Identification and Self-tuning Control of Time-delay Systems

VLADIMÍR BOBÁL, PETR CHALUPA, MAREK KUBALČÍK, PETR DOSTÁL
Tomas Bata University in Zlín, Faculty of Applied Informatics
Nám. T. G. Masaryka 5555, 7605 Zlín
CZECH REPUBLIC
bobal@fai.utb.cz http://web.fai.utb.cz/cs/docs/CV_english.pdf

Abstract: - Time-delays (dead times) occur in many processes in industry. A Toolbox in the MATLAB/SIMULINK environment was designed for identification and self-tuning control of such processes. The control algorithms are based on modifications of the Smith Predictor (SP). The designed algorithms that are included in the toolbox are suitable not only for simulation purposes but also for implementation in real time conditions. Verification of the designed Toolbox is demonstrated on a self-tuning control of a laboratory heat exchanger in simulation conditions.

Key-Words: - Time-delay; Smith predictor; Process identification; ARX model; Self-tuning control; PID control; Pole assignment; Time-delay Toolbox; Heat exchanger

1 Introduction

The majority of processes in the industrial practice have stochastic characteristics and eventually they exhibit nonlinear behaviour. Traditional controllers with fixed parameters are often unsuitable for such processes because parameters of the process change. One possible alternative for improving the quality of control of such processes is application of adaptive control systems. Different approaches were proposed and utilized. One of the successful approaches is self-tuning control (STC) [1] – [5].

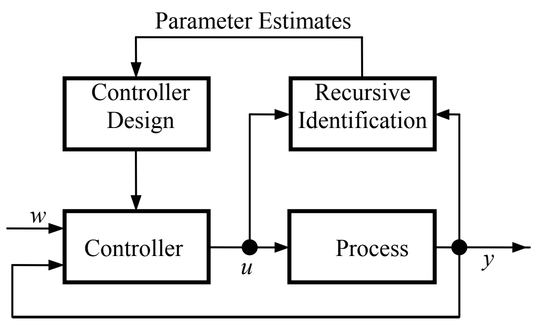

Fig. 1. Self-tuning control system

The block diagram of an STC is shown in Fig. 1, where *y*, *u* and *w* are the process output, the control signal and the reference signal. The main idea of the STC is based on combination of a recursive identification procedure and a particular controller synthesis. The self-tuning strategy was applied for design of control of time-delay systems.

Time-delays appear in many processes in industry and other fields, including economical and biological areas. They are caused by some of the following phenomena [6]:

- the time needed to transport mass, energy or information,
- the accumulation of time lags in a great numbers of low order systems connected in series,
- the required processing time for sensors, such as analyzers; controllers that need some time to implement a complicated control algorithms or processes.

Consider a continuous time dynamical linear SISO (single input $u(t)$ – single output $y(t)$) system with time-delay T_d. The transfer function of a pure transportation lag is $e^{-T_d s}$ where s is complex variable. Overall transfer function with time-delay is in the form

$$G_d(s) = G(s)e^{-T_d s} \tag{1}$$

where $G(s)$ is the transfer function without time-delay. Processes with significant time-delay are difficult to control using standard feedback controllers. When a high performance of the control process is desired or the relative time-delay is very large, a predictive control strategy must be used. The predictive control strategy includes a model of the process in the structure of the controller. The first time-delay compensation algorithm was proposed by Smith 1957 [7]. This control algorithm known as the Smith Predictor (SP) contained a dynamic model of the time-delay process and it can be considered as the first model predictive algorithm.

Although time-delay compensators appeared in the mid 1950s, their implementation with analog technique was very difficult and these were not used in industry. Since 1980s digital time-delay compensators can be implemented. In spite of the fact that all these algorithms are implemented on digital platforms, most works analyze only the continuous case. The digital time-delay compensators are presented e.g. in [8], [9], [10]. Two STC modifications of the digital Smith Predictors (STCSP) are designed in [11] and implemented into MATLAB/SIMULINK Toolbox [12].

The paper is organized in the following way. The principle of the digital Smith Predictor is described in Section 2. Section 3 contains description of the off-line and on-line (recursive) identification procedure. Two modifications of digital controllers that are used for self-tuning versions SPs are proposed in Section 4. The designed Toolbox is briefly described in Section 5. An example of the real-time identification and simulation control of the laboratory heat exchanger contains Section 6. Section 7 concludes the paper.

2 Digital Smith Predictors

The discrete versions of the SP and its modifications are suitable for time-delay compensation in industrial practice. Most of authors designed the digital SP using discrete PID controllers with fixed parameters. However, the SP is more sensitive to process parameter variations and therefore requires an auto-tuning or adaptive approach in many practical applications.

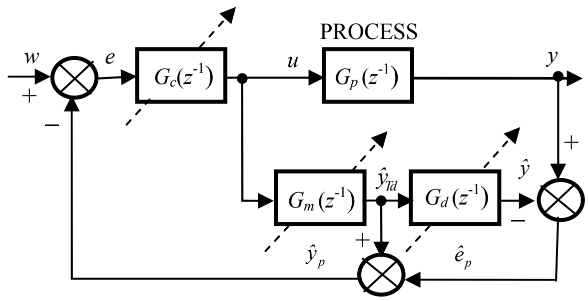

Fig. 2. Block Diagram of a Digital Smith Predictor

The block diagram of a digital SP [13], [14] is shown in Fig. 2. The function of the digital version is similar to the classical analog version. The block $G_m(z^{-1})$ represents process dynamics without the time-delay and is used to compute an open-loop prediction. The difference between the output of the process y and the model including time delay $\hat{y}$ is the predicted error $\hat{e}_p$ as shown in Fig. 2, whereas e and e_s are the error and the noise, respectively and w is the reference signal. If there are no modelling errors or disturbances, the error between the current process output y and the model output $\hat{y}$ will be null. Then the predictor output signal $\hat{y}_p$ will be the time-delay-free output of the process. Under these conditions, the controller $G_c(z^{-1})$ can be tuned, at least in the nominal case, as if the process had no time-delay. The primary (main) controller $G_c(z^{-1})$ can be designed by different approaches (for example digital PID control or methods based on algebraic approach). The outward feedback-loop through the block $G_d(z^{-1})$ in Fig. 2 is used to compensate for load disturbances and modelling errors. The dash arrows indicate the self-tuned parts of the Smith Predictor.

