Predictive Control of Higher Order Systems Approximated by Lower Order Time-Delay Models

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

Abstract: - In technical practice often occur higher order processes when a design of an optimal controller leads to complicated control algorithms. One of possibilities of control of such processes is their approximation by lower-order model with time-delay (dead time). The contribution is focused on a choice of a suitable experimental identification method and a suitable excitation input signals for an estimation of process model parameters with time-delay. One of the possible approaches to control of time-delay processes is application of model-based predictive control (MPC) methods. The further contribution is design of an algorithm for predictive control of high-order processes which are approximated by second-order model of the process with time-delay. The controller was tested and verified by control of several simulation models and a model of a laboratory heat exchanger.

Key-Words: - predictive control, time-delay systems, digital control, higher order systems, simulation

1 Introduction

Some technological processes in industry are characterized by high-order dynamic behaviour or large time constants and time-delays. Time-delay in a process increases the difficulty of controlling it. However using the approximation of higher-order process by lower-order model with time-delay provides simplification of the control algorithms. Let us 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 a 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. Methods and applications of control of time-delay systems are for example in [1], [2], [3].

Processes with time-delay are difficult to control using standard feedback controllers. One of the possible approaches to control processes with time delay is predictive control [4], [5], [6]. 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 [8]. This control algorithm known as the Smith Predictor (SP) contains a dynamic model of the time-delay process and it can be considered as the first model predictive algorithm. An alternative method implemented to analyze heat diffusion system with time–delay, are the integer and fractional order controllers with a Smith Predictor controller [9].

Model Predictive Control (MPC) or only Predictive Control is one of the control methods which have developed considerably over a few past years. Predictive control is essentially based on discrete or sampled models of processes. Computation of appropriate control algorithms is then realized namely in the discrete domain.

The term Model Predictive Control designates a class of control methods which have common particular attributes [10], [11].

- Mathematical model of a system is used for prediction of future systems output.
- The input reference trajectory in future is known.
- A computation of the future control sequence includes minimization of an appropriate objective function (usually quadratic one) with the future trajectories of control increments and control errors.
- Only the first element of the control sequence is applied and the whole procedure of the objective function minimization is repeated in the next sampling period.

The principle of MPC is shown in Fig. 1, where $u(t)$ is the manipulated variable, $y(t)$ is the process output and $w(t)$ is the reference signal, N_1, N_2 and N_u are called minimum, maximum and control horizon. This principle is possible to define as follows:

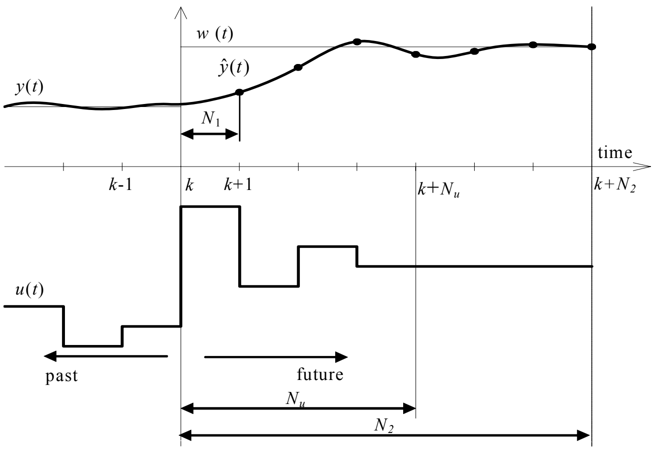

Fig. 1. Principle of MPC

1. The process model is used to predict the future outputs $\hat{y}(t)$ over some horizon *N*. The predictions are calculated based on information up to time *k* and on the future control actions that are to be determined.
2. The future control trajectory is calculated as a solution of an optimisation problem consisting of an objective function and constraints. The cost function comprises future output predictions, future reference trajectory, and future control actions.
3. Although the whole future control trajectory was calculated in the previous step, only first element $u(k)$ is actually applied to the process. At the next sampling time the procedure is repeated. This is known as the *Receding Horizon* concept.

Theoretical research in the area of predictive control has a great impact on the industrial world and there are many applications of predictive control in industry. Its development has been significantly influenced by industrial practice. At present, predictive control with a number of real industrial applications belongs among the most often implemented modern industrial process control approaches. First predictive control algorithms were implemented in industry as an effective tool for control of multivariable industrial processes with constraints more than twenty five years ago. The use of predictive control was limited on control of namely rather slow processes due to the amount of computation required. At present, with the computing power available today, this is not an essential problem. A fairly actual and extensive surveys of industrial applications of predictive control are presented in [12], [13], [14].

High-order processes are largely approximated by the FOTD (first-order-time-delay) model. The aim of the paper is implementation of a predictive controller for control of high-order processes which are approximated by second-order model with time delay of two steps. This model approximates the higher order dynamics more accurately than the first order time delay model whilst design of control algorithms is still quite simple. The designed controller was tested and verified by control of several simulation models and a model of a laboratory heat exchanger.

The paper is organized as follows: section 2 describes identification of time-delay processes; section 3 presents design and implementation of predictive control; section 4 introduces computation of predictor for time-delay systems; section 5 gives the simulation results; section 6 contains experimental results and finally section 7 concludes the paper.

2 Identification of Time-Delay Processes

In this paper, the time-delay model is obtained separately from an off-line identification using the least squares method (LSM). 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

$$\boldsymbol{y} = \boldsymbol{F\Theta} + \boldsymbol{e} \tag{2}$$

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

$$\hat{\boldsymbol{\Theta}} = \left(\boldsymbol{F}^T\boldsymbol{F}\right)^{-1}\boldsymbol{F}^T\boldsymbol{y} \tag{3}$$

The matrix $\boldsymbol{F}$ has dimension (N-n-d, $2n$) and rank 2, 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, d is a number of time-delay steps.

