2017 21st International Conference on Process Control (PC)
June 6–9, 2017, Štrbské Pleso, Slovakia

Predictive Control of the Magnetic Levitation Model

Lukáš Rušar, Adam Krhovják, Vladimír Bobál
Department of Process Control
Faculty of Applied Informatics, Tomas Bata University in Zlin
Czech Republic
e-mail: rusar@fai.utb.cz

Abstract — **This paper presents a possible way to control the a very fast nonlinear systems. The system of the magnetic levitation was chosen as an exemplar process. This is an example of the process with a sampling period in order of milliseconds. We chose a predictive control method to control this system. The state-space CARIMA mathematical model is used for prediction of the output values. This paper describes the magnetic levitation model, its linearization, prediction of the output values and a calculation of the control signal by using a predictor-corrector method which turned out to be the best solution out of the selected ones. The results compare several optimization methods to achieve the fastest calculation of the control signal. All of the simulation was done in Matlab.**

Keywords— predictive control, state-space, magletic levitation, predictor-corrector

I. INTRODUCTION

The real world contains many types of processes. Many of them are nonlinear and their mathematical models are very complex. These processes are also differed in the required sampling period. This paper focuses on the very fast processes with a sampling period in the order of milliseconds. It is very difficult to control these processes due to their complexity. The basic control methods may not handle with this situation with required precision so we need a more advanced method. The predictive control is a great example of the modern control method that can be used to solve the complex control problems.

This method is based on the mathematical model of the controlled process which is used to the prediction of the output values on the chosen time horizon. This time horizon should be long enough to cover the step response of the controlled system. The model of the magnetic levitation is described by the state-space CARIMA mathematical model for the single-input single-output (SISO) system [1].

The goal of the predictive control method is to calculate the control signal which guarantee the required output signal in the near future time horizon. This is achieved by minimization of the cost function that usually has a quadratic form and it minimize the differences between the reference value and the output value and the control signal increments. We can also take into account the constraints of the process variables in the cost function minimization process. Several method such as quadratic programming method, fast-gradient method, predictor-corrector method etc. can be used to minimize the cost function [2].

However, the chosen CARIMA mathematical model used to the prediction of the output values is linear whereas the model of the magnetic levitation is nonlinear. This means that we have to linearize the nonlinear mathematical model. The final linear model is made by combination of several linearized models about different operation points.

This paper is divided into the following sections. The model of the magnetic levitation is described in the section II. The predictive control and the calculation of the control signal are described in the sections III. and IV. The section V. shows the results of the research and section VI. contains the conclusion.

II. THE MODEL OF THE MAGNETIC LEVITATION

The mathematical of the magnetic levitation describes the behavior of the magnetic levitation system CE 152 which is a laboratory-scale model designed by TQ Education and Training Ltd for studying system dynamics and experimenting with control algorithms. It demonstrates control problems associated with nonlinear unstable systems. The system consists of a coil levitating a steel ball in the magnetic field with the position sensed by an inductive linear sensor connected to an A/D converter [3].

Fig. 1. The CE 152 magnetic levitation apparatus

The coil is driven by a power amplifier connected to a D/A converter. A basic control task is to control the position of the ball freely levitating in the magnetic field of the coil. From the control theory point of view, the magnetic levitation system is a nonlinear unstable system with one input and one output.

A. Mathematical model

The mathematical model of the magnetic levitation system can be described by a second-order nonlinear differential equation

$$\frac{m_k}{k_{AD}k_x}\ddot{y}-\frac{k_{fv}}{k_{AD}k_x}\dot{y}=\frac{k_{DA}^2k_i^2u^2k_c}{\left(\frac{y-k_{AD}y_0}{k_{AD}k_x}-x_0\right)^2}-m_k g \tag{1}$$

where y is the controlled output variable which is ball position and u is the control input signal. The other symbols are defined in Table 1 [3].