Most industrial processes can be approximated by a reduced order model with a pure time-delay. Consider the following second order linear model with a time-delay

$$G(z^{-1}) = \frac{B(z^{-1})}{A(z^{-1})} z^{-d} = \frac{b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}} z^{-d} \qquad (2)$$

for demonstration of some approaches to the design of the adaptive Smith Predictor. The term z^{-d} represents the pure discrete time-delay. The time-delay is equal to dT_0 where T_0 is the sampling period. Model (2) is used in control algorithms of the designed Toolbox.

3. Identification Procedure

3.1 Identification of Time-delay

In this paper, the time-delay models are obtained separately from an off-line identification using the least squares method (LSM) [15]. The measured process output $y(k)$ is generally influenced by noise. These nonmeasurable disturbances cause errors $\boldsymbol{e}$ in the determination of model parameters and therefore real output vector is in the form

$$\boldsymbol{y} = \boldsymbol{F\Theta} + \boldsymbol{e} \qquad (3)$$

It is possible to obtain the LSM expression for calculation of the vector of the parameter estimates

$$\hat{\boldsymbol{\Theta}} = \left(\boldsymbol{F}^T \boldsymbol{F}\right)^{-1} \boldsymbol{F}^T \boldsymbol{y} \qquad (4)$$

The matrix $\boldsymbol{F}$ has dimension (N-n-d, $2n$), the vector $\boldsymbol{y}$ (N-n-d) and the vector of parameter model estimates $\hat{\boldsymbol{\Theta}}$ ($2n$). N is the number of samples of measured input and output data, n is the model order [16].

Equation (4) serves for calculation of the vector of the parameter estimates $\hat{\boldsymbol{\Theta}}$ using N samples of measured input-output data. The individual vectors and matrices in equations (3) and (4) have the form

$$\boldsymbol{y}^T = \left[y(n+d+1) \quad y(n+d+2) \quad \cdots \quad y(N) \right] \quad (5)$$

$$\boldsymbol{e}^T = \left[\hat{e}(n+d+1) \quad \hat{e}(n+d+2) \quad \cdots \quad \hat{e}(N) \right] \quad (6)$$

$$\hat{\boldsymbol{\Theta}}^T = \left[\hat{a}_1 \quad \hat{a}_2 \quad \cdots \quad \hat{a}_n \quad \hat{b}_1 \quad \hat{b}_2 \quad \cdots \quad \hat{b}_n \right] \quad (7)$$

$$\boldsymbol{F} = \begin{bmatrix} -y(n+d) & -y(n+d-1) & \cdots & -y(d+1) & u(n) & u(n-1) & \cdots & u(1) \\ -y(n+d+1) & -y(n+d) & \cdots & -y(d+2) & u(n+1) & u(n) & \cdots & u(2) \\ \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \cdots & \vdots \\ -y(N-1) & -y(N-2) & \cdots & -y(N-n) & u(N-d-1) & u(N-d-2) & \cdots & u(N-d-n) \end{bmatrix} \quad (8)$$

It is obvious that the quality of time-delay systems identification is very dependent on the choice of a suitable input exciting signal $u(k)$. Therefore the MATLAB function from the System Identification Toolbox

$$u = \text{idinput}(N, type, band, levels)$$

was used. This MATLAB code generates input signals u of different kinds, which are typically used for identification purposes. N determines the number of generated input data. *Type* defines the type of input signal to be generated. This argument takes one of the following values [17]:

type = 'rgs': Gives a random, Gaussian signal.
type = 'rbs': Gives a random, binary signal. This is the default.
type = 'prbs': Gives a pseudorandom, binary signal.
type = 'sine': Gives a signal that is a sum of sinusoids.

The frequency contents of the signal is determined by the argument *band*. For the choices type = 'rs', 'rbs', and 'sine', this argument is a row vector with two entries *band* = [wlow, whigh] that determine the lower and upper bound of the passband. The frequencies wlow and whigh are expressed in fractions of the Nyquist frequency. A white noise character input is thus obtained for *band* = [0 1], which is also the default value.
For the choice type = 'prbs', *band* = [0, B], where B is such that the signal is constant over intervals of length 1/B (the clock period). In this case the default is *band* = [0 1].

The argument *levels* defines the input level. It is a row vector *levels* = [minu, maxu] such that the signal u will always be between the values minu and maxu for the choices type = 'rbs', 'prbs', and 'sine'. For type = 'rgs', the signal level is such that minu is the mean value of the signal, minus one standard deviation, while maxu is the mean value plus one standard deviation. Gaussian white noise with zero mean and variance one is thus obtained for levels = [-1, 1], which is also the default value. The example of exciting input signals of the "idinput" function are depicted in the Fig. 3.

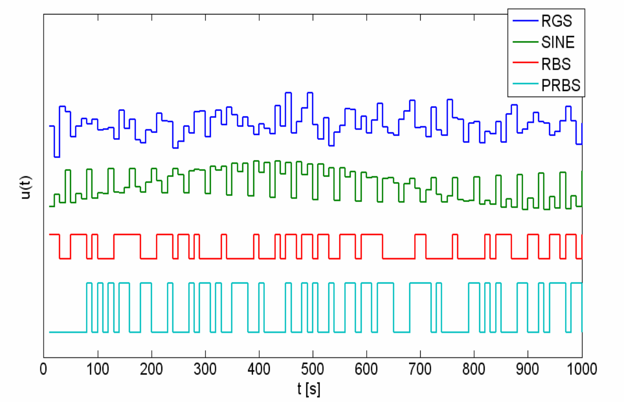

Fig. 3. Example of exciting input signals of "idinput" function

Consider that model (2) is the deterministic part of the stochastic process described by the ARX (regression) model

$$\begin{aligned} y(k) = &-a_1 y(k-1) - a_2 y(k-2) + \\ &+ b_1 y(k-1-d) + b_2 y(k-2-d) + e_s(k) \end{aligned} \quad (9)$$

where $e_s(k)$ is the random nonmeasurable component. The vector of parameter model estimates is computed by solving equation (4)

$$\hat{\boldsymbol{\Theta}}^T(k) = \left[\hat{a}_1 \quad \hat{a}_2 \quad \hat{b}_1 \quad \hat{b}_2 \right] \quad (10)$$

and is used for computation of the prediction output

$$\begin{aligned} \hat{y}(k) = &-\hat{a}_1 y(k-1) - \hat{a}_2 y(k-2) + \\ &\hat{b}_1 u(k-1-d) + \hat{b}_2 u(k-2-d) \end{aligned} \quad (11)$$

The quality of identification can be considered according to error, i.e. the deviation

$$\hat{e}(k) = y(k) - \hat{y}(k) \tag{12}$$

In this paper, the error was used for suitable choice of the time-delay dT_0. The LSM algorithm (4) – (8) is computed for several time-delays dT_0 and the suitable time-delay is chosen according to quality of identification based on the prediction error (12).

