Predictive Control Dead-Time Processes

MAREK KUBALČÍK, VLADIMÍR BOBÁL
Tomas Bata University in Zlín
Department of Process Control
Nad Stráněmi 4511, 76005, Zlín
CZECH REPUBLIC
kubalcik@fai.utb.cz, bobal@fai.utb.cz, http://web.utb.cz/fai

Abstract: - One of the possible approaches to control of dead-time processes is application of predictive control methods. In technical practice often occur higher order processes when a design of an optimal controller leads to complicated control algorithms. One of the possibilities of control of such processes is their approximation by lower-order model with dead-time (time-delay). The first part of the paper deals with a design of an algorithm for predictive control of high-order processes which are approximated by a second-order model of the process with time-delay. The second part of the paper deals with a design of an analogical algorithm for predictive control of multivariable processes with time-delay. The predictive controllers are based on the recursive computation of predictions which was extended for the time-delay system. The designed control algorithms were verified by simulation.

Key-Words: - predictive control, time-delay systems, transfer function models, matrix fraction models

1 Introduction

Some technological processes particularly in chemical industry are characterized by time-delays. Time-delays are mainly caused by the time required to transport mass, energy or information, but they can also be caused by processing time or accumulation. Time-delay may be defined as the time interval between the start of an event at one point in a system and its resulting action at another point in the system. One older classification of techniques for the compensation of time-delayed processes is introduced in [1],[2] and newer overview of recent advances and open problems it is possible to find in [3]. Processes with time-delay in general are difficult to control using standard feedback controllers. One of the possible approaches to control processes with time delay is model predictive control (MPC) [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 in [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.

When using most of other approaches, the control actions are based on past errors. MPC uses also future values of the reference signals. It is essentially based on discrete or sampled models of processes. Computation of appropriate control algorithms is then realized especially in the discrete domain. The basic idea of the generalized predictive control [8], [9] is to use a model of a controlled process to predict a number of future outputs of the process. A trajectory of future manipulated variables is given by solving an optimization problem incorporating a suitable cost function and constraints. Only the first element of the obtained control sequence is applied. The whole procedure is repeated in following sampling period. This principle is known as the receding horizon strategy.

Some technological processes in industry are characterized by high-order dynamic behaviour or large time constants which increase the difficulty of controlling it. However using the approximation of a higher-order process by a lower-order model with time-delay provides simplification of the control algorithms. The paper then introduces a design and verification of an algorithm for predictive control of second order linear systems with two steps time delay. A number of higher order industrial processes can be approximated by this model.

Typical technological processes require the simultaneous control of several variables related to one system. Each input may influence all system outputs. The design of a controller for such a system must be quite sophisticated if the system is to be controlled adequately. One of the most effective approaches to control of multivariable systems is model predictive control. An advantage of model predictive control is that multivariable systems can be handled in a straightforward manner. In technical practice also often occur multivariable processes with time delay. Typical examples of such processes are e.g. liquid storing tanks, distillation columns or some types of chemical reactors. The paper then deals also with a design of an analogical algorithm for predictive control of multivariable processes with time-delay. Both for control of the single input-single output and multivariable systems was applied the same approach. The predictive controllers are based on input-output models. In case of the SISO control it is a transfer function model and in case of the MIMO control the model is considered in the form of the matrix fraction. The models are used for a recursive computation of predictions which was extended for the time-delay systems. In case of the input-output model it is not necessary to examine observability. Feasibility is ensured by a suitable setting of constraints. The proposed algorithms were verified by simulation.

2 Model of the Controlled System

2.1 Model of SISO System

A model of the second order which is widely used in practice and has proved to be effective for control of a range of various processes was applied. The model without a time-delay is described by the transfer function

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

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

Model predictive control has an ability to deal with control difficulties such as nonlinearity, constrained variables, time-delay and also control of unstable systems. A system described by transfer function (1) may be then also unstable i.e. with roots of the denominator outside the Unite Circle. The proposed predictive controller then ensures BIBO (Bounded Input Bounded Output) stability of the whole closed loop system despite the fact that the controlled system is unstable. The model can be also written in the form

$$A\left(z^{-1}\right)y(k)=B\left(z^{-1}\right)u(k) \tag{3}$$

A widely used model in general model predictive control is the CARIMA model which we can obtain from the nominal model (3) by adding a disturbance model

$$A\left(z^{-1}\right)y(k)=B\left(z^{-1}\right)u(k)+\frac{C\left(z^{-1}\right)}{\Delta}n(k) \tag{4}$$

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

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

$$\Delta A\left(z^{-1}\right)y(k)=B\left(z^{-1}\right)\Delta u(k-1)+n(k) \tag{5}$$

The nominal model with time-delay is considered as

$$G\left(z^{-1}\right)=\frac{B\left(z^{-1}\right)}{A\left(z^{-1}\right)}z^{-d}=\frac{b_1 z^{-1}+b_2 z^{-2}}{1+a_1 z^{-1}+a_2 z^{-2}}z^{-d} \tag{6}$$

The CARIMA model for time-delay system takes the form

$$\Delta A\left(z^{-1}\right)y(k)=z^{-d}B\left(z^{-1}\right)\Delta u(k-1)+n(k) \tag{7}$$

where d is the dead time. In our case d is equal to 2.

2.2 Model of MIMO System

Let us consider a two input – two output system. The two – input/two – output (TITO) processes are very often encountered multivariable processes in practice and many processes with inputs/outputs beyond two can be treated as several TITO subsystems [10].

A general transfer matrix of a two-input–two-output system with significant cross-coupling between the control loops is expressed as:

$$\boldsymbol{G}(z)=\begin{bmatrix} G_{11}(z) & G_{12}(z) \\ G_{21}(z) & G_{22}(z) \end{bmatrix} \tag{8}$$

$$\boldsymbol{Y}(z)=\boldsymbol{G}(z)\boldsymbol{U}(z) \tag{9}$$

where $\boldsymbol{U}(z)$ and $\boldsymbol{Y}(z)$ are vectors of the manipulated variables and the controlled variables, respectively.

$$\boldsymbol{U}(z)=\left[u_1(z), u_2(z)\right]^T \; \boldsymbol{Y}(z)=\left[y_1(z), y_2(z)\right]^T \tag{10}$$