Equation (3) 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 (2) and (3) have the form

$$\hat{\boldsymbol{\Theta}}^T = \begin{bmatrix} \hat{a}_1 & \hat{a}_2 & \cdots & \hat{a}_n & \hat{b}_1 & \hat{b}_2 & \cdots & \hat{b}_n \end{bmatrix} \tag{4}$$

$$\boldsymbol{y}^T = \begin{bmatrix} y(n+d+1) & y(n+d+2) & \cdots & y(N) \end{bmatrix} \tag{5}$$

$$\boldsymbol{e}^T = \begin{bmatrix} \hat{e}(n+d+1) & \hat{e}(n+d+2) & \cdots & \hat{e}(N) \end{bmatrix} \tag{6}$$

$$\boldsymbol{F} = \begin{bmatrix} -y(n+d) & -y(n+d-1) & \cdots & -y(d+1) & u(n) & u(n-1) & \cdots & u(1) \\ -y(n+d+1) & -y(n+d) & \cdots & -y(d+2) & u(n+1) & u(n) & \cdots & u(2) \\ \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \cdots & \vdots \\ -y(N-1) & -y(N-2) & \cdots & -y(N-n) & u(N-d-1) & u(N-d-2) & \cdots & u(N-d-n) \end{bmatrix} \tag{7}$$

Most of higher-order industrial processes can be approximated by a model of reduced order with pure time-delay. Let us consider the following second order linear model with a time-delay

$$G_d\left(z^{-1}\right) = \frac{B\left(z^{-1}\right)}{A\left(z^{-1}\right)} z^{-d} = \frac{b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}} z^{-d} \tag{8}$$

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.

Our experience proved that quality of system identification when the higher-order process is identified by the lower-order model is very dependent on the choice of an input excitation signal $u(k)$. The best results were achieved using a Random Gaussian Signal (RGS).

Let us consider that model (8) is the deterministic part of the stochastic process described by the ARX (regression) model

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

where $e_s(k)$ is the non-measurable random component. The vector of parameter model estimates is computed by solving equation (3)

$$\hat{\boldsymbol{\Theta}}^T(k) = \begin{bmatrix} \hat{a}_1 & \hat{a}_2 & \hat{b}_1 & \hat{b}_2 \end{bmatrix} \tag{10}$$

and is used for computation of the prediction output.

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

The quality of identification can be considered according to error, i.e. the difference between the measured and modeled value of the systems output

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

In this paper, a suitable choice of the number of time-delay steps was performed according to the error. The LSM algorithm (3) – (7) is computed for several numbers of time-delays steps and a suitable time-delay is chosen according to quality of identification based on the prediction error (12).

2.1 Stable Process

Let us consider the following stable fifth order linear system

$$G_A(s) = \frac{2}{(s+1)^5} = \frac{2}{s^5 + 5s^4 + 10s^3 + 10s^2 + 5s + 1} \tag{13}$$

The system (13) was identified by the discrete model (11) using off-line LSM (3) – (6) for different numbers of time-delay steps. As the input signal was used the Random Gaussian Signal (RGS). A criterion of the identification quality is based on sum of squares of error

$$J_{\hat{e}^2}(d) = \sum_{k=1}^{N} \hat{e}^2(k) \tag{14}$$

This criterion evaluates accuracy of the identification process. From Fig 2. , it is obvious that value of the criterion (14) decreases with increasing number of time-delay steps *d*. This is caused by the fact that the increasing of the number of time-delay steps improves estimation of the static gain

$$\hat{K}_g = \frac{\hat{b}_1 + \hat{b}_2}{1 + \hat{a}_1 + \hat{a}_2} \tag{15}$$

The difference between estimates of the static gain $\hat{K}_g$ of the discrete model (8) and the continuous-time model (13) plays an important role for the quality of identification because the identification time was relatively long (300 s) with regard to the response time (about 15 s).

The system was identified by the following model

$$G_A\left(z^{-1}\right)=\frac{-0.0424z^{-1}+0.0296z^{-2}}{1-1.6836z^{-1}+0.7199z^{-2}}z^{-d} \qquad (16)$$

Comparisons of step responses of continuous-time (13) and discrete models (16) with sampling period $T_0 = 0.5$ s for different numbers of time-delay steps d are shown in Figs. 3-5, where yc is the step response of the model (13) and yd are step responses of the discrete model (16) for individual numbers of time-delay steps d.

From Figs. 2-5 it results that a suitable model for the design of the predictive controller is the model (13) with d = 2. Its structure is simple and it relatively well approximates the dynamic behaviour of the continuous-time model (16).

