PREDICTIVE CONTROL WITH FILTERING OF INPUT AND OUTPUT VARIABLES

Marek Kubalcik, Vladimir Bobal
Tomas Bata University in Zlín
Department of Process Control, Faculty of Applied Informatics
Nad Stráněmi 4511, 76005, Zlín
Czech Republic
E-mail: kubalcik@fai.utb.cz

KEYWORDS
predictive control, disturbance modelling, colouring polynomial, filtering of variables, transfer function models

ABSTRACT

The paper is focused on an implementation of a predictive controller with a colouring filter *C* in a disturbance model. The filter is often essential for practical applications of predictive control based on transfer function models. It is commonly considered as a design parameter because it has direct effects on closed loop performance. In this paper a computation of predictions for the case with the colouring polynomial is introduced. The computation is based on a particular model of the controlled system in the form of transfer function which is commonly used for description of a range of processes. Performance of closed loop system with and without the colouring polynomial in the disturbance model was compared.

INTRODUCTION

Model Predictive Control (MPC) or only Predictive Control (Camacho and Bordons 2004, Morari and Lee 1999, Mikleš and Fikar 2008) 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 especially in the discrete domain. The basic idea of the generalized predictive control (Clarke and Mohtadi 1987, Clarke and Mohtadi 1987) 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.
An implementation of a predictive controller based on a transfer function model with a colouring filter *C* in a disturbance model is described in this paper. The filter is often essential for practical applications of predictive control based on transfer function models. Surveys of practical applications of predictive control are presented in (Quinn and Bandgwell 1996, Quinn and Bandgwell 2000, Quinn and Bandgwell 2003). It is commonly considered as a design parameter because it has direct effects on closed loop performance. A computation of predictions for the case with the colouring polynomial is introduced. The computation is based on a particular model of the controlled system in the form of transfer function which is commonly used for description of a range of processes.The filtering of variables is the equivalent of the colouring polynomial in the noise model. It is practically very difficult to estimate the coefficients of the colouring polynomial. A model with the *C*-polynomial is then utilized as an example with filtering of input and output variables when the polynomial *C* is a tuning parameter. In the paper are derived prediction equations for an input output model in the form of transfer function both for the case with the *C*-filter and without the *C*-filter. Performance of closed loop system with and without the colouring polynomial in the disturbance model was compared.

MODEL OF THE CONTROLLED 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. It can be expressed by following transfer function

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

A widely used model in general model predictive control is the CARIMA (controller autoregressive integrated moving average) model which we can obtain by adding a disturbance model as

$$A(z^{-1})y(k)=B(z^{-1})u(k)+\frac{C(z^{-1})}{\Delta}n(k) \tag{2}$$

where n is a non-measurable random disturbance that is assumed to have zero mean value and constant covariance and $\Delta=1-z^{-1}$. C is the colouring polynomial. For purpose of simplification it is often supposed to be equal to 1 (Camacho and Bordons 2004). In Model Predictive Control it is also common

Proceedings 29th European Conference on Modelling and Simulation ©ECMS Valeri M. Mladenov, Petia Georgieva, Grisha Spasov, Galidiya Petrova (Editors)
ISBN: 978-0-9932440-0-1 / ISBN: 978-0-9932440-1-8 (CD)

to treat C as a design parameter (Clarke and Mohtadi 1987, Clarke and Mohtadi 1987, Yoon and Clarke 1995). In this paper will be compared cases when C=1 and when C is supposed as the design parameter.

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

$$J(k)=\sum_{j=1}^{N}\boldsymbol{e}(k+j)^2+\lambda\sum_{j=1}^{N_u}\Delta\boldsymbol{u}(k+j)^2 \tag{3}$$

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, 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

$$\hat{\boldsymbol{y}}=\boldsymbol{G}\Delta\boldsymbol{u}+\boldsymbol{y}_0 \tag{4}$$

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 which contains values of the step sequence. It is given as

$$G=\begin{bmatrix} g_1 & 0 & 0 & \cdots & 0 \\ g_2 & g_1 & 0 & \cdots & 0 \\ g_3 & g_2 & g_1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ g_N & g_{N-1} & g_{N-2} & \cdots & g_{N-N_u+1} \end{bmatrix} \tag{5}$$