It may be assumed that the transfer matrix can be transcribed to the following form of the matrix fraction:

$$\boldsymbol{G}(z)=\boldsymbol{A}^{-1}\left(z^{-1}\right)\boldsymbol{B}\left(z^{-1}\right)=\boldsymbol{B}_1\left(z^{-1}\right)\boldsymbol{A}_1^{-1}\left(z^{-1}\right) \tag{11}$$

where the polynomial matrices $\boldsymbol{A}\in R_{22}\left[z^{-1}\right], \boldsymbol{B}\in R_{22}\left[z^{-1}\right]$ are the left coprime factorizations of matrix $\boldsymbol{G}(z)$ and the matrices $\boldsymbol{A}_1\in R_{22}\left[z^{-1}\right], \boldsymbol{B}_1\in R_{22}\left[z^{-1}\right]$ are the right coprime factorizations of $\boldsymbol{G}(z)$. The model can be also written in the form

$$\boldsymbol{A}\left(z^{-1}\right)\boldsymbol{Y}(z)=\boldsymbol{B}\left(z^{-1}\right)\boldsymbol{U}(z) \tag{12}$$

As an example a model with polynomials of second degree was chosen. This model proved to be effective for control of several real TITO processes [11], where controllers based on a model with polynomials of the first degree failed. The model has sixteen parameters. The matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ are defined as follows

$$\boldsymbol{A}\left(z^{-1}\right)=\begin{bmatrix} 1+a_1z^{-1}+a_2z^{-2} & a_3z^{-1}+a_4z^{-2} \\ a_5z^{-1}+a_6z^{-2} & 1+a_7z^{-1}+a_8z^{-2} \end{bmatrix} \quad (13)$$

$$\boldsymbol{B}\left(z^{-1}\right)=\begin{bmatrix} b_1z^{-1}+b_2z^{-2} & b_3z^{-1}+b_4z^{-2} \\ b_5z^{-1}+b_6z^{-2} & b_7z^{-1}+b_8z^{-2} \end{bmatrix} \quad (14)$$

the CARIMA model in the MIMO case is as follows

$$\boldsymbol{A}\left(z^{-1}\right)\boldsymbol{y}(k)=\boldsymbol{B}\left(z^{-1}\right)\boldsymbol{u}(k)+\boldsymbol{C}\left(z^{-1}\right)\boldsymbol{\Delta}^{-1}\left(z^{-1}\right)\boldsymbol{n}(k) \quad (15)$$

where

$$\boldsymbol{\Delta}\left(z^{-1}\right)=\begin{bmatrix} 1-z^{-1} & 0 \\ 0 & 1-z^{-1} \end{bmatrix} \quad (16)$$

$\boldsymbol{C}$ is a colouring polynomial matrix. For purpose of simplification it was supposed to be equal to the identity matrix [4].

The nominal model with d steps of time-delay is considered as

$$\boldsymbol{G}(z)=\boldsymbol{A}^{-1}\left(z^{-1}\right)\boldsymbol{B}\left(z^{-1}\right)z^{-d}=\boldsymbol{B}_1\left(z^{-1}\right)\boldsymbol{A}_1^{-1}\left(z^{-1}\right)z^{-d} \quad (17)$$

For the purpose of simplification it was considered an equal time-delay in all particular transfer functions of the transfer matrix. The CARIMA model for time-delay system then takes the form

$$\boldsymbol{A}\left(z^{-1}\right)\boldsymbol{y}(k)=z^{-d}\boldsymbol{B}\left(z^{-1}\right)\boldsymbol{u}(k)+\boldsymbol{C}\left(z^{-1}\right)\boldsymbol{\Delta}^{-1}\left(z^{-1}\right)\boldsymbol{n}(k) \quad (18)$$

3 Implementation of predictive controller

The basic idea of MPC is to use a model of a controlled process to predict N future outputs of the process. A trajectory of future manipulated variables is given by solving an optimization problem incorporating a suitable cost function and constraints. Only the first element of the obtained control sequence is applied. The whole procedure is repeated in following sampling period. This principle is known as the receding horizon strategy. The computation of a control law of MPC is based on minimization of the following criterion

$$J(k)=\sum_{j=N_1}^{N}\boldsymbol{e}(k+j)^2+\lambda\sum_{j=1}^{N_u}\Delta\boldsymbol{u}(k+j)^2 \quad (19)$$

where $\boldsymbol{e}(k+j)$ is a vector of predicted control errors, $\Delta\boldsymbol{u}(k+j)$ is a vector of future increments of the manipulated variable (for the system with two inputs and two outputs each vector has two elements), N is a length of the prediction horizon, N_u is a length of the control horizon and λ is a weighting factor of control increments.

A predictor in a vector form is given by

$$\hat{\boldsymbol{y}}=\boldsymbol{G}\Delta\boldsymbol{u}+\boldsymbol{y}_0 \quad (20)$$

where $\hat{\boldsymbol{y}}$ is a vector of system predictions along the horizon of the length N, $\Delta\boldsymbol{u}$ is a vector of control increments, $\boldsymbol{y}_0$ is the free response vector. $\boldsymbol{G}$ is a matrix of the dynamics. It contains values of the step sequence. In SISO case it is given as

$$\boldsymbol{G}=\begin{bmatrix} g_1 & 0 & 0 & \cdots & 0 \\ g_2 & g_1 & 0 & \cdots & 0 \\ g_3 & g_2 & g_1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ g_N & g_{N-1} & g_{N-2} & \cdots & g_{N-N_u+1} \end{bmatrix} \quad (21)$$

In TITO case the matrix $\boldsymbol{G}$ takes the following form

$$\boldsymbol{G}=\begin{bmatrix} \boldsymbol{G}_0 & 0 & \cdots & \cdots & 0 \\ \boldsymbol{G}_1 & \boldsymbol{G}_0 & 0 & \cdots & 0 \\ \vdots & & \ddots & \ddots & \vdots \\ \vdots & & & \boldsymbol{G}_0 & 0 \\ \boldsymbol{G}_{N-1} & \cdots & \cdots & \cdots & \boldsymbol{G}_0 \end{bmatrix} \quad (22)$$

where sub-matrices $\boldsymbol{G}_i$ have dimension 2x2 and contain values of the step sequence.

The criterion (12) can be written in a general vector form

$$J=\left(\hat{\boldsymbol{y}}-\boldsymbol{w}\right)^T\left(\hat{\boldsymbol{y}}-\boldsymbol{w}\right)+\lambda\Delta\boldsymbol{u}^T\Delta\boldsymbol{u} \quad (23)$$

where $\boldsymbol{w}$ is a vector of the reference trajectory. The criterion can be modified using the expression (19) to

$$J=2\boldsymbol{g}^T\Delta\boldsymbol{u}+\Delta\boldsymbol{u}^T\boldsymbol{H}\Delta\boldsymbol{u} \quad (24)$$

where the gradient $\boldsymbol{g}$ and the Hess matrix $\boldsymbol{H}$ are defined by following expressions

$$\boldsymbol{g}^T=\boldsymbol{G}^T\left(\boldsymbol{y}_0-\boldsymbol{w}\right) \quad (25)$$

$$\boldsymbol{H}=\boldsymbol{G}^T\boldsymbol{G}+\lambda\boldsymbol{I} \quad (26)$$

Handling of constraints is one of main advantages of predictive control. General formulation of predictive control with constraints is then as follows

$$\min_{\Delta \boldsymbol{u}} 2\boldsymbol{g}^T \Delta \boldsymbol{u} + \Delta \boldsymbol{u}^T \boldsymbol{H} \Delta \boldsymbol{u} \tag{27}$$

owing to

$$\boldsymbol{A}\Delta \boldsymbol{u} \le \boldsymbol{b} \tag{28}$$

The inequality (28) expresses the constraints in a compact form.