For the off-line process identification the MATLAB function from the Optimization Toolbox

$$x = \text{fminsearch}('name\_fce', x_0)$$

was also used. This function find minimum of unconstrained multivariable function using derivative-free method. Algorithm "fminsearch" uses the simplex search method of [18]. This is a direct search method that does not use numerical or analytic gradients.

3.2 Recursive Identification Algorithm

The regression (ARX) model of the following form

$$y(k) = \boldsymbol{\Theta}^T(k)\boldsymbol{\Phi}(k) + e_s(k) \tag{13}$$

is used in the identification part of the designed controller algorithms, where

$$\boldsymbol{\Theta}^T(k) = \begin{bmatrix} a_1 & a_2 & b_1 & b_2 \end{bmatrix} \tag{14}$$

is the vector of model parameters and

$$\boldsymbol{\Phi}^T(k-1) = \left[-y(k-1) - y(k-2)\, u(k-d-1)\, u(k-d-2)\right] \tag{15}$$

is the regression vector. The non-measurable random component $e_s(k)$ is assumed to have zero mean value $E[e_s(k)] = 0$ and constant covariance (dispersion)
$R = E[e_s^2(k)]$.

Both digital adaptive SP controllers use the algorithm of identification based on the Recursive Least Squares Method (RLSM) extended to include the technique of directional (adaptive) forgetting. Numerical stability is improved by means of the LD decomposition [5], [19]. This method is based on the idea of changing the influence of input-output data pairs to the current estimates. The weights are assigned according to amount of information carried by the data.

When using the self-tuning principle, the model parameter estimates must approach the true values right from the start of the control. This means that as the self-tuning algorithm begins to operate, identification must be run from suitable conditions – the result of the possible *a priori* information. The role of suitable initial conditions in recursive identification is often underestimated.

4 Controller Algorithms

4.1 Digital PID Smith Predictor

Hang *et al.* [13], [14] used the Dahlin PID algorithm [20] for the design of the main controller $G_c(z^{-1})$. This algorithm is based on the desired close-loop transfer function in the form

$$G_e(z^{-1}) = \frac{1-e^{-\alpha}}{1-z^{-1}}; \quad \alpha = \frac{T_0}{T_m} \tag{16}$$

where T_m is a desired time constant of the first order closed-loop response. It is not practical to set T_m to be small since it will demand a large control signal $u(k)$ which may easily exceed the saturation limit of the actuator. Then the individual parts of the controller are described by the transfer functions

$$G_c(z^{-1}) = \frac{\left(1-e^{-\alpha}\right)}{\left(1-z^{-1}\right)}\frac{\hat{A}(z^{-1})}{\hat{B}(1)}; \; G_m(z^{-1}) = \frac{z^{-1}\hat{B}(1)}{\hat{A}(z^{-1})}$$

$$G_d(z^{-1}) = \frac{z^{-d}\hat{B}(z^{-1})}{z^{-1}\hat{B}(1)} \tag{17}$$

where $B(1) = \hat{B}(z^{-1})\big|_{z=1} = \hat{b}_1 + \hat{b}_2$.

Since $G_m(z^{-1})$ is the second order transfer function, the main controller $G_c(z^{-1})$ becomes a digital PID controller having the following form:

$$G_c(z^{-1}) = \frac{U(z)}{E(z)} = \frac{q_0 + q_1 z^{-1} + q_2 z^{-2}}{1-z^{-1}} \tag{18}$$

where $q_0 = \gamma, q_1 = \hat{a}_1\gamma, q_2 = \hat{a}_2\gamma$ using by the substitution $\gamma = \left(1-e^{-\alpha}\right)/\hat{B}(1)$. The PID controller output is given by

$$u(k) = q_0 e(k) + q_1 e(k-1) + q_2 e(k-2) + u(k-1) \tag{19}$$

Some simulation experiments using this digital PID SP are presented in [11].

4.2 Digital Pole Assignment (PA) Smith Predictor

The digital pole assignment SP was designed using a polynomial approach in [11]. Polynomial control theory is based on the apparatus and methods of linear algebra (see e.g. [21] - [24]). The design of the controller algorithm is based on the general block scheme of a closed-loop with two degrees of freedom (2DOF) according to Fig. 4 [25].

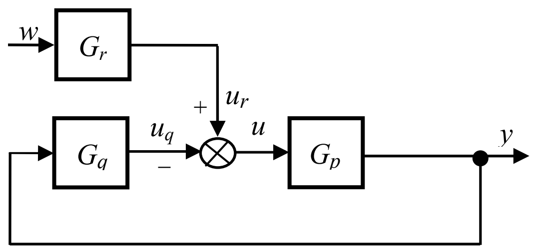

Fig. 4. Block Diagram of a Closed Loop 2DOF Control System

The controlled process is given by the transfer function in the form

$$G_p(z^{-1}) = \frac{Y(z)}{U(z)} = \frac{B(z^{-1})}{A(z^{-1})} \tag{20}$$

where A and B are the second order polynomials. The controller contains the feedback part G_q and the feedforward part G_r. Then the digital controllers can be expressed in the form of discrete transfer functions

$$G_r\left(z^{-1}\right) = \frac{R\left(z^{-1}\right)}{P\left(z^{-1}\right)} = \frac{r_0}{1 + p_1 z^{-1}} \tag{21}$$

$$G_q\left(z^{-1}\right) = \frac{Q\left(z^{-1}\right)}{P\left(z^{-1}\right)} = \frac{q_0 + q_1 z^{-1} + q_2 z^{-2}}{\left(1 + p_1 z^{-1}\right)\left(1 - z^{-1}\right)} \tag{22}$$