The criterion (4) can be written in a general vector form

$$J=(\hat{\boldsymbol{y}}-\boldsymbol{w})^T(\hat{\boldsymbol{y}}-\boldsymbol{w})+\lambda\Delta\boldsymbol{u}^T\Delta\boldsymbol{u} \tag{6}$$

where $\boldsymbol{w}$ is a vector of the reference trajectory. The criterion can be modified using the expression (6) to

$$J=2\boldsymbol{g}^T\Delta\boldsymbol{u}+\Delta\boldsymbol{u}^T\boldsymbol{H}\Delta\boldsymbol{u} \tag{7}$$

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

$$\boldsymbol{g}^T=\boldsymbol{G}^T(\boldsymbol{y}_0-\boldsymbol{w}) \tag{8}$$

$$\boldsymbol{H}=\boldsymbol{G}^T\boldsymbol{G}+\lambda\boldsymbol{I} \tag{9}$$

As it was mentioned handling of constraints is one of main advantages of predictive control. General formulation of predictive control with constraints is then as follows

$$\min_{\Delta\boldsymbol{u}} 2\boldsymbol{g}^T\Delta\boldsymbol{u}+\Delta\boldsymbol{u}^T\boldsymbol{H}\Delta\boldsymbol{u} \tag{10}$$

owing to

$$\boldsymbol{A}\Delta\boldsymbol{u}\le\boldsymbol{b} \tag{11}$$

The inequality (11) expresses the constraints in a compact form.

COMPUTATION OF PREDICTIONS C=1

An important task in predictive control is computation of predictions for arbitrary prediction and control horizons.
The difference equation of the CARIMA model without the unknown term can be expressed as:

$$\begin{aligned} y(k)&=(1-a_1)y(k-1)+(a_1-a_2)y(k-2)+\\ &+a_2y(k-3)+b_1\Delta u(k-1)+b_2\Delta u(k-2) \end{aligned} \tag{12}$$

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.

$$\begin{aligned} \hat{y}(k+1)&=(1-a_1)y(k)+(a_1-a_2)y(k-1)+a_2y(k-2)+\\ &+b_1\Delta u(k)+b_2\Delta u(k-1)\\ \hat{y}(k+2)&=(1-a_1)\hat{y}(k+1)+(a_1-a_2)y(k)+a_2y(k-1)+\\ &+b_1\Delta u(k+1)+b_2\Delta u(k)\\ \hat{y}(k+3)&=(1-a_1)\hat{y}(k+2)+(a_1-a_2)\hat{y}(k+1)+a_2y(k)+\\ &+b_1\Delta u(k+2)+b_2\Delta u(k+1) \end{aligned} \tag{13}$$

The predictions after modification can be written in a matrix form

$$\begin{aligned} \begin{bmatrix}\hat{y}(k+1)\\ \hat{y}(k+2)\\ \hat{y}(k+3)\end{bmatrix}&=\begin{bmatrix} g_1 & 0\\ g_2 & g_1\\ g_3 & g_2\end{bmatrix}\begin{bmatrix}\Delta u(k)\\ \Delta u(k+1)\end{bmatrix}+\\ &+\begin{bmatrix}p_1\\ p_2\\ p_3\end{bmatrix}\Delta u(k-1)+\begin{bmatrix} q_{11} & q_{12} & q_{13}\\ q_{21} & q_{22} & q_{23}\\ q_{31} & q_{32} & q_{33}\end{bmatrix}\begin{bmatrix}y(k)\\ y(k-1)\\ y(k-2)\end{bmatrix} \end{aligned} \tag{14}$$

$$\begin{bmatrix}\hat{y}(k+1)\\ \hat{y}(k+2)\\ \hat{y}(k+3)\end{bmatrix}=\boldsymbol{G}\begin{bmatrix}\Delta u(k)\\ \Delta u(k+1)\end{bmatrix}+\boldsymbol{P}\Delta u(k-1)+\boldsymbol{Q}\begin{bmatrix}y(k)\\ y(k-1)\\ y(k-2)\end{bmatrix} \tag{15}$$