TABLE I. PARAMETERS OF THE MODEL

Symbol	Meaning	Value
k_{AD}	A/D converter gain	0.2 MU/V
k_{DA}	D/A converter gain	20 V/MU
k_{fv}	damping constant	0.02 Ns/m
k_x	position sensor gain	821 V/M
k_i	power amplifier gain	0.3 A/V
k_c	coil constant	$1.768\times10^{-6}\,\mathrm{Nm^2/A^2}$
m_k	ball mass	$8.27\times10^{-3}\,\mathrm{kg}$
x_0	coil offset	$7.6\times10^{-3}\,\mathrm{m}$
g	gravity constant	$9.81\mathrm{m/s^2}$
y_0	position sensor offset	0.0183 V
y	ball position	MU
u	input signal	MU

The differential equation (1) can be expressed in the state-space representation

$$\begin{aligned}\dot{x}_1&=x_2\\ \dot{x}_2&=\frac{k_{fv}}{m_k}x_2+\frac{k_{AD}k_xk_{DA}^2k_i^2u^2k_c}{m_k\left(\frac{x_1-k_{AD}y_0}{k_{AD}k_x}-x_0\right)^2}-k_{AD}k_x g\end{aligned} \tag{2}$$

where x_1 represents the ball position y and x_2 represents its speed. This nonlinear state-space model was linearized about the selected operating points since the chosen predictive control method works with linear models. The new linear model than represents deviations of the process variables from their steady-states (y^s, u^s) chosen as the operating points. The linear state-space model is described by following equations [3,5].

$$\begin{aligned}\boldsymbol{x}&=\boldsymbol{Ax}+\boldsymbol{B}u\\ y&=\boldsymbol{Cx}\end{aligned} \tag{3}$$

where

$$\begin{aligned}\boldsymbol{A}&=\begin{pmatrix}0 & 1\\ \dfrac{2g}{k_{AD}k_iu^s\sqrt{\dfrac{k_c}{m_k g}}} & \dfrac{k_{fv}}{m_k}\end{pmatrix}\\ \boldsymbol{B}&=\begin{pmatrix}0\\ \dfrac{2k_{AD}k_x g}{u^s}\end{pmatrix}\\ \boldsymbol{C}&=\begin{pmatrix}1 & 0\end{pmatrix}\end{aligned} \tag{4}$$

However, this is still a continuous-time model and it needs to be transferred into a discrete-time form suitable for the chosen predictive control method. It can be done by transferring the state-space model into the input-output model

$$A(s)\,y(t)=B(s)\,u(t) \tag{5}$$

and then into its discrete representation

$$\tilde{A}(z^{-1})\,y(k)=B(z^{-1})\Delta u(k) \tag{6}$$

where the polynom $\tilde{A}(z^{-1})$ is

$$\tilde{A}(z^{-1})=(1-z^{-1})A(z^{-1}) \tag{7}$$

and Δ is symbol for backward difference $(1-z^{-1})$. Then the $\Delta u(k)$ is

$$\Delta u(k)=u(k)-u(k-1) \tag{8}$$

III. STATE-SPACE PREDICTIVE CONTROL

The chosen predictive control method uses the state-space CARIMA model for prediction of the output values. This model is described by equation [4,8-10]

$$\begin{aligned}\boldsymbol{x}(k+1)&=\tilde{\boldsymbol{A}}\boldsymbol{x}(k)+\boldsymbol{B}\Delta u(k)\\ y(k)&=\boldsymbol{Cx}(k)\end{aligned} \tag{9}$$

where the vector of state variables has form

$$\begin{aligned}\boldsymbol{x}(k)=[&y(k),y(k-1),\cdots,y(k-na),\\ &\Delta u(k-1),\cdots,\Delta u(k-nb+1)]^T\end{aligned} \tag{10}$$