According to the scheme presented in Fig. 3 and Equations (20) – (22) it is possible to derive the characteristic polynomial

$$A(z^{-1})P(z^{-1}) + B(z^{-1})Q(z^{-1}) = D(z^{-1}) \tag{23}$$

where

$$D\left(z^{-1}\right) = 1 + d_1 z^{-1} + d_2 z^{-2} + d_3 z^{-3} + d_4 z^{-4} \tag{24}$$

The feedback part of the controller is given by solution of the polynomial Diophantine equation (23). The procedure leading to determination of controller parameters in polynomials Q, R and P (21) and (22) is in [5]. The asymptotic tracking is provided by the feedforward part of the controller given by solution of the polynomial Diophantine equation

$$S(z^{-1})D_w\left(z^{-1}\right) + B(z^{-1})R(z^{-1}) = D(z^{-1}) \tag{25}$$

For a step-changing reference signal value $D_w\left(z^{-1}\right) = 1 - z^{-1}$ holds and S is an auxiliary polynomial which does not enter into controller design and it is possible to solve Equation (25) by substituting $z = 1$

$$R\left(z^{-1}\right) = r_0 = \frac{D(1)}{B(1)} = \frac{1 + d_1 + d_2 + d_3 + d_4}{b_1 + b_2} \tag{26}$$

The 2DOF controller output is given by

$$\begin{aligned} u(k) &= r_0 w(k) - q_0 y(k) - q_1 y(k-1) - \\ &- q_2 y(k-2) + (1 + p_1) u(k-1) + p_1 u(k-2) \end{aligned} \tag{27}$$

The control quality is very dependent on the pole assignment of the characteristic polynomial

$$D(z) = z^4 + d_1 z^3 + d_2 z^2 + d_3 z + d_4 \tag{28}$$

inside the unit circle. The simple method for choice of individual poles is based on the following approach. Consider 1DOF control loop where controlled process (20) with second-order polynomials A and B is controlled using PID controller which is given by transfer function

$$G_q\left(z^{-1}\right) = \frac{Q\left(z^{-1}\right)}{P\left(z^{-1}\right)} = \frac{q_0\left(1 + a_1 z^{-1} + a_2 z^{-2}\right)}{\left(1 - z^{-1}\right)} \tag{29}$$

Substitution of polynomials A, B, Q, P into Equation (23) yields the following relation

$$\begin{aligned} &\hat{A}(z^{-1})\left(1 - z^{-1}\right) + \hat{B}(z^{-1}) q_0 \hat{A}(z^{-1}) = \\ &= \hat{A}(z^{-1})\left[\left(1 - z^{-1}\right) + \hat{B}(z^{-1}) q_0\right] = D(z^{-1}) \end{aligned} \tag{30}$$

where

$$\hat{A}\left(z^{-1}\right) = 1 + \hat{a}_1 z^{-1} + \hat{a}_2 z^{-2}; \quad \hat{B}\left(z^{-1}\right) = \hat{b}_1 z^{-1} + \hat{b}_2 z^{-2} \tag{31}$$

are polynomials with model parameter estimates.

From Equation (30) it is obvious that polynomial

$$A(z) = z^2 + a_1 z + a_2 \tag{32}$$

which have two different real poles α, β, is included in polynomial $D(z)$ (28). Its parameter estimates are known from process identification. Two possibilities are likely to solve using the Time-delay Toolbox.

Pole assignment with user-defined multiple pole (PAMP) method:

Polynomial (24) has two different real poles α, β and user-defined multiple pole γ. Then polynomial (24) has the form

$$D(z) = (z - \alpha)(z - \beta)(z - \gamma)^2$$

and it is possible to express its individual parameters as:

$$\begin{aligned} d_1 &= -(2\gamma + \alpha + \beta) \\ d_2 &= 2\gamma(\alpha + \beta) + \alpha\beta + \gamma^2 \\ d_3 &= -(2\alpha\beta\gamma + \gamma^2(\alpha + \beta)) \\ d_4 &= \alpha\beta\gamma^2 \end{aligned} \tag{33}$$

Pole assignment with user-defined different real poles (PADP) method:
Polynomial (24) has two different real poles α, β and user-defined real poles γ, δ. Then polynomial (24) has the form

$$D(z) = (z-\alpha)(z-\beta)(z-\gamma)(z-\delta)$$

and it is possible to express its individual parameters as:

$$\begin{aligned} d_1 &= -(\alpha + \beta + \gamma + \delta) \\ d_2 &= \alpha\beta + \gamma\delta + (\alpha + \beta)(\gamma + \delta) \\ d_3 &= -\left[(\alpha + \beta)\gamma\delta + (\gamma + \delta)\alpha\beta\right] \\ d_4 &= \alpha\beta\gamma\delta \end{aligned} \tag{34}$$

5 Toolbox Functions

The Toolbox [11] contains three main scripts (**start_PAMP.m, start_PADP.m** and **start_PID.m**) and other programs functions, models and scripts) that are called by these main scripts. These scripts perform similar sequence of operations:

- definition of the controlled system (transfer function, time delay), sample time and controller parameters,
- off-line identification of the controlled system,
- pole assignment control or PID control of the system.

Toolbox files are summarized in Table 1. The detailed instructions for use of the Toolbox are introduced in the User's Guide [12].

A typical control scheme used is depicted in Fig. 5. This scheme is used for systems with time-delay of two sample steps. Individual blocks of the SIMULINK scheme correspond to blocks of the general control scheme presented in Fig. 1. The green blocks represent the controlled system. Constants bc0, ac2, ac1, and ac0 are parameters of a continuous-time system. Blocks Compensator 1 and Compensator 2 are parts of the Smith Predictor and they correspond to $G_m(z^{-1})$ and $G_d(z^{-1})$ blocks of Fig. 2 respectively. The control algorithm is encapsulated in Main Pole Assignment Controller which corresponds to $G_c(z^{-1})$ Fig. 2 block. The Identification block performs the on-line identification of a controlled system and outputs the estimates of the 2nd order ARX model (a1, b1, a2, b2) parameters.