$$\begin{aligned} &\boldsymbol{y}(k+1)=\boldsymbol{G}\Delta\boldsymbol{u}(k+j-1)+\boldsymbol{P}\Delta\boldsymbol{u}(k-1)+\boldsymbol{Q}\boldsymbol{y}(k+j-1)\\ &j\le N \end{aligned} \tag{16}$$

where

$$\boldsymbol{G}\begin{bmatrix}\Delta u(k)\\ \Delta u(k+1)\end{bmatrix}=\begin{bmatrix} b_1 & 0\\ b_1(1-a_1)+b_2 & b_1\\ (a_1-a_2)b_1+(1-a_1)^2 b_1+(1-a_1)b_2 & (1-a_1)b_1+b_2\end{bmatrix}\begin{bmatrix}\Delta u(k)\\ \Delta u(k+1)\end{bmatrix} \tag{17}$$

$$\boldsymbol{P}\Delta u(k-1)=\begin{bmatrix} b_2\\ (1-a_1)b_2\\ (1-a_1)^2 b_2+(a_1-a_2)b_2\end{bmatrix}\Delta u(k-1) \tag{18}$$

$$\boldsymbol{Q}\begin{bmatrix}y(k)\\ y(k-1)\\ y(k-2)\end{bmatrix}=$$
$$=\begin{bmatrix}(1-a_1) & (a_1-a_2) & a_2\\ (1-a_1)^2+(a_1-a_2) & (1-a_1)(a_1-a_2)+a_2 & a_2(1-a_1)\\ (1-a_1)^3+2(1-a_1)(a_1-a_2)+a_2 & (1-a_1)^2(a_1-a_2)+a_2(1-a_1)+(a_1-a_2)^2 & a_2(1-a_1)^2+(a_1-a_2)a_2\end{bmatrix}\begin{bmatrix}y(k)\\ y(k-1)\\ y(k-2)\end{bmatrix} \tag{19}$$

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

$$p_4=(1-a_1)p_3+(a_1-a_2)p_2+a_2 p_1 \tag{20}$$

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

The recursion of the matrix ***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:

$$g_4=(1-a_1)g_3+(a_1-a_2)g_2+a_2 g_1 \tag{22}$$

COMPUTATION OF PREDICTIONS C≠1

Computation of predictions for C≠1 is solved for example in (Rossitter 2003). Including the C-filter the CARIMA model takes the form

$$\Delta A(z^{-1})y(k)=B(z^{-1})\Delta u(k)+C(z^{-1})n(k) \tag{23}$$

Equation (23) can be modified to

$$\Delta A(z^{-1})\frac{y(k)}{C(z^{-1})}=B(z^{-1})\frac{\Delta u(k)}{C(z^{-1})}+n(k) \tag{24}$$

Where the unknown term is supposed to be the white noise and the input and output variables are filtered. Using of (24) for prediction improves prediction accuracy.
The filtered variables are defined as

$$y_f(k)=\frac{y(k)}{C(z^{-1})} \tag{25}$$

$$u_f(k)=\frac{u(k)}{C(z^{-1})} \tag{26}$$

In this case the polynomial C is a design parameter. It is a stable polynomial. In case of the system (1) it was chosen to be of the second order as

$$C(z^{-1})=1+c_1 z^{-1}+c_2 z^{-2} \tag{27}$$

The input and output data are filtered before prediction. $1/C$ is a low-pass filter which reduces high frequency noise. It is easy to prove by simulation that the cases when the noise is coloured (23) and when the noise is white and the input and output variables are filtered (24) are equal.
The prediction equation for filtered variables takes the following form

$$\hat{\boldsymbol{y}}_f(k+j)=\boldsymbol{G}\Delta\boldsymbol{u}_f(k+j-1)+\boldsymbol{P}\Delta\boldsymbol{u}_f(k-1)+\boldsymbol{Q}\boldsymbol{y}_f(k-j+1)$$
$$j\le N \tag{28}$$