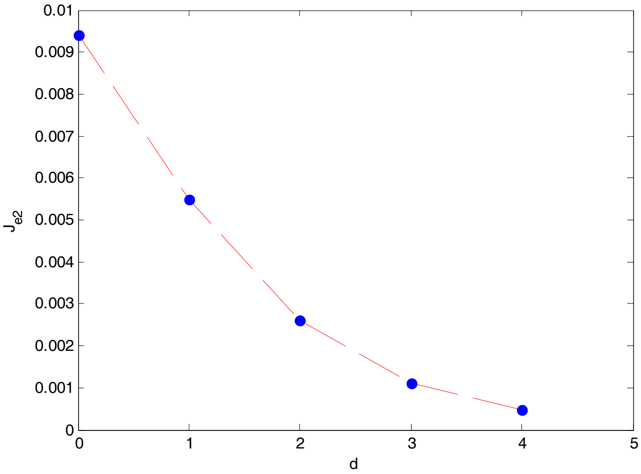

Fig. 2. Criterion of quality identification for $d \in [0,5]$

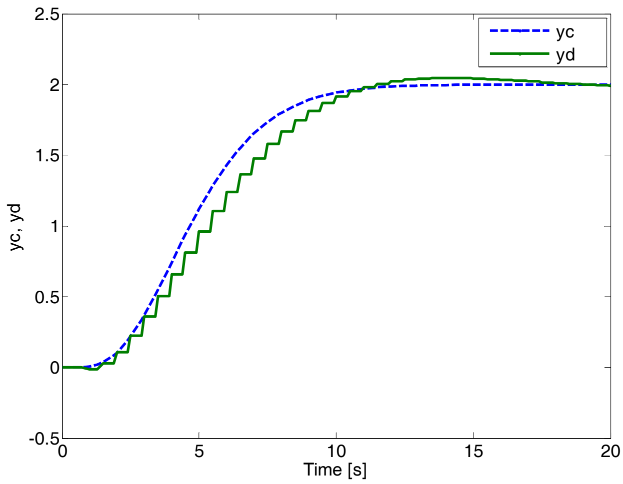

Fig. 3. Comparison of step responses yc, yd for d =0 (process (13))

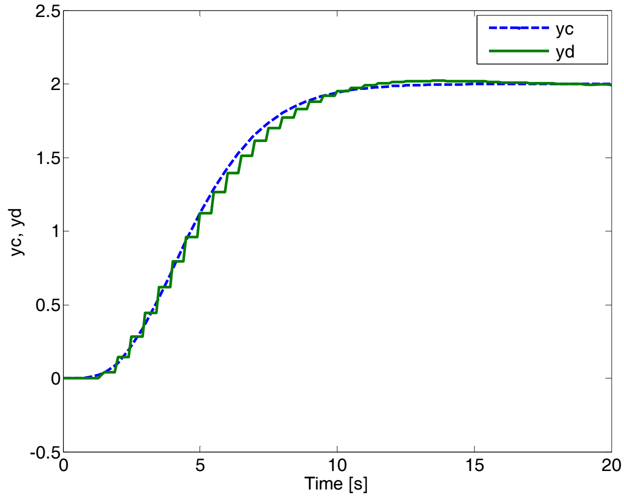

Fig. 4. Comparison of step responses yc, yd for d =2 (process (13))

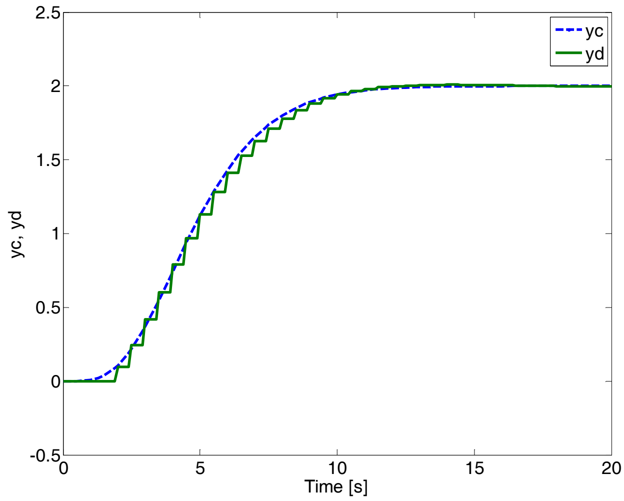

Fig. 5. Comparison of step responses yc, yd for d =3 (process (13))

2.2 Stable Non-Minimum Phase Process

Let us consider the following fifth-order linear system with non-minimum phase

$$G_B(s)=\frac{2(1-5s)}{s^5+5s^4+10s^3+10s^2+5s+1} \qquad (17)$$

The process (17) was identified by the model (8) with a time-delay d=2 and sampling period $T_0 = 0{,}5s$. The discrete model which was obtained from the model (17) by Z-transform is in the following form

$$G_B\left(z^{-1}\right)=\frac{-0.7723z^{-1}+0.8514z^{-2}}{1-1.6521z^{-1}+0.8514z^{-2}}z^{-2} \qquad (18)$$

The comparison of the step responses of the continuous-time model (17) and the discrete model (18) is shown in Fig. 6.

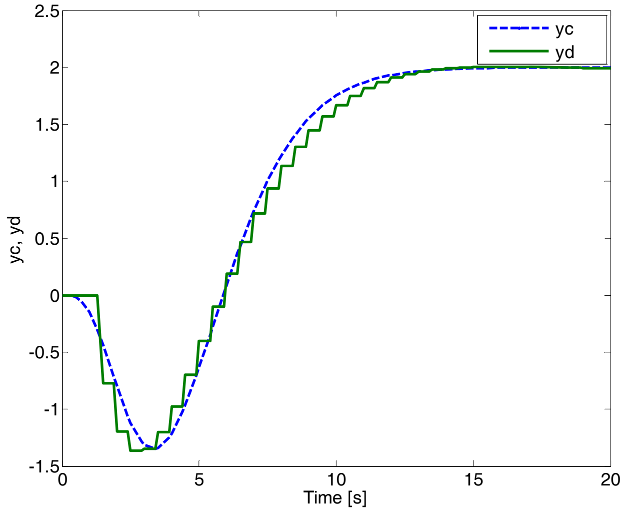

Fig. 6. Comparison of step responses yc, yd for d =2 (process (17))