Table 1. Toolbox Files

File	Description
start_PAMP.m	top-level script for pole assignment control (multiple pole γ)
start_PADP.m	top-level script for pole assignment control (poles γ,δ)
start_PID.m	top-level script for PID control
LSM_2or2td.m	off-line identification
Sm_adapt_pp2i.m	computation of control value in pole assignment control scheme SmP_ad_PA.mdl.
sid.m	on-line identification s-function used by both control schemes (SmP_ad_PA.mdl and SmP_ad_PID.mdl)
Ident_c_LSM.mdl	Simulink scheme used to collect data for off-line identification
SmP_ad_PA.mdl	Simulink control scheme of pole assignment control
SmP_ad_PID.mdl	Simulink control scheme of PID control

6 Experimental results

The experimental identification methods and use of the Time-delay Toolbox is demonstrated on a control of laboratory heat exchanger in simulation conditions. The laboratory heat exchanger [26], [27], {28] is based on the principle of transferring heat from a source through a piping system using a heat transferring media to a heat-consuming appliance. A scheme of the laboratory heat exchanger is depicted in Fig. 6.

The heat transferring fluid (e. g. water) is transported using a continuously controllable DC pump (6) into a flow heater (1) with max. power of 750 W. The temperature of a fluid at the heater output T_1 is measured by a platinum thermometer. Warmed liquid then goes through a 15 meters long insulated coiled pipeline (2) which causes the significant delay in the system. The air-water heat exchanger (3) with two cooling fans (4, 5) represents a heat-consuming appliance. The speed of the first fan can be continuously adjusted, whereas the second one is of on/off type. Input and

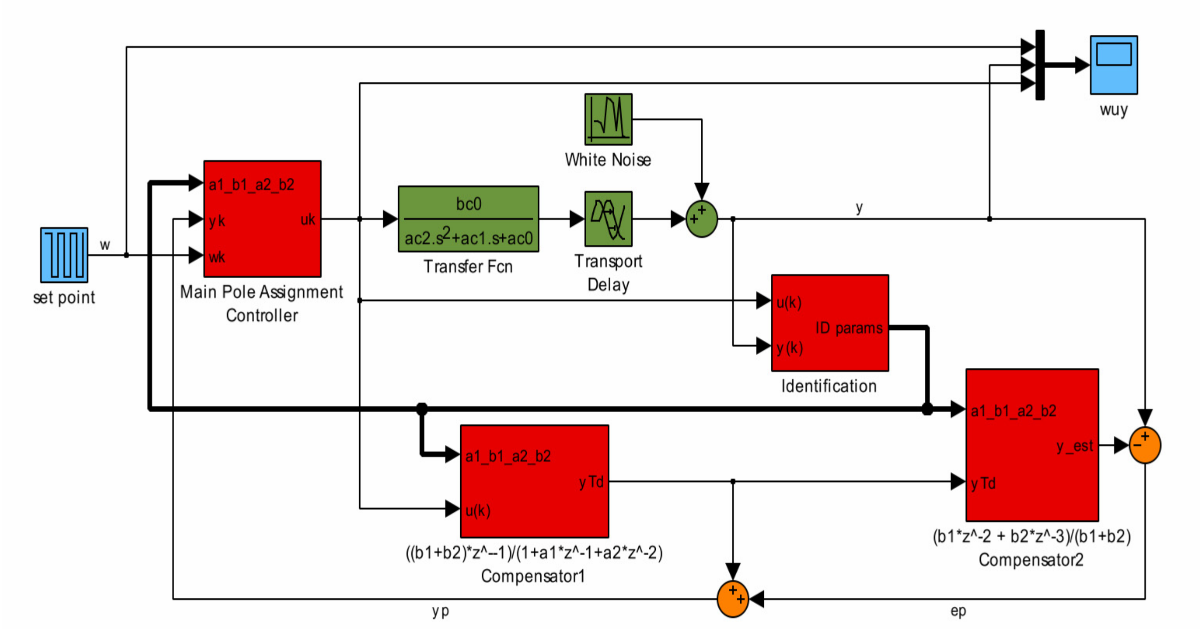

Fig. 5. Simulink control scheme

output temperatures of the cooler are measured again by platinum thermometers as T_2, respective T_3. The laboratory heat exchanger is connected to a standard PC via technological multifunction I/O card. For all monitoring and control functions the MATLAB/SIMULINK environment with Real Time Toolbox.

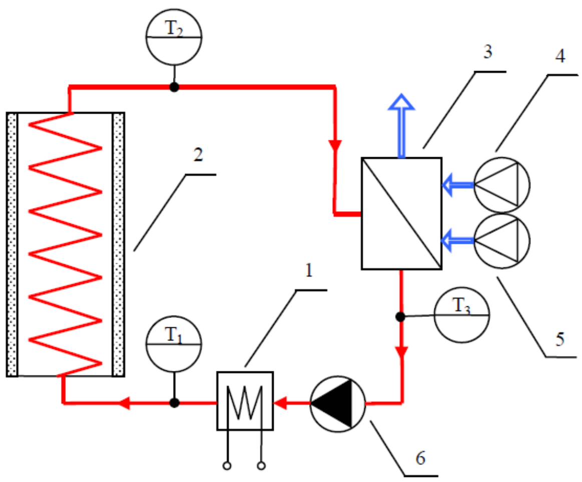

Fig. 6. Scheme of laboratory heat exchanger

6.1 Real-time Identification Experiments

The dynamic model of the laboratory heat exchanger was obtained from processed input (the power of a flow heater P [W]) and output (the temperature of a T_2 [°C]) of the cooler) data. The input signal $u(k)$ was generated using the MATLAB function "idinput" and discrete parameter estimates of model (2) for sampling period T_0 = 100 s and time delay T_d = 200 s were computed using off-line LSM and MATLAB function "fminsearch" (see Paragraph 3.1).