For practical application the equation (28) is inapplicable. Prediction of the unfiltered output must be expressed by means of future control increments.
The relationship between filtered and unfiltered variables can be expressed as follows

$$y_f(k)=\frac{y(k)}{1+c_1 z^{-1}+c_2 z^{-2}} \tag{29}$$

$$y(k)=y_f(k)+c_1 y_f(k-1)+c_2 y_f(k-2) \tag{30}$$

For three step ahead predictions

$$\begin{aligned}\hat{y}(k+1)&=y_f(k+1)+c_1 y_f(k)+c_2 y_f(k-1)\\ \hat{y}(k+2)&=y_f(k+2)+c_1 y_f(k+1)+c_2 y_f(k)\\ \hat{y}(k+3)&=y_f(k+3)+c_1 y_f(k+2)+c_2 y_f(k+1)\end{aligned} \tag{31}$$

In a matrix form the equations (31) can be expressed as follows

$$\begin{pmatrix}\hat{y}(k+1)\\ \hat{y}(k+2)\\ \hat{y}(k+3)\end{pmatrix}=\begin{pmatrix}1 & 0 & 0\\ c_1 & 1 & 0\\ c_2 & c_1 & 1\end{pmatrix}\begin{pmatrix}y_f(k+1)\\ y_f(k+2)\\ y_f(k+3)\end{pmatrix}+$$
$$+\begin{pmatrix}c_1 & c_2 & 0\\ c_2 & 0 & 0\\ 0 & 0 & 0\end{pmatrix}\begin{pmatrix}y_f(k)\\ y_f(k-1)\\ y_f(k-2)\end{pmatrix} \tag{32}$$

The relationship between filtered and unfiltered control increments can be expressed similarly. Using matrix notation we can define following equations

$$\hat{\boldsymbol{y}}(k+j)=\boldsymbol{C}_C\boldsymbol{y}_f(k+j)+\boldsymbol{H}_C\boldsymbol{y}_f(k-j+1) \tag{33}$$

$$\Delta\boldsymbol{u}(k+j)=\boldsymbol{C}_C\Delta\boldsymbol{u}_f(k+j)+\boldsymbol{H}_C\Delta\boldsymbol{u}_f(k-j+1) \tag{34}$$

Where matrices $\boldsymbol{C}_c$ and $\boldsymbol{H}_c$ are defined as follows

$$\boldsymbol{C}_C = \begin{pmatrix} 1 & 0 & 0 \\ c_1 & 1 & 0 \\ c_2 & c_1 & 1 \end{pmatrix} \tag{35}$$

$$\boldsymbol{H}_C = \begin{pmatrix} c_1 & c_2 & 0 \\ c_2 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \tag{36}$$

From equations (33) and (34) we can express the filtered variables

$$\boldsymbol{y}_f(k+j) = \boldsymbol{C}_C^{-1}\left(\hat{\boldsymbol{y}}(k+j) - \boldsymbol{H}_C \boldsymbol{y}_f(k-j+1)\right) \tag{37}$$

$$\Delta \boldsymbol{u}_f(k+j) = \boldsymbol{C}_C^{-1}\left(\Delta \boldsymbol{u}(k+j) - \boldsymbol{H}_C \Delta \boldsymbol{u}_f(k-j+1)\right) \tag{38}$$

After substitution of equations (37) and (38) to equation (28) we obtain

$$\begin{aligned} &\boldsymbol{C}_C^{-1}\left(\hat{\boldsymbol{y}}(k+j) - \boldsymbol{H}_C \boldsymbol{y}_f(k-j+1)\right) = \\ &\boldsymbol{G}\boldsymbol{C}_C^{-1}\left(\Delta \boldsymbol{u}(k+j-1) - \boldsymbol{H}_C \Delta \boldsymbol{u}_f(k-1)\right) + \\ &+ \boldsymbol{P}\Delta \boldsymbol{u}_f(k-1) + \boldsymbol{Q}\boldsymbol{y}_f(k-j+1) \end{aligned} \tag{39}$$