4 Computation of predictions – SISO system

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 *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 [12]. In [13] 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.

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 Computation of predictions without time-delay

The difference equation of the CARIMA model without the unknown term can be expressed as:

$$\begin{aligned} y(k) &= (1-a_1)y(k-1) + (a_1-a_2)y(k-2) + \\ &+ a_2 y(k-3) + b_1\Delta u(k-1) + b_2\Delta u(k-2) \end{aligned} \tag{29}$$

It was necessary to directly compute three steps-ahead predictions in a straightforward way by establishing of previous predictions to later predictions. The model order defines that computation of one step-ahead prediction is based on the three past values of the system output.

$$\begin{aligned} \hat{y}(k+1) &= (1-a_1)y(k) + (a_1-a_2)y(k-1) + a_2 y(k-2) + \\ &+ b_1\Delta u(k) + b_2\Delta u(k-1) \\ \hat{y}(k+2) &= (1-a_1)\hat{y}(k+1) + (a_1-a_2)y(k) + a_2 y(k-1) + \\ &+ b_1\Delta u(k+1) + b_2\Delta u(k) \\ \hat{y}(k+3) &= (1-a_1)\hat{y}(k+2) + (a_1-a_2)\hat{y}(k+1) + a_2 y(k) + \\ &+ b_1\Delta u(k+2) + b_2\Delta u(k+1) \end{aligned} \tag{30}$$

The predictions after modification can be written in a matrix form

$$\begin{bmatrix} \hat{y}(k+1) \\ \hat{y}(k+2) \\ \hat{y}(k+3) \end{bmatrix} = \begin{bmatrix} g_1 & 0 \\ g_2 & g_1 \\ g_3 & g_2 \end{bmatrix} \begin{bmatrix} \Delta u(k) \\ \Delta u(k+1) \end{bmatrix} + \begin{bmatrix} p_1 \\ p_2 \\ p_3 \end{bmatrix} \Delta u(k-1) + \begin{bmatrix} q_{11} & q_{12} & q_{13} \\ q_{21} & q_{22} & q_{23} \\ q_{31} & q_{32} & q_{33} \end{bmatrix} \begin{bmatrix} y(k) \\ y(k-1) \\ y(k-2) \end{bmatrix} \tag{31}$$

$$\begin{bmatrix} \hat{y}(k+1) \\ \hat{y}(k+2) \\ \hat{y}(k+3) \end{bmatrix} = \boldsymbol{G} \begin{bmatrix} \Delta u(k) \\ \Delta u(k+1) \end{bmatrix} + \boldsymbol{P}\Delta u(k-1) + \boldsymbol{Q} \begin{bmatrix} y(k) \\ y(k-1) \\ y(k-2) \end{bmatrix} \tag{32}$$

$$\begin{aligned} &\boldsymbol{y}(k+1) = \boldsymbol{G}\Delta \boldsymbol{u}(k+j-1) + \boldsymbol{P}\Delta \boldsymbol{u}(k-1) + \boldsymbol{Q}\boldsymbol{y}(k+j-1) \\ &j \le N \end{aligned} \tag{33}$$

where

$$\boldsymbol{G} \begin{bmatrix} \Delta u(k) \\ \Delta u(k+1) \end{bmatrix} = \begin{bmatrix} b_1 & 0 \\ b_1(1-a_1) + b_2 & b_1 \\ (a_1-a_2)b_1 + (1-a_1)^2 b_1 + (1-a_1)b_2 & (1-a_1)b_1 + b_2 \end{bmatrix} \begin{bmatrix} \Delta u(k) \\ \Delta u(k+1) \end{bmatrix} \tag{34}$$

$$\boldsymbol{P}\Delta u(k-1) = \begin{bmatrix} b_2 \\ (1-a_1)b_2 \\ (1-a_1)^2 b_2 + (a_1-a_2)b_2 \end{bmatrix} \Delta u(k-1) \tag{35}$$

$$\boldsymbol{Q} \begin{bmatrix} y(k) \\ y(k-1) \\ y(k-2) \end{bmatrix} = \begin{bmatrix} (1-a_1) & (a_1-a_2) & a_2 \\ (1-a_1)^2 + (a_1-a_2) & (1-a_1)(a_1-a_2) + a_2 & a_2(1-a_1) \\ (1-a_1)^3 + 2(1-a_1)(a_1-a_2) + a_2 & (1-a_1)^2(a_1-a_2) + a_2(1-a_1) + (a_1-a_2)^2 & a_2(1-a_1)^2 + (a_1-a_2)a_2 \end{bmatrix} \begin{bmatrix} y(k) \\ y(k-1) \\ y(k-2) \end{bmatrix} \tag{36}$$

The coefficients of the matrices $\boldsymbol{G}$, $\boldsymbol{P}$ and $\boldsymbol{Q}$ for further predictions are computed recursively. Based on the three previous predictions it is repeatedly computed the next row of the matrices $\boldsymbol{P}$ and $\boldsymbol{Q}$ in the following way:

$$p_4 = (1-a_1)p_3 + (a_1 - a_2)p_2 + a_2 p_1 \tag{37}$$

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

The recursion of the matrix $\boldsymbol{G}$ is similar. The next element of the first column is repeatedly computed and the remaining columns are shifted. 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 a 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 Computation of predictions with time-delay

In order to compute the control action it is necessary to determine the predictions from d+1 to d+N.