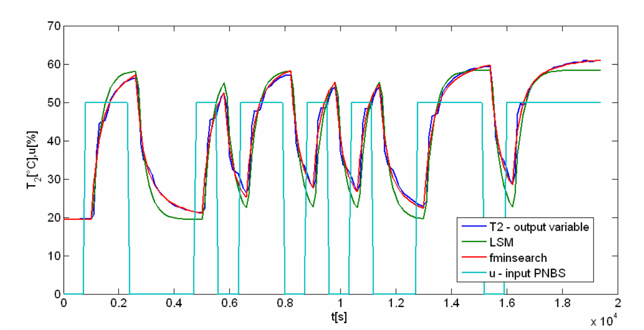

Fig. 7. Identification results: input PNBS

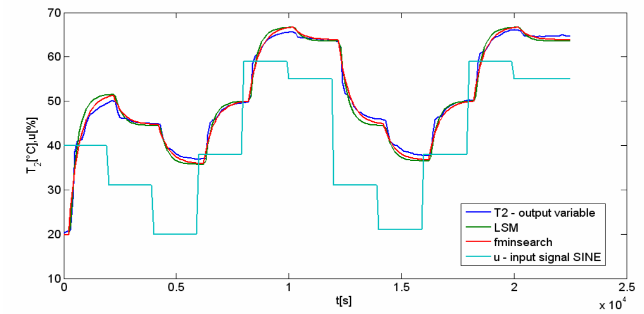

Fig. 8. Identification results: input signal SINE

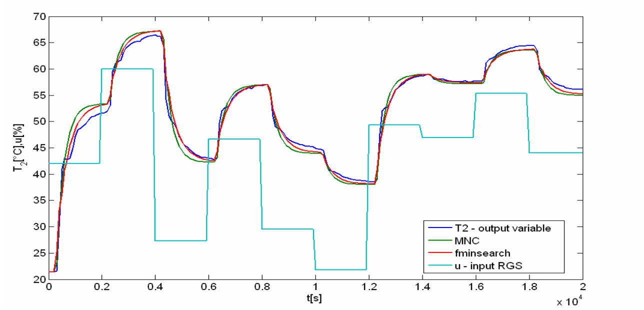

Fig. 9. Identification results: input signal RGS

The graphical variable courses of individual identification experiments are shown in Figs. 7 – 9. The discrete models which were obtained from individual experiments and criterions of identification quality are presented in Tab. 2. From comparison of the real output variable T_2 and the modelled output variables it is obvious that the criterion of identification quality

$$S_y = \frac{1}{N}\sum_{k=1}^{N}\left[y(k)-\hat{y}(k)\right]^2 \tag{35}$$

and the estimate of the static gain

$$\hat{K}_g = \frac{\hat{b}_1+\hat{b}_2}{1+\hat{a}_1+\hat{a}_2} \tag{36}$$

are relatively very good. This fact is confirmed also from courses of unit step responses in Figs. 10 and 11.

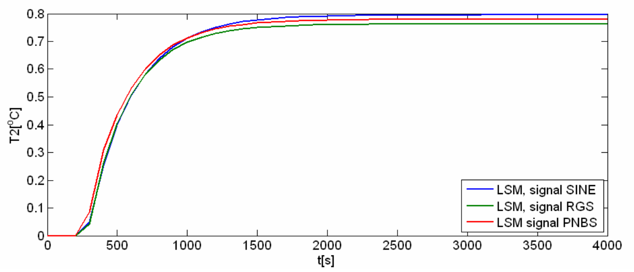

Fig. 10. Comparison of unit step responses, LSM identification

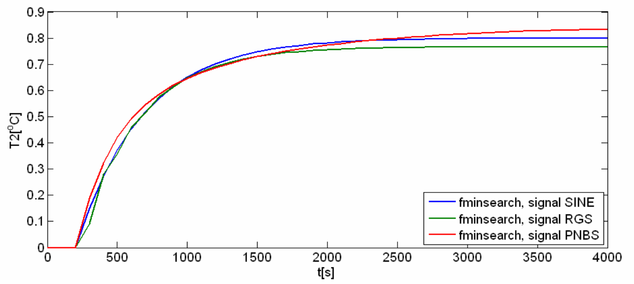

Fig. 11. Comparison of unit step responses, “fminsearch” identification

6.2 Simulation of Closed Control Loops

Simulation is a useful tool for the synthesis of control systems, allowing us not only to create mathematical models of a process but also to design virtual controllers in a computer [29]. The provided mathematical models are close enough to a real object and simulation can be used to verify the dynamic characteristics of control loops when the structure or parameters of the controller have changed. The models of the processes may also be excited by various random noise generators which can simulate the stochastic characteristics of the processes noise signals with similar properties as disturbance signals measured in the machinery can be directly used. The simulation results are valuable for an implementation of a chosen controller (control algorithm) under laboratory and industrial conditions. It must be borne in mind, however, that the practical application of a controller verified by simulation can not be taken as a routine event. Obviously simulation and laboratory conditions can be quite different from those in real plants, and therefore we must verify its practicability with regard to the process dynamics and the required standard of control quality (for example maximum sufferable overshoot, accuracy, settling time, etc.).

For the simulation verification of the proposed control algorithms was chosen the model (43) - see Tab. 2.

$$G\left(z^{-1}\right) = \frac{0.1494z^{-1}+0.028z^{-2}}{1-0.6376z^{-1}-0.1407z^{-2}}z^{-2} \tag{37}$$

Its qualitative identification parameters are the best (see Tab. 2). The followed simulation conditions were chosen for all control experiments: sampling period T_0 = 100 s, time-delay T_d = 200 s, as a random disturbance signal was used the white noise with the mean value μ=0 and the variance σ^2=0.01.

Table 2. Comparison of identification methods and input signals

Identification method	Input signal $u(k)$	Model $G(z^{-1})$	Static gain $\hat{K}$ [°C/%]	Criterion quality S_y
LSM	PNBS	$G\left(z^{-1}\right)=\frac{0.0862z^{-1}+0.1811z^{-2}}{1-0.4934z^{-1}-0.1636z^{-2}}z^{-2}$ (38)	0.7794	10.2726
	RGS	$G\left(z^{-1}\right)=\frac{0.0424z^{-1}+0.1917z^{-2}}{1-0.6445z^{-1}-0.0484z^{-2}}z^{-2}$ (39)	0.7626	1.8761
	SINE	$G\left(z^{-1}\right)=\frac{0.0493z^{-1}+0.1691z^{-2}}{1-0.7063z^{-1}-0.0191z^{-2}}z^{-2}$ (40)	0.7849	1.6583
fminsearch	PNBS	$G\left(z^{-1}\right)=\frac{0.1885z^{-1}-0.1647z^{-2}}{1-1.586z^{-1}-0.6151z^{-2}}z^{-2}$ (41)	0.8197	6.0292
	RGS	$G\left(z^{-1}\right)=\frac{0.0907z^{-1}+0.1708z^{-2}}{1-0.19z^{-1}-0.4689z^{-2}}z^{-2}$ (42)	0.7676	1.3249
	SINE	$G\left(z^{-1}\right)=\frac{0.1494z^{-1}+0.028z^{-2}}{1-0.6376z^{-1}-0.1407z^{-2}}z^{-2}$ (43)	0.7901	1.0963