3 Implementation of Predictive Control

In this Section, GPC (General predictive control) will be briefly described. The GPC method is in principle applicable to both SISO and MIMO processes and is based on input-output models. The standard cost function used in GPC contains quadratic terms of (possible filtered) control error and control increments on a finite horizon into the future

$$J=\sum_{i=N_1}^{N_2}\delta(i)\left[\hat{y}(k+i)-w(k+i)\right]^2+\sum_{i=1}^{N_u}\left[\lambda(i)\Delta u(k+i-1)\right]^2 \quad (19)$$

where $\hat{y}(k+i)$ is the process output of i steps in the future predicted on the base of information available upon the time k, $w(k+1)$ is the sequence of the reference signal and $\Delta u(k+i-1)$ is the sequence of the future control increments that have to be calculated.

Implicit constraints on Δu are placed between N_u and N_2 as

$$\Delta u(k+i-1)=0, \qquad N_u < i \le N_2 \quad (20)$$

The parameters $\delta(i)$ and $\lambda(i)$ are sequences which affect future behaviour of the controlled process. Generally, they are chosen in the form of constants or exponential weights, according to our requirements on control.

3.1 Calculation of the Optimal Control

The objective of predictive control is a computation of a sequence of future increments of the manipulated variable $[\Delta u(k), \Delta u(k+1), \ldots]$ so that the criterion (19) was minimized. For further computation, it is necessary to transform the criterion (19) to a matrix form.

The output of the model (predictor) is computed as the sum of the free response $\boldsymbol{y}_0$ and the forced response of the model $\boldsymbol{y}_n$

$$\hat{\boldsymbol{y}}=\boldsymbol{y}_n+\boldsymbol{y}_0 \quad (21)$$

It is possible to compute the forced response as the multiplication of the matrix $\boldsymbol{G}$ (Jacobian of the model) and the vector of future control increments $\Delta\boldsymbol{u}$, which is generally a priori unknown

$$\boldsymbol{y}_n=\boldsymbol{G}\Delta\boldsymbol{u} \quad (22)$$

where

$$\boldsymbol{G}=\begin{bmatrix} g_1 & 0 & 0 & \cdots & 0 \\ g_2 & g_1 & 0 & \cdots & 0 \\ g_3 & g_2 & g_1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ g_{N_2} & g_{N_2-1} & g_{N_2-2} & \cdots & g_{N_2-N_u+1} \end{bmatrix} \quad (23)$$

is a matrix containing values of the step sequence.

It follows from equations (21) and (22) that the predictor in a vector form is given by

$$\hat{\boldsymbol{y}}=\boldsymbol{G}\Delta\boldsymbol{u}+\boldsymbol{y}_0 \quad (24)$$

and the cost function (19) can be modified to the form below

$$\begin{aligned} J &= (\hat{\boldsymbol{y}}-\boldsymbol{w})^T(\hat{\boldsymbol{y}}-\boldsymbol{w})+\lambda\Delta\boldsymbol{u}^T\Delta\boldsymbol{u}= \\ &= (\boldsymbol{G}\Delta\boldsymbol{u}+\boldsymbol{y}_0-\boldsymbol{w})^T(\boldsymbol{G}\Delta\boldsymbol{u}+\boldsymbol{y}_0-\boldsymbol{w})+\lambda\Delta\boldsymbol{u}^T\Delta\boldsymbol{u} \end{aligned} \quad (25)$$

where $\boldsymbol{w}$ is the vector of future reference trajectory.

Minimisation of the cost function (25) now becomes a direct problem of linear algebra. The solution in an unconstrained case can be found by setting partial derivative of J with respect to $\Delta\boldsymbol{u}$ to zero and yields

$$\Delta\boldsymbol{u}=-\left(\boldsymbol{G}^T\boldsymbol{G}+\lambda\boldsymbol{I}\right)^{-1}\boldsymbol{G}^T(\boldsymbol{y}_0-\boldsymbol{w}) \quad (26)$$

where the gradient $\boldsymbol{g}$ and Hessian $\boldsymbol{H}$ are defined as

$$\boldsymbol{g}^T=\boldsymbol{G}^T(\boldsymbol{y}_0-\boldsymbol{w}) \quad (27)$$

$$\boldsymbol{H}=\boldsymbol{G}^T\boldsymbol{G}+\lambda\boldsymbol{I} \quad (28)$$

Equation (26) gives the whole trajectory of the future control increments and such is an open-loop strategy. To close the loop, only the first element $\boldsymbol{u}$, e. g. $\Delta u(k)$ is applied to the system and the whole algorithm is recomputed at time k+1. This strategy is called the *Receding Horizon Principle* and is one of the key issues in the MBPC concept.

If the first row of the matrix $\left(\boldsymbol{G}^T\boldsymbol{G}+\lambda\boldsymbol{I}\right)^{-1}\boldsymbol{G}^T$ is denoted as $\boldsymbol{K}$ then the actual control increment can be calculated as

$$\Delta u(k) = \boldsymbol{K}\left(\boldsymbol{w}-\boldsymbol{y}_0\right) \tag{29}$$

4 Computation of Predictor

An important task is computation of predictions for arbitrary prediction and control horizons. Dynamics of most of processes requires horizons of length where it is not possible to compute predictions in a simple straightforward way. Recursive expressions for computation of the free response and the matrix $\boldsymbol{G}$ in each sampling period had to be derived. There are several different ways of deriving the prediction equations for transfer function models. Some papers make use of Diophantine equations to form the prediction equations (e.g. [14]). In [10] matrix methods are used to compute predictions. We derived a method for recursive computation of both the free response and the matrix of the dynamics [15].