The predictor (31) is then modified to

$$\begin{bmatrix} \hat{y}(k+1) \\ \hat{y}(k+2) \\ \hat{y}(k+3) \end{bmatrix} = \begin{bmatrix} g_1 & 0 \\ g_2 & g_1 \\ g_3 & g_2 \end{bmatrix} \begin{bmatrix} \Delta u(k) \\ \Delta u(k+1) \end{bmatrix} + \\ + \begin{bmatrix} g_d & g_{d+1} & \cdots & g_{d+d-1} & p_{(1+d)4} \\ g_{d+1} & g_{d+2} & \cdots & g_{d+d} & p_{(2+d)4} \\ g_{d+2} & g_{d+3} & \cdots & g_{d+d+1} & p_{(3+d)4} \end{bmatrix} \begin{bmatrix} \Delta u(k-1) \\ \Delta u(k-2) \\ \Delta u(k-3) \\ \vdots \\ \Delta u(k-d-1) \end{bmatrix} + \\ + \begin{bmatrix} q_{(1+d)1} & q_{(1+d)2} & q_{(1+d)3} \\ q_{(2+d)1} & q_{(2+d)2} & q_{(2+d)3} \\ q_{(3+d)1} & q_{(3+d)2} & q_{(3+d)3} \end{bmatrix} \begin{bmatrix} y(k) \\ y(k-1) \\ y(k-2) \end{bmatrix} \tag{40}$$

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

The predictor modified for two steps of time – delay is then given as follows

$$\begin{bmatrix} \hat{y}(k+1) \\ \hat{y}(k+2) \\ \hat{y}(k+3) \end{bmatrix} = \begin{bmatrix} g_1 & 0 \\ g_2 & g_1 \\ g_3 & g_2 \end{bmatrix} \begin{bmatrix} \Delta u(k) \\ \Delta u(k+1) \end{bmatrix} + \\ + \begin{bmatrix} g_2 & g_3 & p_3 \\ g_3 & g_4 & p_4 \\ g_4 & g_5 & p_5 \end{bmatrix} \begin{bmatrix} \Delta u(k-1) \\ \Delta u(k-2) \\ \Delta u(k-3) \end{bmatrix} + \begin{bmatrix} q_{31} & q_{32} & q_{33} \\ q_{41} & q_{42} & q_{43} \\ q_{51} & q_{52} & q_{53} \end{bmatrix} \begin{bmatrix} y(k) \\ y(k-1) \\ y(k-2) \end{bmatrix} \tag{41}$$

5 Computation of predictions – TITO system

5.1 Computation of predictions without time delay

The difference equations of the CARIMA model without the unknown term are as follows

$$\begin{aligned} y_1(k+1) &= (1-a_1)y_1(k) + (a_1 - a_2)y_1(k-1) + a_2 y_1(k-2) - \\ &- a_3 y_2(k) + (a_3 - a_4)y_2(k-1) + a_4 y_2(k-2) + \\ &+ b_1 \Delta u_1(k) + b_2 \Delta u_1(k-1) + b_3 \Delta u_2(k) + b_4 \Delta u_2(k-1) \\ y_2(k+1) &= (1-a_7)y_2(k) + (a_7 - a_8)y_2(k-1) + a_8 y_2(k-2) - \\ &- a_5 y_1(k) + (a_5 - a_6)y_1(k-1) + a_6 y_1(k-2) + \\ &+ b_5 \Delta u_1(k) + b_6 \Delta u_1(k-1) + b_7 \Delta u_2(k) + b_8 \Delta u_2(k-1) \end{aligned} \tag{42}$$

These equations can be written into a matrix form

$$\begin{aligned} \boldsymbol{y}(k+1) &= \boldsymbol{A}_1 \boldsymbol{y}(k) + \boldsymbol{A}_2 \boldsymbol{y}(k-1) + \boldsymbol{A}_3 \boldsymbol{y}(k-2) + \\ &+ \boldsymbol{B}_1 \Delta \boldsymbol{u}(k) + \boldsymbol{B}_2 \Delta \boldsymbol{u}(k-1) \end{aligned} \tag{43}$$

where

$$\boldsymbol{A}_1 = \begin{bmatrix} 1-a_1 & -a_3 \\ -a_5 & 1-a_7 \end{bmatrix} \quad \boldsymbol{A}_2 = \begin{bmatrix} a_1 - a_2 & a_3 - a_4 \\ a_5 - a_6 & a_7 - a_8 \end{bmatrix}$$

$$\boldsymbol{A}_3 = \begin{bmatrix} a_2 & a_4 \\ a_6 & a_8 \end{bmatrix} \tag{44}$$

$$\boldsymbol{B}_1 = \begin{bmatrix} b_1 & b_3 \\ b_5 & b_7 \end{bmatrix} \quad \boldsymbol{B}_2 = \begin{bmatrix} b_2 & b_4 \\ b_6 & b_8 \end{bmatrix} \tag{45}$$

The computation of three steps-ahead predictions can be expressed as

$$\begin{aligned} \hat{\boldsymbol{y}}(k+1) &= \boldsymbol{A}_1 \boldsymbol{y}(k) + \boldsymbol{A}_2 \boldsymbol{y}(k-1) + \boldsymbol{A}_3 \boldsymbol{y}(k-2) + \\ &+ \boldsymbol{B}_1 \Delta \boldsymbol{u}(k) + \boldsymbol{B}_2 \Delta \boldsymbol{u}(k-1) \\ \hat{\boldsymbol{y}}(k+2) &= \boldsymbol{A}_1 \boldsymbol{y}(k+1) + \boldsymbol{A}_2 \boldsymbol{y}(k) + \boldsymbol{A}_3 \boldsymbol{y}(k-1) + \\ &+ \boldsymbol{B}_1 \Delta \boldsymbol{u}(k+1) + \boldsymbol{B}_2 \Delta \boldsymbol{u}(k) \\ \hat{\boldsymbol{y}}(k+3) &= \boldsymbol{A}_1 \boldsymbol{y}(k+2) + \boldsymbol{A}_2 \boldsymbol{y}(k+1) + \boldsymbol{A}_3 \boldsymbol{y}(k) + \\ &+ \boldsymbol{B}_1 \Delta \boldsymbol{u}(k+2) + \boldsymbol{B}_2 \Delta \boldsymbol{u}(k+1) \end{aligned} \tag{46}$$

The equations (46) can be written in a compact form using (20) as follows

$$\begin{bmatrix} \hat{\boldsymbol{y}}(k+1) \\ \hat{\boldsymbol{y}}(k+2) \\ \hat{\boldsymbol{y}}(k+3) \end{bmatrix} = \boldsymbol{G} \Delta \boldsymbol{u} + \boldsymbol{y}_0 \tag{47}$$

It is possible to divide computation of the predictions to recursion of the free response and recursion of the matrix of the dynamics. The free response vector predictions can be expressed as:

$$\boldsymbol{y}_0=\left[\begin{array}{cc} p_{11} & p_{12} \\ p_{21} & q_{22} \\ \hline p_{31} & p_{32} \\ p_{41} & p_{42} \\ \hline p_{51} & p_{52} \\ p_{61} & p_{62} \end{array}\right]\begin{bmatrix}\Delta u_1(k-1)\\ \Delta u_2(k-1)\end{bmatrix}+\left[\begin{array}{cc|cc|cc} q_{11} & q_{12} & q_{13} & q_{14} & q_{15} & q_{16} \\ q_{21} & q_{22} & q_{23} & q_{24} & q_{25} & q_{26} \\ \hline q_{31} & q_{32} & q_{33} & q_{34} & q_{35} & q_{36} \\ q_{41} & q_{42} & q_{43} & q_{44} & q_{45} & q_{46} \\ \hline q_{51} & q_{52} & q_{53} & q_{54} & q_{55} & q_{56} \\ q_{61} & q_{62} & q_{63} & q_{64} & q_{65} & q_{66} \end{array}\right]\left[\begin{array}{c} y_1(k) \\ y_2(k) \\ \hline y_1(k-1) \\ y_2(k-1) \\ \hline y_1(k-2) \\ y_2(k-2) \end{array}\right]=$$