The initial model parameter estimates were chosen using *a priori* information from previous off-line identification experiment

$$\hat{\boldsymbol{\Theta}}^T(0) = [-0.65 \quad -0.15 \quad 0.15 \quad 0.03].$$

6.2.1 PID Smith Predictor

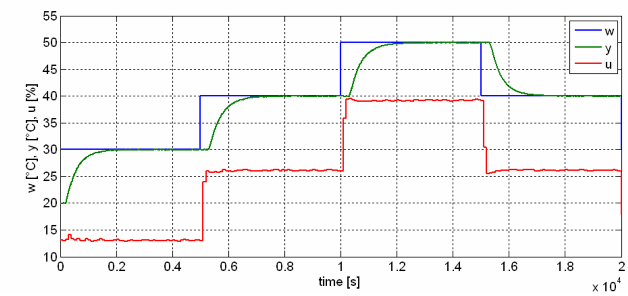

Fig. 11. PID Control of the model (43), T_m = 400 s

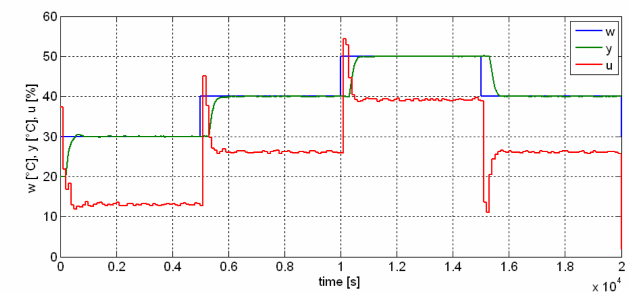

Fig. 12. PID Control of the model (43), T_m = 100 s

The simulation verification of the control model (43) using the main PID controller (19) is shown in Figs. 11 and 12. From these Figs. it is obvious that user-defined time constant T_m influences a speed of the step response and an overshoot of the controller output $u(k)$.

6.2.1 Pole Assignment (PA) Smith Predictor

The simulation verification of the control model (43) using the main PA controller (27) is shown in Figs. 13 and 14.

Simulated control responses when parameters of characteristic polynomial (28) were computed using equations (33) are shown in Fig. 13 and 14. Characteristic polynomials and individual poles are:

Fig. 13:

$$D(z) = z^4 - 1.25z^3 + 0.33z^2 + 0.0315z - 0.0135$$

α = 0.8111; β= -0.1735; $\gamma = \delta$ = 0.3.

Fig. 14:

$$D(z) = z^4 - 0.6576z^3 - 0.1278z^2 + 0.0028z$$

α=0.8111; β=-0.1735; γ= δ=0.01. It is obvious that a small multiple pole leads to $d_4 = 0$.

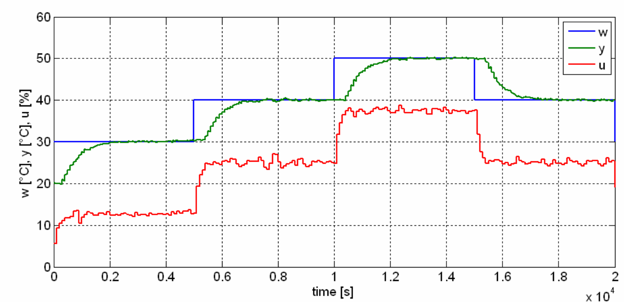

Fig. 13. PAMP Control of the model (43), $\gamma = \delta = 0.3$

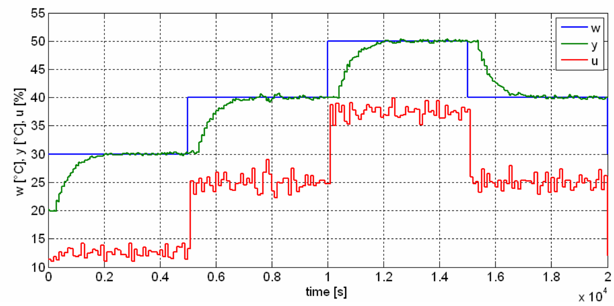

Fig. 14. PAMP Control of the model (43), $\gamma = \delta = 0.01$

Simulated control responses when parameters of characteristic polynomial (28) were computed using equations (34) are shown in Fig. 15. Characteristic polynomial has the form

$$D(z) = z^4 - 0.8876z^3 + 0.0337z^2 + 0.0256z - 0.0021$$

with poles α=0.8111; β=-0.1735; γ=0.1; δ=0.15.

The control quality using main PA controller is very dependent on the pole assignment in the characteristic polynomial (28). The simulation experiments proved that except poles of polynomial $A(z)$ it is suitable to choose next two real positive poles near the centre coordinates. But very small real poles can cause an oscillatory behaviour of the controller output $u(k)$ – see Fig. 14.

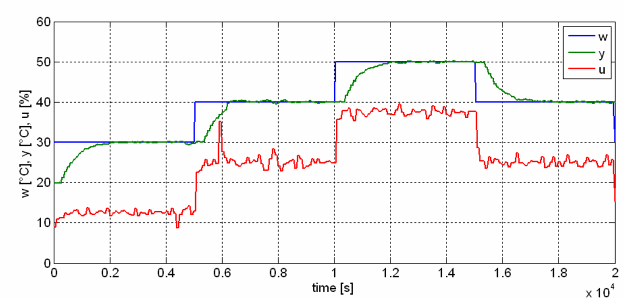

Fig. 15. PADP Control of the model (44), $\gamma = 0.1$, $\delta = 0.15$

7 Conclusion

Two methods for off-line identification with combination of several input exciting signals suitable for time-delay systems were analyzed.

These methods were experimentally verified by an identification of the laboratory heat exchanger in real-time conditions. The best discrete experimental model from view point of the identification quality was used for design of adaptive controllers, which are included in the *MATLAB Toolbox for CAD and Verification of Digital Adaptive Control Time-Delay Systems* [12]. This Toolbox is available free of charge from the Tomas Bata University Zlín Internet site. Both controllers were derived purposely by analytical way (without utilization of numerical methods) to obtain algorithms with easy implementability in industrial practice. The identification part of the adaptive controllers uses the regression ARX model, recursive identification is solved by the Least Squares Method with directional (adaptive) forgetting. Both controllers were successfully verified not only by simulation but also in real-time laboratory conditions for control of the heat exchanger. Very good results were achieved by implementation of the Adaptive Model Predictive Controller in simulation and real-time conditions [30].