The matrices $\tilde{\boldsymbol{A}}$, $\boldsymbol{B}$ and $\boldsymbol{C}$ from the model (9) can be expressed as

$$\begin{aligned}\tilde{\boldsymbol{A}}&=\begin{bmatrix}-\tilde{a}_1 & \cdots & -\tilde{a}_{na} & -\tilde{a}_{na+1} & b_2 & \cdots & b_{nb-1} & b_{nb}\\ 1 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0\\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots\\ 0 & \cdots & 1 & 0 & 0 & \cdots & 0 & 0\\ 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0\\ 0 & \cdots & 0 & 0 & 1 & \cdots & 0 & 0\\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots\\ 0 & \cdots & 0 & 0 & 0 & \cdots & 1 & 0\end{bmatrix}\\ \boldsymbol{B}&=[b_1\ \ 0\ \ \cdots\ \ 0\ \ 0\ \ 1\ \ 0\ \ \cdots\ \ 0\ \ 0]^T\\ \boldsymbol{C}&=[1\ \ 0\ \ \cdots\ \ 0\ \ 0]\end{aligned} \tag{11}$$

The values $-\tilde{a}_i$ for $i = 1,\ldots,na+1$ and b_j for $j = 1,\ldots,nb$ consist of the coefficients of the polynoms $\tilde{A}(z^{-1})$ and $B(z^{-1})$ from the equation (6).

The prediction of the output values is obtained recursively using the CARIMA model represented by equation (9). This means that for the prediction of the next output value we obtain the equation

$$y(k+1) = \boldsymbol{Cx}(k+1) \tag{12}$$

where $\boldsymbol{x}(k+1)$ is the first equation of the CARIMA model (9).

The final matrix form of this prediction is

$$\hat{\boldsymbol{y}} = \boldsymbol{Fx} + \boldsymbol{H}_f \Delta\boldsymbol{u}_f \tag{13}$$

where $\hat{\boldsymbol{y}}$ is the vector of the predicted output values and $\Delta\boldsymbol{u}_f$ is the vector of the future control increments

$$\hat{\boldsymbol{y}} = \begin{bmatrix} \hat{y}(k+1) \\ \hat{y}(k+2) \\ \vdots \\ \hat{y}(k+N) \end{bmatrix}$$
$$\Delta\boldsymbol{u}_f = \begin{bmatrix} \Delta u(k) \\ \Delta u(k+1) \\ \vdots \\ \Delta u(k+N) \end{bmatrix} \tag{14}$$

where N is the chosen time horizon for prediction [7,11].

The aim of the predictive control is minimize the difference between the future reference values and the predicted output values and also minimize the control signal demand. The quadratic cost function is used to this optimization problem.

$$J = \left(\boldsymbol{w} - \hat{\boldsymbol{y}}\right)^T \boldsymbol{Q}_\delta \left(\boldsymbol{w} - \hat{\boldsymbol{y}}\right) + \Delta\boldsymbol{u}_f^T \boldsymbol{Q}_\lambda \Delta\boldsymbol{u}_f \tag{15}$$

where $\boldsymbol{w}$ is a vector of the future reference values, $\hat{\boldsymbol{y}}$ is the vector of the predicted outputs values, $\boldsymbol{Q}_\lambda$ and $\boldsymbol{Q}_\delta$ are the diagonal weighting matrices. The weighting coefficients of these matrices have to be positive numbers. The vector $\Delta\boldsymbol{u}_f$ is unknown vector of the future control increments[7,11,12].