$$=\begin{bmatrix}\boldsymbol{P}_1\\ \boldsymbol{P}_2\\ \boldsymbol{P}_3\end{bmatrix}\Delta\boldsymbol{u}(k-1)+\begin{bmatrix}\boldsymbol{Q}_{11} & \boldsymbol{Q}_{12} & \boldsymbol{Q}_{13}\\ \boldsymbol{Q}_{21} & \boldsymbol{Q}_{22} & \boldsymbol{Q}_{23}\\ \boldsymbol{Q}_{31} & \boldsymbol{Q}_{32} & \boldsymbol{Q}_{33}\end{bmatrix}\begin{bmatrix}\boldsymbol{y}(k)\\ \boldsymbol{y}(k-1)\\ \boldsymbol{y}(k-2)\end{bmatrix}=\boldsymbol{P}\Delta\boldsymbol{u}(k-1)+\boldsymbol{Q}\begin{bmatrix}\boldsymbol{y}(k)\\ \boldsymbol{y}(k-1)\\ \boldsymbol{y}(k-2)\end{bmatrix} \quad (48)$$

The coefficients of the matrices $\boldsymbol{P}$ and $\boldsymbol{Q}$ for further predictions are computed recursively. Based on the three previous predictions it is repeatedly computed the next row of the matrices $\boldsymbol{P}$ and $\boldsymbol{Q}$ in the following way:

$$\boldsymbol{P}_4=\begin{bmatrix}p_{71} & p_{72}\\ p_{81} & p_{82}\end{bmatrix}=\boldsymbol{A}_1\boldsymbol{P}_{31}+\boldsymbol{A}_2\boldsymbol{P}_{21}+\boldsymbol{A}_3\boldsymbol{P}_{11} \quad (49)$$

$$\boldsymbol{Q}_{41}=\begin{bmatrix}q_{71} & q_{72}\\ q_{81} & q_{82}\end{bmatrix}=\boldsymbol{A}_1\boldsymbol{Q}_{31}+\boldsymbol{A}_2\boldsymbol{Q}_{21}+\boldsymbol{A}_3\boldsymbol{Q}_{11} \quad (50)$$

$$\boldsymbol{Q}_{42}=\begin{bmatrix}q_{73} & q_{74}\\ q_{83} & q_{84}\end{bmatrix}=\boldsymbol{A}_1\boldsymbol{Q}_{32}+\boldsymbol{A}_2\boldsymbol{Q}_{22}+\boldsymbol{A}_3\boldsymbol{Q}_{12} \quad (51)$$

$$\boldsymbol{Q}_{43}=\begin{bmatrix}q_{75} & q_{76}\\ q_{85} & q_{86}\end{bmatrix}=\boldsymbol{A}_1\boldsymbol{Q}_{33}+\boldsymbol{A}_2\boldsymbol{Q}_{23}+\boldsymbol{A}_3\boldsymbol{Q}_{13} \quad (52)$$

The recursion of the matrix $\boldsymbol{G}$ is analogical. The computation is similar as it was introduced in section 4.1. It is apparent from equations (53) and (54).

$$\boldsymbol{G}\Delta\boldsymbol{u}=\left[\begin{array}{cc|cc} g_{11} & g_{12} & 0 & 0 \\ g_{21} & g_{22} & 0 & 0 \\ \hline g_{31} & g_{32} & g_{11} & g_{12} \\ g_{41} & g_{42} & g_{21} & g_{22} \\ \hline g_{51} & g_{52} & g_{31} & g_{32} \\ g_{61} & g_{62} & g_{41} & g_{42} \end{array}\right]\left[\begin{array}{c}\Delta u_1(k)\\ \Delta u_2(k)\\ \hline \Delta u_1(k+1)\\ \Delta u_2(k+1)\end{array}\right]= \quad (53)$$

$$=\begin{bmatrix}\boldsymbol{G}_1 & 0\\ \boldsymbol{G}_2 & \boldsymbol{G}_1\\ \boldsymbol{G}_3 & \boldsymbol{G}_2\end{bmatrix}\begin{bmatrix}\Delta\boldsymbol{u}(k)\\ \Delta\boldsymbol{u}(k+1)\end{bmatrix}$$

$$\boldsymbol{G}_4=\begin{bmatrix}g_{71} & g_{72}\\ g_{81} & g_{82}\end{bmatrix}=\boldsymbol{A}_1\boldsymbol{G}_3+\boldsymbol{A}_2\boldsymbol{G}_2+\boldsymbol{A}_3\boldsymbol{G}_1 \quad (54)$$

The predictions can be written in a compact matrix form

$$\begin{aligned}&\hat{\boldsymbol{y}}(k+j)=\boldsymbol{G}\Delta\boldsymbol{u}(k+j-1)+\boldsymbol{P}\Delta\boldsymbol{u}(k-1)+\boldsymbol{Q}\boldsymbol{y}(k-j+1)\\ &j\le N\end{aligned} \quad (55)$$

5.2 Computation of predictions with time-delay

The predictor modified for an arbitrary time –delay is given as follows.

$$\begin{bmatrix}\hat{\boldsymbol{y}}(k+1+d)\\ \hat{\boldsymbol{y}}(k+2+d)\\ \hat{\boldsymbol{y}}(k+3+d)\end{bmatrix}=\boldsymbol{G}\Delta\boldsymbol{u}+\boldsymbol{y}_0 \quad (56)$$

The computation of the free response is then modified to

$$\boldsymbol{y}_0=\left[\begin{array}{cc|cc|c|cc|cc} g_{(d+1)1} & g_{(d+1)2} & g_{(d+3)1} & g_{(d+3)2} & \cdots & g_{(d+d+1)1} & g_{(d+d+1)2} & p_{(d+3)1} & p_{(d+3)2} \\ g_{(d+2)1} & g_{(d+2)2} & g_{(d+4)1} & g_{(d+4)2} & \cdots & g_{(d+d+2)1} & g_{(d+d+2)1} & p_{(d+4)1} & p_{(d+4)2} \\ \hline g_{(d+3)1} & g_{(d+3)2} & g_{(d+5)1} & g_{(d+5)2} & \cdots & g_{(d+d+3)1} & g_{(d+d+3)1} & p_{(d+5)1} & p_{(d+5)2} \\ g_{(d+4)1} & g_{(d+4)2} & g_{(d+6)1} & g_{(d+6)2} & \cdots & g_{(d+d+4)1} & g_{(d+d+4)1} & p_{(d+6)1} & p_{(d+6)2} \\ \hline g_{(d+5)1} & g_{(d+5)2} & g_{(d+7)1} & g_{(d+7)2} & \cdots & g_{(d+d+5)1} & g_{(d+d+5)1} & p_{(d+7)1} & p_{(d+7)2} \\ g_{(d+6)1} & g_{(d+6)2} & g_{(d+8)1} & g_{(d+8)2} & \cdots & g_{(d+d+6)1} & g_{(d+d+6)1} & p_{(d+8)1} & p_{(d+8)2} \end{array}\right].$$