Computation of the predictor for the time-delay system can be obtained by modification of the predictor for the corresponding system without a time-delay. At first we will consider the second order system without time-delay and then we will modify the computation of predictions for the time-delay system.

4.1 Second Order System without Time-Delay

The model is described by the transfer function

$$G\left(z^{-1}\right)=\frac{b_1z^{-1}+b_2z^{-2}}{1+a_1z^{-1}+a_2z^{-2}}=\frac{B\left(z^{-1}\right)}{A\left(z^{-1}\right)} \tag{30}$$

$$A\left(z^{-1}\right)=1+a_1z^{-1}+a_2z^{-2};\; B\left(z^{-1}\right)=b_1z^{-1}+b_2z^{-2} \tag{31}$$

The model can be also written in the form

$$A\left(z^{-1}\right)y(k)=B\left(z^{-1}\right)u(k) \tag{32}$$

A widely used model in general model predictive control is the CARIMA model which we can obtain from the nominal model (32) by adding a disturbance model

$$A\left(z^{-1}\right)y(k)=B\left(z^{-1}\right)u(k)+\frac{C\left(z^{-1}\right)}{\Delta}n_c(k) \tag{33}$$

where $n_c(k)$ is a non-measurable random disturbance that is assumed to have zero mean value and constant covariance and the operator delta is $1-z^{-1}$. Inverted delta is then an integrator.

The polynomial $C\left(z^{-1}\right)$ will be further considered as $C\left(z^{-1}\right)=1$. The CARIMA description of the system is then in the form

$$\Delta A\left(z^{-1}\right)y(k)=B\left(z^{-1}\right)\Delta u(k-1)+n_c(k) \tag{34}$$

The difference equation of the CARIMA model without the unknown term $n_c(k)$ can be expressed as:

$$\begin{aligned} y(k)&=(1-a_1)y(k-1)+(a_1-a_2)y(k-2)+a_2y(k-3)+\\ &+b_1\Delta u(k-1)+b_2\Delta u(k-2) \end{aligned} \tag{35}$$

It was necessary to compute three step ahead predictions in straightforward way by establishing of lower predictions to higher predictions. The model order defines that computation of one step ahead prediction is based on three past values of the system output. The three step ahead predictions are as follows

$$\begin{aligned} \hat{y}(k+1)&=(1-a_1)y(k)+(a_1-a_2)y(k-1)+a_2y(k-2)+\\ &+b_1\Delta u(k)+b_2\Delta u(k-1)\\ \hat{y}(k+2)&=(1-a_1)y(k+1)+(a_1-a_2)y(k)+a_2y(k-1)+\\ &+b_1\Delta u(k+1)+b_2\Delta u(k)\\ \hat{y}(k+3)&=(1-a_1)y(k+2)+(a_1-a_2)y(k+1)+a_2y(k)+\\ &+b_1\Delta u(k+2)+b_2\Delta u(k+1) \end{aligned} \tag{36}$$

The predictions after modification can be written in a matrix form

$$\begin{aligned} &\begin{bmatrix}\hat{y}(k+1)\\ \hat{y}(k+2)\\ \hat{y}(k+3)\end{bmatrix}=\begin{bmatrix}g_1 & 0\\ g_2 & g_1\\ g_3 & g_2\end{bmatrix}\begin{bmatrix}\Delta u(k)\\ \Delta u(k+1)\end{bmatrix}+\begin{bmatrix}p_{11} & p_{12} & p_{13} & p_{14}\\ p_{21} & p_{22} & p_{23} & p_{24}\\ p_{31} & p_{32} & p_{33} & p_{34}\end{bmatrix}\begin{bmatrix}y(k)\\ y(k-1)\\ y(k-2)\\ \Delta u(k-1)\end{bmatrix}=\\ &=\begin{bmatrix}b_1 & 0\\ b_1(1-a_1)+b_2 & b_1\\ (a_1-a_2)b_1+(1-a_1)^2b_1+(1-a_1)b_2 & b_1(1-a_1)+b_2\end{bmatrix}\begin{bmatrix}\Delta u(k)\\ \Delta u(k+1)\end{bmatrix}+\\ &+\begin{bmatrix}(1-a_1) & (a_1-a_2) & a_2 & b_2\\ (1-a_1)^2+(a_1-a_2) & (1-a_1)(a_1-a_2)+a_2 & a_2(1-a_1) & b_2(1-a_1)\\ (1-a_1)^3+2(1-a_1)(a_1-a_2)+a_2 & (1-a_1)^2(a_1-a_2)+a_2(1-a_1)+(a_1-a_2)^2 & a_2(1-a_1)^2+(a_1-a_2)a_2 & b_2(1-a_1)^2+(a_1-a_2)b_2\end{bmatrix}\begin{bmatrix}y(k)\\ y(k-1)\\ y(k-2)\\ \Delta u(k-1)\end{bmatrix} \end{aligned} \tag{37}$$

It is possible to divide computation of the predictions to recursion of the free response and recursion of the matrix of the dynamics. Based on the three previous predictions it is repeatedly computed the next row of the free response matrix in the following way:

$$\begin{aligned} p_{41} &= (1-a_1)p_{31} + (a_1-a_2)p_{21} + a_2 p_{11} \\ p_{42} &= (1-a_1)p_{32} + (a_1-a_2)p_{22} + a_2 p_{12} \\ p_{43} &= (1-a_1)p_{33} + (a_1-a_2)p_{23} + a_2 p_{13} \\ p_{44} &= (1-a_1)p_{34} + (a_1-a_2)p_{24} + a_2 p_{14} \end{aligned} \tag{38}$$

The first row of the matrix is omitted in the next step and further prediction is computed based on the three last rows including the one computed in the previous step. This procedure is cyclically repeated. It is possible to compute an arbitrary number of rows of the matrix.