Acknowledgment:
This work was supported in part by the Operational Programme Research and Development for Innovations co-funded by the Research Development Fund (ERDF) and national budget of Czech Republic within framework of the Centre of Polymer Systems project (reg. number: CZ. 1.05/2.1.00/03.0111) and the Ministry of Education of the Czech Republic under grant 1M0567.

References:

[1] K. J. Åström and B. Wittenmark, *Adaptive Control*, Massachusetts, Addison-Wesley Publishing Company, 1995.

[2] R. Isermann, K. Lachmann and D. Matko, *Adaptive Control Systems,* Prentice Hall, Inc., Englewood Clils, New Jersey, 1991.

[3] P. E. Welstead and M. B. Zarrop, *Self-Tuning Systems: Control and Signal Processing,* John Wiley, Chichester, 1991.

[4] D. Landau, R. Lozano and M'Saad, *Adaptive Control,* Springer-Verlag, London, 1998.

[5] V. Bobál, J. Böhm J. Fessl and J. Macháček, *Digital Self-tuning Controllers: Algorithms, Implementation and Applications,* Springer-Verlag, London, 2005.

[6] J. E. Normey-Rico, and E. F. Camacho, *Control of Dead-time Processes,* London, Springer-Verlag, 2007.

[7] J. G. Ziegler and N. B. Nichols, Optimum settings for automatic controllers, *Trans. ASME*, Vol. 64, pp. 759-768, 1942.

[8] E. F. Vogel, and T. F. Edgar, A new dead time compensator for digital control, In: *Proc. ISA Annual Conference*, pp. 29-46, Houston, 1980.

[9] Z. J. Palmor, and Y. Halevi, Robustness properties of sampled-data systems with dead time compensators, *Automatica*, Vol. 26, pp. 637-640, 1990.

[10] J. E. Normey-Rico and E. F. Camacho, Dead-time compensators: A unified approach, In: *Proc. of IFAC Workshop on Linear Time-Delay Systems (LDTS'98)*, Grenoble, 1998, pp. 141-146.

[11] V. Bobál, P. Chalupa, P. Dostál and M. Kubalčík, Design and simulation verification of self-tuning Smith Predictors, *International Journal of Mathematics and Computers in Simulation*, Vol. 5, No. 4, pp. 342-351, 2011.

[12] V. Bobál, P. Chalupa, and J. Novák, *Toolbox for CAD and Verfication of Digital Adaptive Control Time-Delay Systems,* Tomas Bata University in Zlín, Faculty of Applied Informatics, 2011. Available from http://nod32.fai.utb.cz/promotion/Software_OBD/Time_Delay_Tool.zip.

[13] C. C Hang, K. W. Lim and B. W. Chong, A dual-rate adaptive digital Smith predictor, *Automatica*, Vol. 20, No. 1, pp. 1-16, 1989.

[14] C. C Hang, H. L. Tong, and K. H. Weng, *Adaptive Control*, North Carolina, Instrument Society of America, 1993.

[15] L. Ljung, *System Identification: Theory for the User,* Englewood Cliffs, NJ, Prentice-Hall, 1987.

[16] V. Bobál, M. Kubalčík, P. Chalupa and P. Dostál, Identification and digital control of higher-order processes, In: *Proc. of the 26th European Control Conference on Modelling and Simulation*, Koblenz, Germany, 2012, pp. 426-433.

[17] L. Ljung, *System Identification Toolbox^{TM}*, User's Guide, The MathWorks, Natick, MA, 2012.

[18] J. C Lagaris, J. A Reeds, M. H. Wright and P. E. Wright, Convergence properties of the Nelder-Mead simplex method in low dimensions, *SIAM Journal of Optimization*, Vol. 9, No. 1, pp. 112-147, 1998.

[19] R. Kulhavý, Restricted exponential forgetting in real time identification, *Automatica*, Vol. 23, pp. 586-600, 1987.

[20] D. B. Dahlin, Designing and tuning digital controllers, *Inst. Control Systems*, Vol. 42, pp.77-73, 1968.
[21] V. Kučera, *Discrete Linear Control: the Polynomial Equation Approach,* Chichester, John Wiley, 1979.
[22] M. Vidyasagar, *Control System Synthesis: A Factorization Approach,* Cambridge M.A., MIT Press, 1985.
[23] V. Kučera, *Analysis and Design of Discrete Linear Control Systems,* London, Prentice Hall, 1991.
[24] V. Kučera, Diophantine equations in control – a survey, *Automatica*, Vol. 29, pp. 1361-1375, 1993.
[25] T. Kawabe, Robust 2DOF PID controller design of time-delay systems based on evolutionary computation, In: *Proceedings of 4th International Conference on ELACTRONICS, CONTROL and SIGNAL PROCESSING*, Miami, Florida, USA, 2005, pp. 144-149.
[26] Dostálek, P., Vašek, V. and J. Dolinay, Design and implementation of portable data acquisition unit in process control and supervision applications, *WSEAS Transactions on Systems and Control*, Vol. 3, No. 9, 2008, pp. 779-788.
[27] L. Pekař, R. Prokop and P. Dostálek, Circuit heating plant model with internal delays, *WSEAS Transactions on Systems and Control*, Vol. 8, No. 9, 2009, pp. 1093-1104.
[28] L. Pekař, R. Prokop and P. Dostálek, An anisochronic model of a laboratory heating system. In: *Proceedings of 13th WSEAS International Conference on Systems*, Rhodes, Greece, 2009, pp. 165-172.
[29] F. Neri, *Agent Based Modeling under Partial and Full Knowledge Learning Settings to Simulate Financial Markets*, AI Communications, IOS Press, 2012 (in print).
[30] V. Bobál, M. Kubalčík, P. Dostál and J. Matějíček, Adaptive predictive control of time-delay systems, In: Nostradamus - Conference on Prediction, Modeling and Analysis of Complex System Dynamics, Ostrava, Czech Republic, 2012 (accepted).