$$\cdot\left[\begin{array}{c}\Delta u_1(k-1)\\ \Delta u_2(k-1)\\ \hline \Delta u_1(k-2)\\ \Delta u_2(k-2)\\ \hline \vdots \\ \hline \Delta u_1(k-d-2)\\ \Delta u_2(k-d-2)\\ \hline \Delta u_1(k-d-1)\\ \Delta u_2(k-d-1)\end{array}\right]+\left[\begin{array}{cc|cc|cc} q_{(d+3)1} & q_{(d+3)2} & q_{(d+3)3} & q_{(d+3)4} & q_{(d+3)5} & q_{(d+3)6} \\ q_{(d+4)1} & q_{(d+4)2} & q_{(d+4)3} & q_{(d+4)4} & q_{(d+4)5} & q_{(d+4)6} \\ \hline q_{(d+5)1} & q_{(d+5)2} & q_{(d+5)3} & q_{(d+5)4} & q_{(d+5)5} & q_{(d+5)6} \\ q_{(d+6)1} & q_{(d+6)2} & q_{(d+6)3} & q_{(d+6)4} & q_{(d+6)5} & q_{(d+6)6} \\ \hline q_{(d+7)1} & q_{(d+7)2} & q_{(d+7)3} & q_{(d+7)4} & q_{(d+7)5} & q_{(d+7)6} \\ q_{(d+8)1} & q_{(d+8)2} & q_{(d+8)3} & q_{(d+8)4} & q_{(d+8)5} & q_{(d+8)6} \end{array}\right]\left[\begin{array}{c} y_1(k) \\ y_2(k) \\ \hline y_1(k-1) \\ y_2(k-1) \\ \hline y_1(k-2) \\ y_2(k-2) \end{array}\right]=$$

$$=\begin{bmatrix}\boldsymbol{G}_d & \boldsymbol{G}_{d+1} & \cdots & \boldsymbol{G}_{d+d-1} & \boldsymbol{P}_{d+1}\\ \boldsymbol{G}_{d+1} & \boldsymbol{G}_{d+2} & \cdots & \boldsymbol{G}_{d+d} & \boldsymbol{P}_{d+2}\\ \boldsymbol{G}_{d+2} & \boldsymbol{G}_{d+3} & \cdots & \boldsymbol{G}_{d+d+1} & \boldsymbol{P}_{d+3}\end{bmatrix}\begin{bmatrix}\Delta\boldsymbol{u}(k-1)\\ \Delta\boldsymbol{u}(k-2)\\ \vdots\\ \Delta\boldsymbol{u}(k-d-2)\\ \Delta\boldsymbol{u}(k-d-1)\end{bmatrix}+$$

$$+\begin{bmatrix}\boldsymbol{Q}_{(d+1)1} & \boldsymbol{Q}_{(d+1)2} & \boldsymbol{Q}_{(d+1)3}\\ \boldsymbol{Q}_{(d+2)1} & \boldsymbol{Q}_{(d+2)2} & \boldsymbol{Q}_{(d+1)3}\\ \boldsymbol{Q}_{(d+3)1} & \boldsymbol{Q}_{(d+3)2} & \boldsymbol{Q}_{(d+1)3}\end{bmatrix}\begin{bmatrix}\boldsymbol{y}(k)\\ \boldsymbol{y}(k-1)\\ \boldsymbol{y}(k-2)\end{bmatrix}=$$

$$=\boldsymbol{P}_2\begin{bmatrix}\Delta\boldsymbol{u}(k-1)\\ \Delta\boldsymbol{u}(k-2)\\ \vdots\\ \Delta\boldsymbol{u}(k-d-2)\\ \Delta\boldsymbol{u}(k-d-1)\end{bmatrix}+\boldsymbol{Q}_2\begin{bmatrix}\boldsymbol{y}(k)\\ \boldsymbol{y}(k-1)\\ \boldsymbol{y}(k-2)\end{bmatrix} \quad (57)$$

The computation of the forced response is again given by equation (53)

6 Simulation Verification

6.1 SISO control

As simulation examples were chosen a fifth order linear system described by following transfer function

$$G_A(s)=\frac{2}{(s+1)^5}=\frac{2}{s^5+5s^4+10s^3+10s^2+5s+1} \quad (58)$$

and a fifth-order linear system with non-minimum phase

$$G_B(s)=\frac{2(1-5s)}{s^5+5s^4+10s^3+10s^2+5s+1} \quad (59)$$

The systems were identified by model (6) using off-line LSM (least squares method) [14]. System (58) was approximated by

$$G_A\left(z^{-1}\right)=\frac{0{,}0424z^{-1}+0{,}0296z^{-2}}{1-1{,}6836z^{-1}+0{,}7199z^{-2}}\,z^{-2} \tag{60}$$

and system (59) was approximated by

$$G_B\left(z^{-1}\right)=\frac{-0{,}7723z^{-1}+0{,}8514z^{-2}}{1-1{,}6521z^{-1}+0{,}6920z^{-2}}\,z^{-2} \tag{61}$$

Both for sampling period $T_0 = 0.5\,s$. The step responses of models (58) and (59) together with discrete step responses of their approximations (60) and (61) are in the following figures

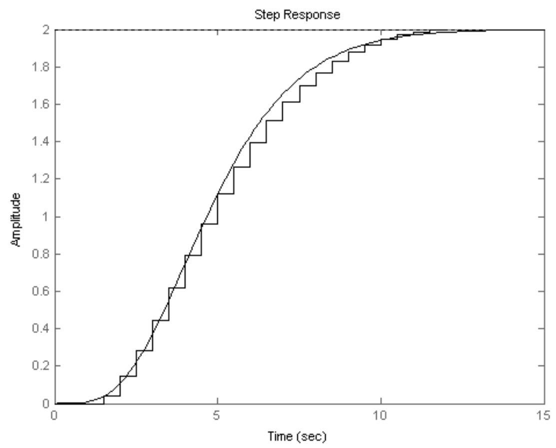

Fig. 1 Step responses of models (58) and (60)

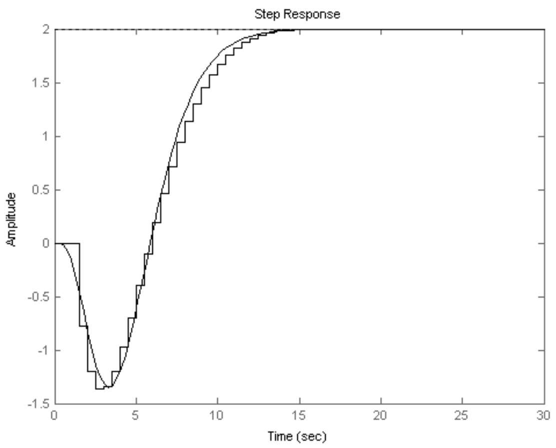

Fig. 2 Step responses of models (59) and (61)

Control responses are in figures 3, 4, 5 and 6.