After modification we obtain resulting equation of the predictor

$$\begin{aligned} &\hat{\boldsymbol{y}}(k+j) = \boldsymbol{G}\Delta \boldsymbol{u}(k+j-1) + [\boldsymbol{C}_C\boldsymbol{P} - \boldsymbol{G}\boldsymbol{H}_C]\Delta \boldsymbol{u}_f(k-1) + \\ &+ [\boldsymbol{H}_C + \boldsymbol{C}_C\boldsymbol{Q}]\boldsymbol{y}_f(k-j+1) \end{aligned} \tag{40}$$

We can establish following substitutions

$$\widetilde{\boldsymbol{P}} = [\boldsymbol{C}_C\boldsymbol{P} - \boldsymbol{G}\boldsymbol{H}_C] \tag{41}$$

$$\widetilde{\boldsymbol{Q}} = [\boldsymbol{H}_C + \boldsymbol{C}_C\boldsymbol{Q}] \tag{42}$$

The prediction equation then can be written in the form

$$\hat{\boldsymbol{y}}(k+j) = \boldsymbol{G}\Delta \boldsymbol{u}(k+j-1) + \widetilde{\boldsymbol{P}}\Delta \boldsymbol{u}_f(k-1) + \widetilde{\boldsymbol{Q}}\boldsymbol{y}_f(k-j+1) \tag{43}$$

SIMULATION VERIFICATION

Verification by simulation was carried out on a range of plants with various dynamics. The control of the model below is given here as an example.

$$G(s) = \frac{3}{5s^2 + 6s + 1} \tag{44}$$

It does not exist a systematic way for selection of the filter C. Its selection is mostly based on intuition. In our example the filter was chosen as

$$C(z^{-1}) = 1 + 0{,}8z^{-1} + 0{,}05z^{-2} \tag{45}$$

The sampling period was tuned experimentally and the best value was T_0 = 2 *s*. The controlled variable was affected by a noise with zero mean value and constant covariance. Simulation sampling of noise was 0,1 *s*.
In figures 1-6 are simulation results. Figures 1,3 and 5 show time responses of the control without the filtering of the variables introduced in section 4. Figures 2, 4 and 6 show time responses of the control with the filtering of variables described in section 5.
In figures 1 and 2 there is the response of the controlled variable taken by 0,1 *s*. It means with the same sampling period as the simulation noise. Simulation results in this figure are the closest to the reality. In figures 3 and 4 there is the controlled variable taken by 2 *s*. It means with the same sampling period which is used for the control. The data then simulates measured values. In figures 5 and 6 is the manipulated variable.
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 set to N = 5. The length of the control horizon was also set to N_u = 5. The coefficient λ was taken as equal to 0,1.
It is necessary to emphasize that the displayed inputs and outputs in the graphs are not filtered. The filtered values are used only for computation of systems output predictions and consequently for computation of the control law. The displayed inputs and outputs are real unfiltered values.

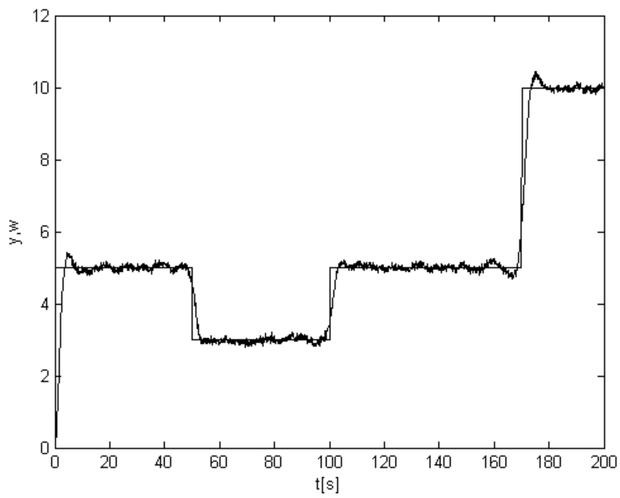

Figure 1: Controlled variable sampled by 0,1*s* – case without filtering of variables