Because of the chosen optimization method, this cost function needs to be modified into the following form [11,13]

$$J = \frac{1}{2}\Delta\boldsymbol{u}_f^T \boldsymbol{H}_c \boldsymbol{u} + \boldsymbol{g}^T \Delta\boldsymbol{u}_f \tag{16}$$

where

$$\begin{aligned} \boldsymbol{H}_c &= 2\left(\boldsymbol{Q}_\lambda + \boldsymbol{H}_f^T \boldsymbol{Q}_\delta \boldsymbol{H}_f\right) \\ \boldsymbol{g}^T &= 2\left(\boldsymbol{Fx} - \boldsymbol{w}\right)^T \boldsymbol{Q}_\delta \boldsymbol{H}_f \end{aligned} \tag{17}$$

IV. PREDICTOR-CORRECTOR METHOD

The predictor-corrector method is one of the primal-dual interior-point methods using to solve the inequality constrained convex quadratic problems

$$\begin{aligned} f(x) &= \frac{1}{2}\boldsymbol{x}^T \boldsymbol{Gx} + \boldsymbol{g}^T \boldsymbol{x} \\ \boldsymbol{A}^T \boldsymbol{x} &\geq \boldsymbol{b} \end{aligned} \tag{18}$$

which is exactly the problem that the predictive control solves [15,16]. The equation (18) represents the general formulation of the constrained quadratic problem. The aim is to find the unknown vector $\boldsymbol{x}$ with respect to the chosen constrains representing the future values of the control signal increments according to the equation (16).

This is the iterative method and we have to set the starting points of the unknown vector $\boldsymbol{x}_0$, the vector of the Lagrange multipliers $\boldsymbol{\lambda}_0$ and the slackvector $\boldsymbol{s}_0$ where $\boldsymbol{s} = \boldsymbol{A}^T\boldsymbol{x} - \boldsymbol{b}, \boldsymbol{s} \geq 0$. These starting points serves to calculate the initial residual vectors $\boldsymbol{r}_d$, $\boldsymbol{r}_s$ and $\boldsymbol{r}_{s\lambda}$

$$\begin{aligned} \boldsymbol{r}_d &= \boldsymbol{Gx}_0 + \boldsymbol{g} - \boldsymbol{A\lambda}_0 \\ \boldsymbol{r}_p &= \boldsymbol{s}_0 - \boldsymbol{A}^T \boldsymbol{x}_0 + \boldsymbol{b} \\ \boldsymbol{r}_{s\lambda} &= \boldsymbol{S}_0 \boldsymbol{\Lambda}_0 \boldsymbol{e} \end{aligned} \tag{19}$$

where $\boldsymbol{S}_0$ and $\boldsymbol{\Lambda}_0$ are the diagonal matrices containing the elements of the $\boldsymbol{s}_0$ and $\boldsymbol{\lambda}_0$. The $\boldsymbol{e}$ is vector of ones [15,16].

There is also need to calculate the initial complementarity measure μ which is need for centering parameter σ

$$\mu = \frac{\boldsymbol{s}_0^T \boldsymbol{\lambda}_0}{m} \tag{20}$$

where m is the number of the inequality constraints.

The whole algorithm can be divided into two parts. The first is the calculation of the predictor step and the second is the calculation of the corrector step. The predictor step is calculated by applying the Newton's method around the current point on the equations (19).

$$\begin{bmatrix} \boldsymbol{G} & -\boldsymbol{A} & \boldsymbol{0} \\ -\boldsymbol{A}^T & \boldsymbol{0} & \boldsymbol{I} \\ \boldsymbol{0} & \boldsymbol{S} & \boldsymbol{\Lambda} \end{bmatrix} \begin{bmatrix} \Delta\boldsymbol{x}^{aff} \\ \Delta\boldsymbol{\lambda}^{aff} \\ \Delta\boldsymbol{s}^{aff} \end{bmatrix} = -\begin{bmatrix} \boldsymbol{r}_d \\ \boldsymbol{r}_p \\ \boldsymbol{r}_{s\lambda} \end{bmatrix} \tag{21}$$