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.

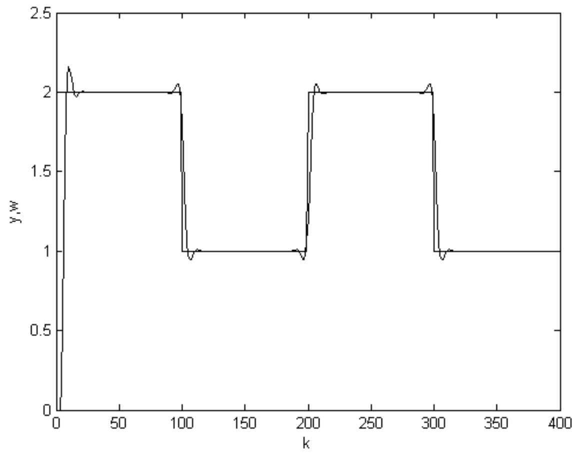

Fig. 3 Control of model (60)

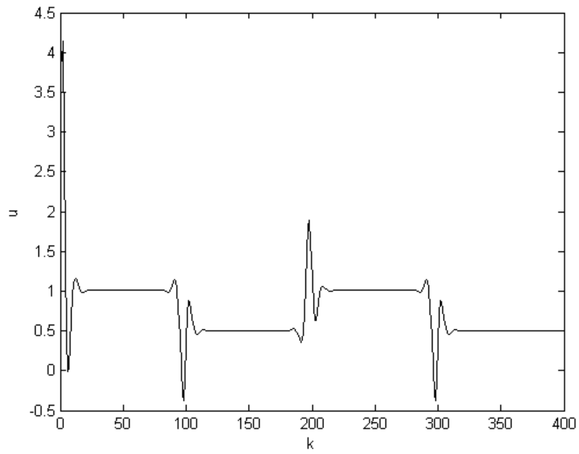

Fig. 4 Control of model (60) –manipulated variable

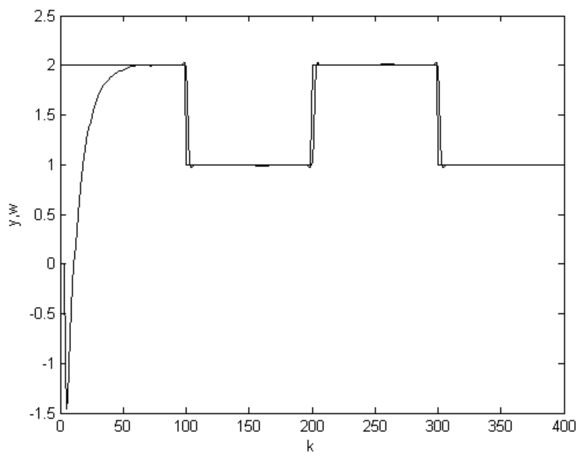

Fig. 5 Control of model (61)

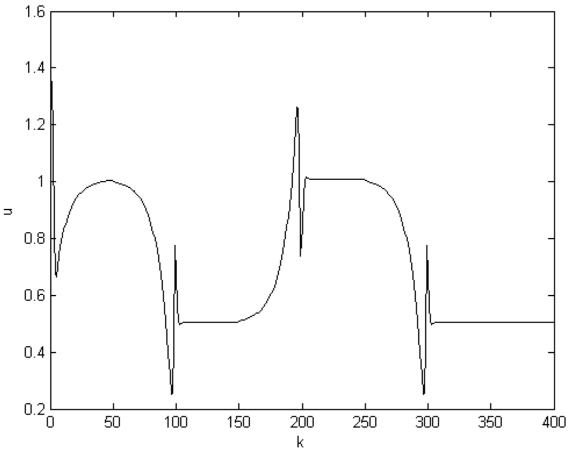

Fig. 6 Control of model (61) –manipulated variable

6.2 TITO control

A TITO system with two steps of time-delay

$$\boldsymbol{G}(z) = \boldsymbol{A}^{-1}\left(z^{-1}\right)\boldsymbol{B}\left(z^{-1}\right)z^{-2} \tag{62}$$

described by polynomial matrices (63) –(64) was chosen as an example

$$\boldsymbol{A}\left(z^{-1}\right) = \begin{bmatrix} 1-0.58z^{-1}+0.17z^{-2} & -0.02z^{-1}+0.17z^{-2} \\ 0.01z^{-1}-0.08z^{-2} & 1-0.45z^{-1}-0.08z^{-2} \end{bmatrix} \tag{63}$$

$$\boldsymbol{B}\left(z^{-1}\right) = \begin{bmatrix} -0.0035z^{-1}+0.09z^{-2} & 0.14z^{-1}+0.21z^{-2} \\ 0.27z^{-1}+0.31z^{-2} & -0.03z^{-1}-0.34z^{-2} \end{bmatrix} \tag{64}$$

In order to compute the right control action it was necessary to determine the predictions from 2+1 to 2+*N*.

$$\begin{bmatrix} \hat{\boldsymbol{y}}(k+3) \\ \hat{\boldsymbol{y}}(k+4) \\ \hat{\boldsymbol{y}}(k+5) \end{bmatrix} = \boldsymbol{G}\Delta\boldsymbol{u} + \boldsymbol{y}_0 \tag{65}$$