The recursion of the dynamics matrix is similar. The next element of the first column is repeatedly computed in the same way as in the previous case and the remaining columns are shifted to form a lower triangular matrix in the way which is obvious from the equation (37). This procedure is performed repeatedly until the prediction horizon is achieved. If the control horizon is lower than the prediction horizon a number of columns in the matrix is reduced. Computation of the new element is performed as follows:

$$g_4 = (1-a_1)g_3 + (a_1-a_2)g_2 + a_2 g_1 \tag{39}$$

4.2 Second Order System with Time-Delay

The nominal model with two steps time-delay is considered as

$$G(z^{-1}) = \frac{B(z^{-1})}{A(z^{-1})} z^{-2} = \frac{b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}} z^{-2} \tag{40}$$

The CARIMA model for time-delay system takes the form

$$\Delta A(z^{-1}) y(k) = z^{-d} B(z^{-1}) \Delta u(k-1) + n_c(k) \tag{41}$$

where d is the dead time. In our case d is equal to 2. In order to compute the control action it is necessary to determine the predictions from d+1 (2+1 in our case) to $d+N_2$ $(2+N_2)$.

The predictor (37) is then modified to

$$\begin{bmatrix} \hat{y}(k+3) \\ \hat{y}(k+4) \\ \hat{y}(k+5) \end{bmatrix} = \begin{bmatrix} p_{31} & p_{32} & p_{33} \\ p_{41} & p_{42} & p_{43} \\ p_{51} & p_{52} & p_{53} \end{bmatrix} \begin{bmatrix} y(k) \\ y(k-1) \\ y(k-2) \end{bmatrix} + \begin{bmatrix} g_1 & 0 \\ g_2 & g_1 \\ g_3 & g_2 \end{bmatrix} \begin{bmatrix} \Delta u(k) \\ \Delta u(k+1) \end{bmatrix} + \begin{bmatrix} g_2 & g_3 & p_{34} \\ g_3 & g_4 & p_{44} \\ g_4 & g_5 & p_{54} \end{bmatrix} \begin{bmatrix} \Delta u(k-1) \\ \Delta u(k-2) \\ \Delta u(k-3) \end{bmatrix} \tag{42}$$

Recursive computation of the matrices is analogical to the recursive computation described in the previous section.

The predictor can be also modified for arbitrary number of steps of time delay

$$\begin{bmatrix} \hat{y}(k+1+d) \\ \hat{y}(k+2+d) \\ \hat{y}(k+3+d) \end{bmatrix} = \begin{bmatrix} p_{(1+d)1} & p_{(1+d)2} & p_{(1+d)3} \\ p_{(2+d)1} & p_{(2+d)2} & p_{(2+d)3} \\ p_{(3+d)1} & p_{(3+d)2} & p_{(3+d)3} \end{bmatrix} \begin{bmatrix} y(k) \\ y(k-1) \\ y(k-2) \end{bmatrix} + \begin{bmatrix} g_1 & 0 \\ g_2 & g_1 \\ g_3 & g_2 \end{bmatrix} \begin{bmatrix} \Delta u(k) \\ \Delta u(k+1) \end{bmatrix} + \begin{bmatrix} g_{1+d-1} & g_{2+d-1} & p_{(1+d)4} \\ g_{2+d-1} & g_{3+d-1} & p_{(2+d)4} \\ g_{3+d-1} & g_{4+d-1} & p_{(3+d)4} \end{bmatrix} \begin{bmatrix} \Delta u(k-1) \\ \Delta u(k-2) \\ \Delta u(k-3) \end{bmatrix} \tag{43}$$

5 Simulation Examples

For simulation examples were chosen the systems introduced in the sections 2.1 and 2.2. Control responses are in the Figs. 7-10.

The tuning parameters that are lengths of the prediction and control horizons and the weighting coefficient λ were tuned experimentally. There is a lack of clear theory relating to the closed loop behavior to design parameters. The length of the prediction horizon, which should cover the important part of the step response, was in both cases set to N = 40. The length of the control horizon was also set to N_u = 40. The coefficient λ was taken as equal to 0,5.

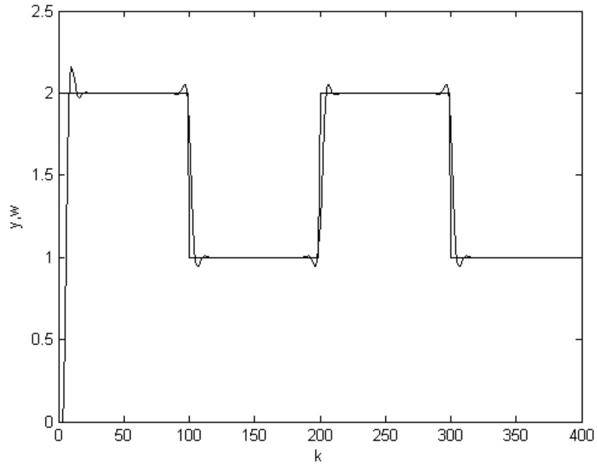

Fig. 7. Control of the model (16)