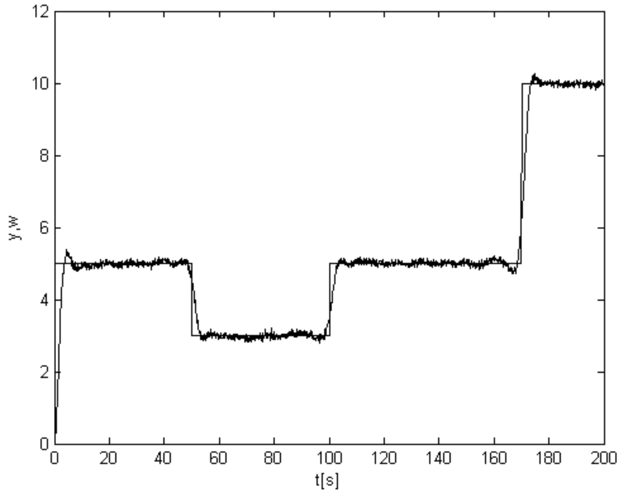

Figure 2: Controlled variable sampled by 0,1*s* – case with filtering of variables

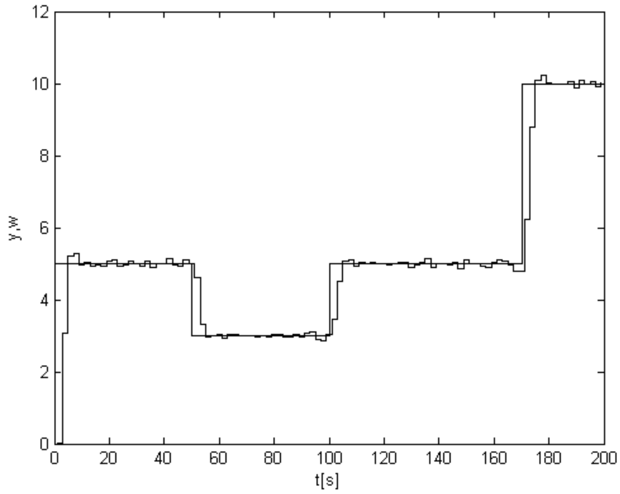

Figure 3: Controlled variable sampled by 2*s* – case without filtering of variables