The computation of the free response was then modified to

$$\boldsymbol{y}_0 = \left[\begin{array}{cc|cc|cc} g_{31} & g_{32} & g_{51} & g_{52} & p_{51} & p_{52} \\ g_{41} & g_{42} & g_{61} & g_{62} & p_{61} & p_{62} \\ \hline g_{51} & g_{52} & g_{71} & g_{72} & p_{71} & p_{72} \\ g_{61} & g_{62} & g_{81} & g_{82} & p_{81} & p_{82} \\ \hline g_{71} & g_{72} & g_{91} & g_{92} & p_{91} & p_{92} \\ g_{81} & g_{82} & g_{101} & g_{102} & p_{101} & p_{102} \end{array}\right] \left[\begin{array}{c} \Delta u_1(k-1) \\ \Delta u_2(k-1) \\ \hline \Delta u_1(k-2) \\ \Delta u_2(k-2) \\ \hline \Delta u_1(k-3) \\ \Delta u_2(k-3) \end{array}\right] +$$

$$+ \left[\begin{array}{cc|cc|cc} q_{51} & q_{52} & q_{53} & q_{54} & q_{55} & q_{56} \\ q_{61} & q_{62} & q_{63} & q_{64} & q_{65} & q_{66} \\ \hline q_{71} & q_{72} & q_{73} & q_{34} & q_{75} & q_{76} \\ q_{81} & q_{82} & q_{83} & q_{44} & q_{85} & q_{86} \\ \hline q_{91} & q_{92} & q_{93} & q_{54} & q_{95} & q_{96} \\ q_{101} & q_{102} & q_{103} & q_{64} & q_{105} & q_{106} \end{array}\right] \left[\begin{array}{c} y_1(k) \\ y_2(k) \\ \hline y_1(k-1) \\ y_2(k-1) \\ \hline y_1(k-2) \\ y_2(k-2) \end{array}\right] =$$

$$= \begin{bmatrix} \boldsymbol{G}_2 & \boldsymbol{G}_3 & \boldsymbol{P}_3 \\ \boldsymbol{G}_3 & \boldsymbol{G}_4 & \boldsymbol{P}_4 \\ \boldsymbol{G}_4 & \boldsymbol{G}_5 & \boldsymbol{P}_5 \end{bmatrix} \begin{bmatrix} \Delta\boldsymbol{u}(k-1) \\ \Delta\boldsymbol{u}(k-2) \\ \Delta\boldsymbol{u}(k-3) \end{bmatrix} + \begin{bmatrix} \boldsymbol{Q}_{31} & \boldsymbol{Q}_{32} & \boldsymbol{Q}_{33} \\ \boldsymbol{Q}_{41} & \boldsymbol{Q}_{42} & \boldsymbol{Q}_{43} \\ \boldsymbol{Q}_{51} & \boldsymbol{Q}_{52} & \boldsymbol{Q}_{53} \end{bmatrix} \begin{bmatrix} \boldsymbol{y}(k) \\ \boldsymbol{y}(k-1) \\ \boldsymbol{y}(k-2) \end{bmatrix} =$$

$$= \boldsymbol{P}_1 \begin{bmatrix} \Delta\boldsymbol{u}(k-1) \\ \Delta\boldsymbol{u}(k-2) \\ \Delta\boldsymbol{u}(k-3) \end{bmatrix} + \boldsymbol{Q}_1 \begin{bmatrix} \boldsymbol{y}(k) \\ \boldsymbol{y}(k-1) \\ \boldsymbol{y}(k-2) \end{bmatrix} \tag{66}$$

Control responses are in Fig. 7 and Fig. 8.

The length of the prediction horizon was set to $N = 10$. The length of the control horizon was also set to $N_u = 10$. The coefficient λ was taken as equal to 0,5.

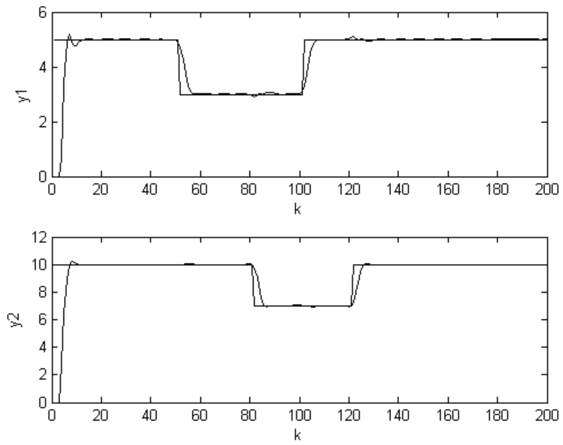

Fig. 7 Simulation results – controlled variable.

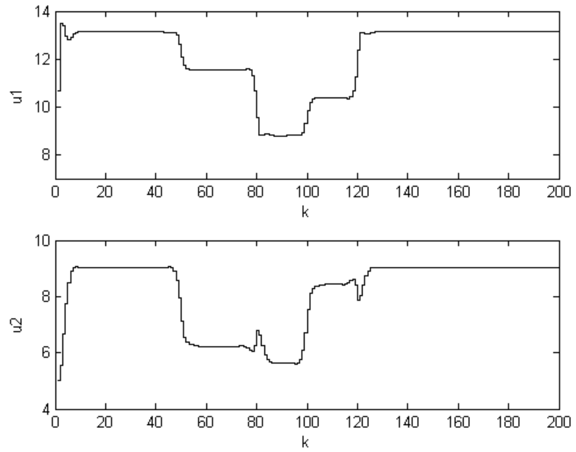

Fig. 8 Simulation results – manipulated variable.

7 Conclusions

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 simulation verification provided good control results. Asymptotic tracking of the reference signal was achieved in both 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. The analogical algorithm for control of the multivariable time-delay systems was also designed. The control of the two – input/two – output system with two steps of time-delay was verified by simulation. Good simulation control results were achieved. Further research can be focused on an extension of the proposed method for control of 2-D (two-Dimensional) discrete time systems.

References:

[1] A. O’Dwyer, Classification of techniques for the compensation of time delayed processes. Part 1: Parameter optimised controllers, *Proc. Of the 3rd IMACS/IEE International Multiconference on Circuits, Systems, Communications and Computers*, Athens, Greece, 1999, 176-186.

[2] A. O’Dwyer, Classification of techniques for the compensation of time delayed processes. Part 2: Structurally optimised controllers, *Proc. of the 3rd IMACS/IEE International Multiconference on Circuits, Systems,Communications and Computers*, Athens, Greece, 1999, 187-198.

[3] J. P. Richard, Time-delay systems: an overview of some recent advances and open problems, *Automatica*, 39, 2003, 1667-1694.

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

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

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

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

[8] D. W. Clarke, C. Mohtadi, P. S. Tuffs, Generalized predictive control, part I: the basic algorithm. *Automatica*, 23, 1987, 137-148.

[9] D. W. Clarke, C. Mohtadi, P. S. Tuffs, Generalized predictive control, part II: extensions and interpretations. *Automatica*, 23, 1987, 149-160.

[10] F. G. Shinskey, *Process Control System*, 4th ed.; McGraw-Hill. New York, 1996.

[11] M. Kubalčík, M., V. Bobál, Adaptive Control of Coupled Drives Apparatus Based on Polynomial Theory. In Proc. IMechE Vol. 220 Part I: *J. Systems and Control Engineering*, 220(I7), 2006, 641-654.

[12] 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.

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

[14] V. Bobál, M. Kubalčík, P. Chalupa and P. Dostál, Identification and Digital Control of Higher-Order Processes Using Predictive Strategy. *Proceedings of 26th European Conference on Modelling and Simulation*, 2012, 426-433.