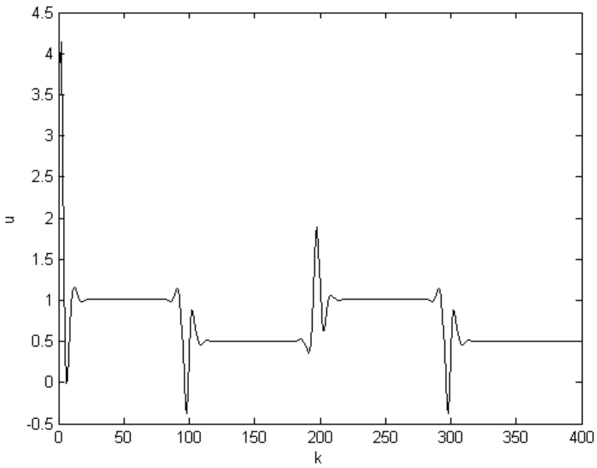

Fig. 8. Control of the model (16) – manipulated variable

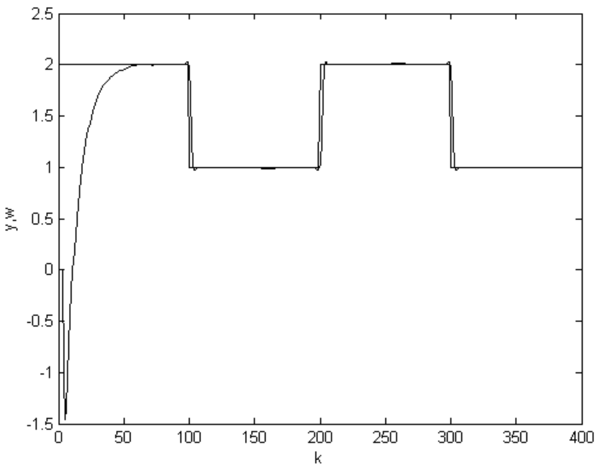

Fig. 9. Control of the model (18)

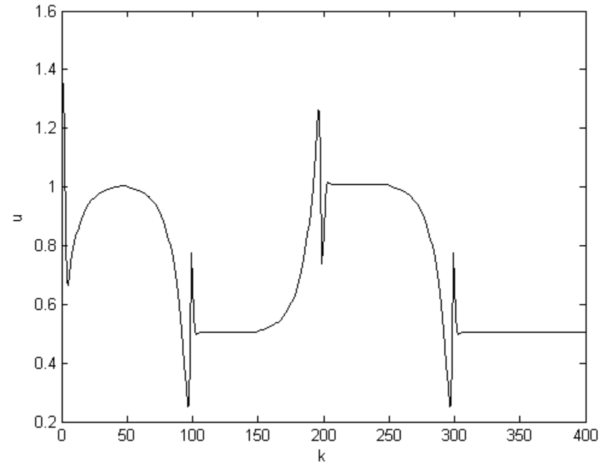

Fig. 10. Control of the model (18) – manipulated variable

Asymptotic tracking of the reference signal was achieved in all cases. The control of non-minimum phase system was rather sensitive to tuning parameters. Experimental tuning of the controller was more complicated in this case.

6 Experimental Example

The use of the predictive control algorithm is also demonstrated on a control of laboratory heat exchanger in simulation conditions. The laboratory heat exchanger [16] 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.

6.1 Laboratory Heat Exchanger Description

A scheme of the laboratory heat exchanger is depicted in Fig. 11 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 output temperatures of the cooler are measured again by platinum thermometers as T_2, resp. 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 was used.

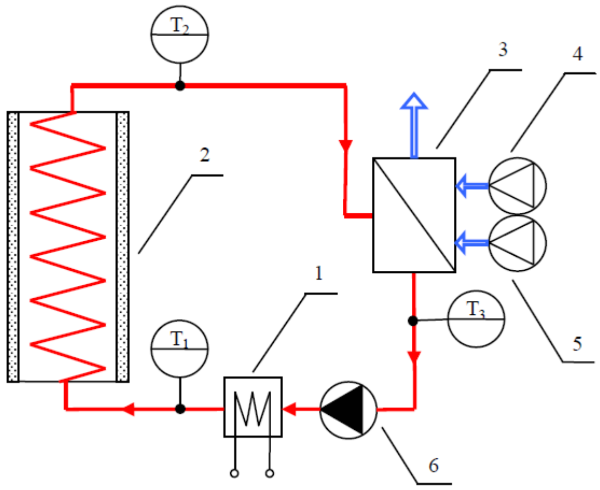

Fig. 11. Scheme of laboratory heat exchanger

6.2 Identification of Laboratory Heat Exchanger

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 [deg] of the cooler) data. As the input signal was again used the Random Gaussian Signal. Following discrete transfer function for sampling period $T_0 = 100$ s was identified

$$G(z^{-1}) = \frac{0.1494z^{-1} + 0.028z^{-2}}{1 - 0.6376z^{-1} - 0.1407z^{-2}} z^{-2} \tag{44}$$

Control responses are in Figs. 12-13.

