept. The method in the present form gives such controller which has the same transfer function from the reference (feed-forward path) and from the process output (feedback path) to the controller output. The advantages of e.g. variable set-point weighting method (see [Åström and Hägglund, 1995]) can therefore not be used.

Our further work will be based on finding the exact value of the parameter T_d, finding the rule for which processes the method can be implemented and investigating if further integrations can give some additional process information (in the presence of noise).

The accent will also be given to finding a more global concept for tuning the controllers given by the rational transfer function and to find the adequate approach in the case if the feedback and the feed-forward controller paths are not the same.

Appendix

The MATLAB procedure STEPTUN.M

```
% function [Kp,Ti,A0,A1,A2,A3,alpha] = steptun (yout);
%
% function STEPTUN.M calculates the appropriate Kp and Ti
% controller parameters from the process step response and
% its first, second and third integral.

function [Kp,Ti,A0,A1,A2,A3,alpha] = steptun (yout);

global K;
global T;
global U;
global yout1;



K = [];
T = [];
U = [];


a = max(size(yout));
Tfin = yout(a,1);

i = a;
while (yout(i,1) > 0.9*Tfin)
  i = i-1;
end

A0 = mean(yout(i:a,2));
K = 1;
T = yout(:,1);
U = yout(:,2)/A0;

[t,x,y] = rk45('openl2',Tfin);

a = max(size(yout1));

i = a;
while (yout1(i,1) > 0.8*Tfin)
  i = i-1;
end

A1 = mean(yout1(i:a,2));

K = A1;
T = yout1(:,1);
U = yout1(:,2);

[t,x,y] = rk45('openl2',Tfin);

a = max(size(yout1));

i = a;
while (yout1(i,1) > 0.8*Tfin)
  i = i-1;
end

A2 = mean(yout1(i:a,2));

K = A2;
T = yout1(:,1);
U = yout1(:,2);

[t,x,y] = rk45('openl2',Tfin);

a = max(size(yout1));

i = a;
```

```
while (yout1(i,1) > 0.8*Tfin)
  i = i-1;
end

A3 = mean(yout1(i:a,2));

alpha = A1*A2/A3 - 1;

Kp = 0.5/(A0*alpha);
Ti = A1/(1+alpha);
```

The MATLAB procedure AREATEST.M

```
% function [Kp,Ti,A0,A1,A2,A3,alpha] = areatest (yout);
%
% function AREATEST.M calculates the appropriate Kp and Ti
% controller parameters from the process step response by
% using the improper calculations of areas A1 to A3.

function [Kp,Ti,A0,A1,A2,A3,alpha] = areatest (yout);

global K;
global T;
global U;
global yout1;

K = [];
T = [];
U = [];

a = max(size(yout));
Tfin = yout(a,1);

A0 = yout(a,2);
K = 0;
T = yout(:,1);
U = -yout(:,2)/A0;

[t,x,y] = rk45('openl2',Tfin);

a = max(size(yout1));

A1 = Tfin - yout1(a,2);

K = 0;
T = yout1(:,1);
U = -yout1(:,2);

[t,x,y] = rk45('openl2',Tfin);

a = max(size(yout1));

A2 = A1*Tfin - Tfin^2/2 + yout1(a,2);

K = 0;
T = yout1(:,1);
U = -yout1(:,2);


[t,x,y] = rk45('openl2',Tfin);

a = max(size(yout1));

A3 = A2*Tfin - A1*Tfin^2/2 + Tfin^3/6 - yout1(a,2);

alpha = A1*A2/A3 - 1;

Kp = 0.5/(A0*alpha);
Ti = A1/(1+alpha);
```