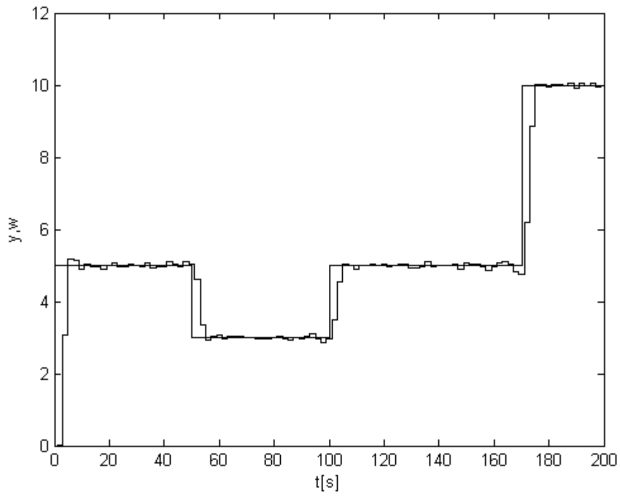

Figure 4: Controlled variable sampled by 2*s* – case with filtering of variables

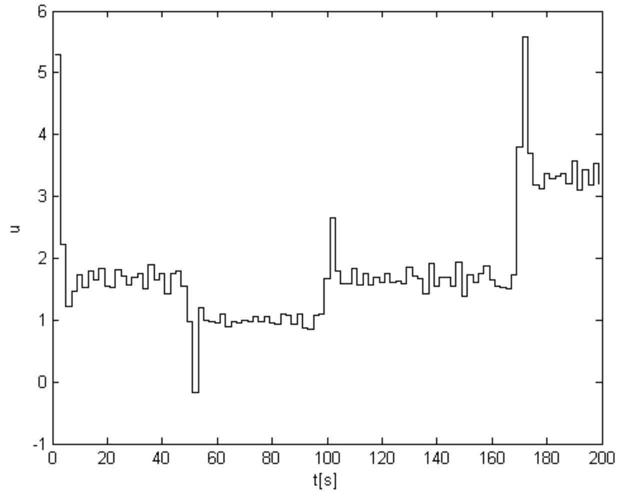

Figure 5: Manipulated variable – case without filtering of variables

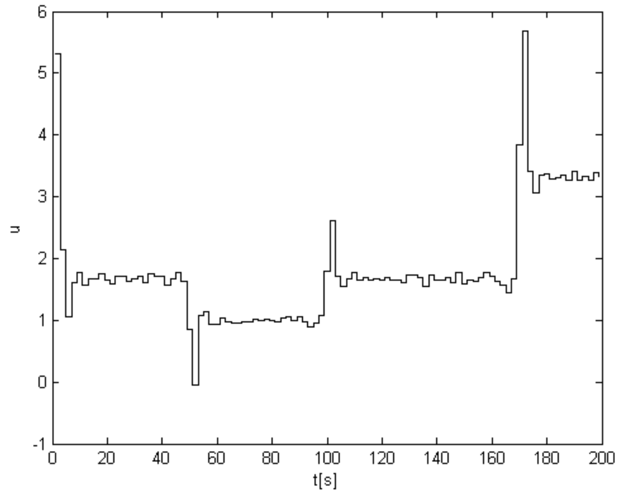

Figure 6: Manipulated variable – case with filtering of variables

CONCLUSIONS

Specific self-contained prediction equations for the input-output model in the form of transfer function were derived for the case with filtering of the input and output variables. Simulations, where the filtered variables are used for computation of the control law and the manipulated variable, were performed. In the simulation results are displayed real unfiltered variables. By simulation control of a range of systems were compared control results of cases with and without the C-filter. In the paper there is introduced one simulation example. The best achieved results are shown. The C-filter is a tuning parameter for which setting we do not have available any exact methodology. The filter was designed by try it and see approach as a low pass filter. Slightly better results were achieved in case with the C-filter particularly regarding rate of oscillations of the input and output variables. It is obvious that the variables are more settled in case with the C-filter. The filter reduces sensitivity of the closed loop system to high frequency noise. Cost for this improvement is a relatively difficult setting of the C-filter as a parameter.

REFERENCES

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

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

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

Mikleš J., Fikar M. 2008. *Process Modelling, Optimisation and Control,* Berlin: Springer-Verlag.

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

Quin S. J., Bandgwell T. A. 1996. An overview of industrial model predictive control technology. *Proceedings of the Chemical Process Control – V.* Vol. 93 of *AIChE Symposium Series.* CACHE & AIChE. Tahoe City, CA, USA, 232-256.

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

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

Rossiter J. A. 2003. *Model Based Predictive Control: a Practical Approach,* CRC Press.

Yoon T.-W., Clarke D.W. 1995. Observer design in receding horizon predictive control, International Journal of Control, 61, 171-191.

AUTHOR BIOGRAPHIES

VLADIMÍR BOBÁL graduated in 1966 from the Brno University of Technology, Czech Republic. He received his Ph.D. degree in Technical Cybernetics at Institute of Technical Cybernetics, Slovak Academy of Sciences, Bratislava, Slovak Republic. He is now Professor at the Department of Process Control, Faculty of Applied Informatics of the Tomas Bata University in Zlín, Czech Republic. His research interests are adaptive and predictive control, system identification, time-delay systems and CAD for automatic control systems. You can contact him on email address bobal@fai.utb.cz.

MAREK KUBALČÍK graduated in 1993 from the Brno University of Technology in Automation and Process Control. He received his Ph.D. degree in Technical Cybernetics at Brno University of Technology in 2000. From 1993 to 2007 he worked as senior lecturer at the Faculty of Technology, Brno University of Technology. From 2007 he has been working as an associate professor at the Department of Process Control, Faculty of Applied Informatics of the Tomas Bata University in Zlín, Czech Republic. Current work covers following areas: control of multivariable systems, self-tuning controllers, predictive control. His e-mail address is: kubalcik@fai.utb.cz.