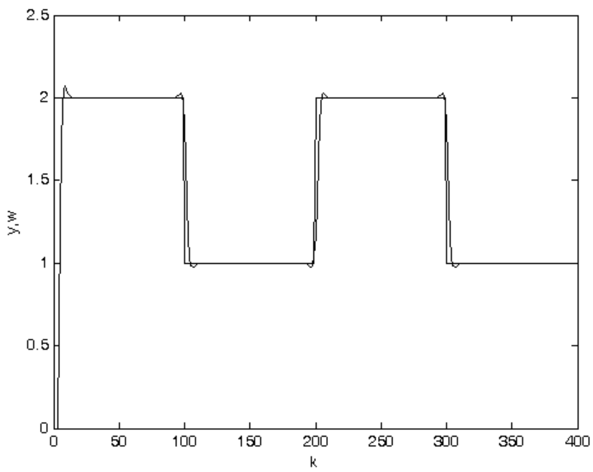

Fig. 12. Control of the model of the laboratory heat exchanger

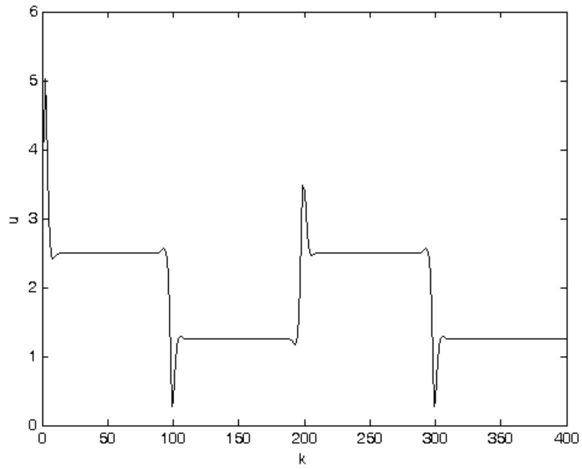

Fig. 13. Control of the model of the laboratory heat exchanger-manipulated variable

7 Conclusion

The algorithm for control of the higher-order processes based on model predictive control was designed. The higher-order process was approximated by the second-order model with time delay. The predictive controller is based on the recursive computation of predictions by direct use of the CARIMA model. The computation of predictions was extended for the time-delay system. The control of two modifications of the higher-order processes (stable and non-minimum phase) were verified by simulation. The laboratory heat exchanger system was identified by an experimental on-line method and its discrete model was also used for verification of the proposed predictive controller. The simulation verification provided good control results. The simulation experiments confirmed that predictive approach is able to cope with the given control problem.

Acknowledgements:

This article was created with support of Operational Programme Research and Development for Innovations co-funded by the European Regional Development Fund (ERDF), national budget of Czech Republic within the framework of the Centre of Polymer Systems project (reg. number: CZ.1.05/2.1.00/03.0111).

References:

[1] T. Hashimoto, T. Amemiya, Stabilization of Linear Time-Varying Uncertain Delay Systems with Double Triangular Configuration, *WSEAS Transactions on Systems and Control*, Vol. 4, Issue 9, September 2009, pp. 465-475, ISSN:1991-8763.

[2] Bahador Makki, Babarak Makki, Control Design for Uncertain Singularly Perturbed Systems with Discrete Time-Delay, *WSEAS Transactions on Systems and Control*, Vol. 6, Issue 12, December 2011, pp. 456-465, ISSN:1991-8763.

[3] F. Neri, Agent Based Modeling under Partial and Full Knowledge Learning Settings to Simulate Financial Markets, AI Communications, IOS Press, 2012.

[4] E. F. Camacho, C. Bordons, *Model Predictive Control,* Springer-Verlag, London, 2004.

[5] J. Mikleš & M. Fikar, *Process Modelling, Optimisation and Control.* (Berlin: Springer-Verlag, 2008).

[6] A. M. Yousef, Model Predictive Control Approach Based Load Frequency Controller, *WSEAS Transactions on Systems and Control*, Vol. 6, Issue 7, July 2011, pp. 265-275, ISSN:1991-8763.

[7] O. J. Smith, Closed control of loops. *Chem. Eng. Progress*, vol. 53, 1957, pp. 217-219.

[8] R. R. Bitmead, M. Gevers, V. Hertz, *Adaptive Optimal Control. The Thinking Man's GPC,* Prentice Hall, Englewood Cliffs, New Jersey, 1990.

[9] Jesus, S. Isabel, Machado, J.A. Tenreiro, Fractional Control of Heat Diffusion Systems, Nonlinear Dynamics, Springer, Vol. 54, Issue 3, pp. 263-282, 2008.

[10] J. A. Rossiter, *Model Based Predictive Control: a Practical Approach* (CRC Press, 2003).

[11] M. Morari, J. H. Lee, Model predictive control: past, present and future. *Computers and Chemical Engineering,* 23, 1999, 667-682.

[12] S. J. Quin & T. A. Bandgwell, An overview of nonlinear model predictive control applications. *Nonlinear Model Predictive Control* (F. Allgöwer & A. Zheng, Ed.), (Basel – Boston – Berlin: Birkhäuser Verlag, 2000), 369-392.

[13] S. J. Quin & T. A. Bandgwell, A survey of industrial model predictive control technology. *Control Engineering Practice*, *11*(7), 2003, 733-764.

[14] W. H. Kwon, H. Choj, D.G. Byun, S. Noh, Recursive solution of generalized predictive control and its equivalence to receding horizon tracking control. *Automatica*, 28(6), 1992, 1235–1238.

[15] M. Kubalčík, V. Bobál, Techniques for Predictor Design in Multivariable Predictive Control, *WSEAS Transactions on Systems and Control*, Vol. 6, Issue 9, September 2011, pp.349-360, ISSN:1991-8763.

[16] L. Pekař, R. Prokop, P. Dostálek, Circuit heating Plant Model with Internal Delays, *WSEAS Transactions on Systems*, Vol. 8, Issue 9, September 2009, pp. 1093-1104, ISSN:1109-2777.