The MATLAB procedure STEPTUN1.M

```
% function [K,Ti,A0,A1,A2,A3,alpha] = steptun1 (yout,Tfin,Tint);
%
% function STEPTUN1.M calculates the appropriate K and Ti
% controller parameters from the process step response and
% its first, second and third integral.

function [K,Ti,A0,A1,A2,A3,alpha] = steptun1 (yout,Tfin,Tint);

global K;
global T;
global U;
global yout1;

K = [];
T = [];
U = [];

a = max(size(yout));

i = a;
while (yout(i,1) > Tint)
  i = i-1;
end

A0 = mean(yout(i:a,2));
K = 1;
T = yout(:,1);
U = yout(:,2)/A0;

[t,x,y] = rk45('openl3',Tint);

a = max(size(yout1));

A1 = yout1(a,2);

K = A1;
T = yout1(:,1);
U = yout1(:,2);

[t,x,y] = rk45('openl3',Tint);

a = max(size(yout1));

A2 = yout1(a,2);

K = A2;
T = yout1(:,1);
U = yout1(:,2);

[t,x,y] = rk45('openl3',Tint);

a = max(size(yout1));

A3 = yout1(a,2);

alpha = A1*A2/A3 - 1;

Kp = 0.5/(A0*alpha);
Ti = A1/(1+alpha);
```

References

K. J. Åström and T. Hägglund, "**Automatic Tuning of Simple Regulators with Specifications on Phase and Amplitude Margins**", Automatica, Vol. 20, No. 5, pp. 645-651, 1984.

K. J. Åström and T. Hägglund, "**New Tuning Methods for PID Controllers**", Proceedings of 3rd European Control Conference, Rome, Vol. 3, pp. 2456-2462, 1995.

P. Boucher and Y. Tanguy, "**Une nouvelle méthode de réglage des régulateurs PI ou PID**", Automatisme, May, 1976.

J. J. DiStefano, "**Feedback and Control Systems**", 2nd edition, Schaum's outline series, McGraw-Hill, 1990.

A. Haalman, "**The adjustment of continuous controllers**", Review A, VIII, 4, pp. 181-199, 1966.

C. C. Hang, K. J. Åström and W. K. Ho, "**Refinements of the Ziegler-Nichols tuning formula**", IEE Proceedings, D, Vol. 138, No. 2, pp. 111-118, 1991.

C. C. Hang and L. S. Cao, "**Improvement of Transient Response by Means of Setpoint Weighting**", 12th World Congress IFAC, Sydney, Proceedings, Vol. 3, pp. 69-72, 1993.

C. C. Hang and K. K. Sin, "**On-line Auto-tuning of PID Controllers Based on the Cross-correlation Technique**", Report CI-91-4, National University of Singapore, Department of Electrical Engineering, 1991.

R. Hanus, "**Determination of controllers parameters in the frequency domain**", Journal A, Vol. XVI, No. 3, 1975.

W. K. Ho, C. C. Hang and L. S. Cao, "**Tuning of PID Controllers Based on Gain and Phase Margin Specifications**", 12th World Congress IFAC, Sydney, Proceedings, Vol. 5, pp. 267-270, 1993.

R. Isermann, "**Experimentelle analyse der dynamik von regelsystemen (Identifikation 1)**", 1971.

Y. Peng, D. Vrančić and R. Hanus, "**Anti-windup, bumpless and conditioned transfer techniques for PID controllers**", To appear in IEEE Control Systems Magazine.

V. Strejc, "**Näherungsverfrahren für aperiodische Übergangscharakteristiken**", Regelungstechnik 7, pp. 124-128, 1959.

B. Thomas, "**A New PID Parameter Tuning Method for Industrial Applications**", IFAC Symposium on Design Methods of Control Systems, ETH Zürich, Switzerland, pp. 42-46, 1991.

D. Vrančić, J. Juričić, "**Transient control by the integration method**", J. Stefan Institute, Report DP-7267, (1995).

D. Vrančić and Y. Peng, "**Amplitude and phase margin detection with on-line PID controller**", J. Stefan Institute, Report DP-7054, (1994).

D. Vrančić, Y. Peng, C. Danz, "**A comparison between different PI controller tuning methods**", J. Stefan Institute, Report DP-7286, (1995).

D. Vrančić, J. Petrovčič, D. Juričić, "**PID parameters setting for the 2nd order processes (in Slovene)**", J. Stefan Institute, Report DP-6782, (1993).