The affine scaling direction $\left(\Delta\boldsymbol{x}^{aff}, \Delta\boldsymbol{\lambda}^{aff}, \Delta\boldsymbol{s}^{aff}\right)$ is obtained by solving these equations. Then the scaling parameter α^{aff} for the predictor step is chosen. This parameter have to satisfy the conditions in the equations

$$\begin{aligned} \boldsymbol{\lambda} + \alpha_\lambda^{aff} \Delta\boldsymbol{\lambda}^{aff} &\geq 0 \\ \boldsymbol{s} + \alpha_s^{aff} \Delta\boldsymbol{s}^{aff} &\geq 0 \end{aligned} \tag{22}$$

The final scaling parameter is chosen in the following way:

$$\alpha_\lambda^{aff} = \min_{i:\Delta\lambda_i<0}\left(1, \min\frac{-\lambda_i}{\Delta\lambda_i^{aff}}\right)$$

$$\alpha_s^{aff} = \min_{i:\Delta s_i<0}\left(1, \min\frac{-s_i}{\Delta s_i^{aff}}\right) \tag{23}$$

$$\alpha^{aff} = \min\left(\alpha_\lambda^{aff}, \alpha_s^{aff}\right)$$

Now the complementarity measure μ^{aff} of the predictor step and the centering parameter σ can be calculated.

$$\mu^{aff} = \frac{\left(\boldsymbol{s} + \alpha^{aff}\Delta\boldsymbol{s}^{aff}\right)^T\left(\boldsymbol{\lambda} + \alpha^{aff}\Delta\boldsymbol{\lambda}^{aff}\right)}{m} \tag{24}$$

$$\sigma = \left(\frac{\mu^{aff}}{\mu}\right)^3 \tag{25}$$

Now we can move to the calculation of the corrector step. This is done by adjusting the right hand side of the equation (21) by computed affine scaling direction and the centering parameter. The resulting equation system is shown as equation [15,16]

$$\begin{bmatrix} \boldsymbol{G} & -\boldsymbol{A} & \boldsymbol{0} \\ -\boldsymbol{A}^T & \boldsymbol{0} & \boldsymbol{I} \\ \boldsymbol{0} & \boldsymbol{S} & \boldsymbol{\Lambda} \end{bmatrix}\begin{bmatrix} \Delta\boldsymbol{x} \\ \Delta\boldsymbol{\lambda} \\ \Delta\boldsymbol{s} \end{bmatrix} = -\begin{bmatrix} \boldsymbol{r}_d \\ \boldsymbol{r}_p \\ \boldsymbol{r}_{s\lambda} + \Delta\boldsymbol{S}^{aff}\Delta\boldsymbol{\Lambda}^{aff}\boldsymbol{e} - \sigma\mu\boldsymbol{e} \end{bmatrix} \tag{26}$$

Solving this system gives us the final scaling direction $(\Delta\boldsymbol{x}, \Delta\boldsymbol{\lambda}, \Delta\boldsymbol{s})$. The step length is chosen in the same way it was in the predictor step calculation in the equations (23).

$$\begin{aligned} \boldsymbol{\lambda} + \alpha_\lambda\Delta\boldsymbol{\lambda} &\geq 0 \\ \boldsymbol{s} + \alpha_s\Delta\boldsymbol{s} &\geq 0 \end{aligned} \tag{27}$$

Now we can update the unknown vector $\boldsymbol{x}$, the vector of the Lagrange multipliers $\boldsymbol{\lambda}$ and the slackvector $\boldsymbol{s}$.

$$\begin{aligned} \boldsymbol{x}_{k+1} &= \boldsymbol{x}_k + \alpha\Delta\boldsymbol{x} \\ \boldsymbol{\lambda}_{k+1} &= \boldsymbol{\lambda}_k + \alpha\Delta\boldsymbol{\lambda} \\ \boldsymbol{s}_{k+1} &= \boldsymbol{s}_k + \alpha\Delta\boldsymbol{s} \end{aligned} \tag{28}$$

The final step of this algorithm is updating the residuals vectors $\boldsymbol{r}_d$, $\boldsymbol{r}_s$ and $\boldsymbol{r}_{s\lambda}$ and the complementarity measure μ [15,16].

$$\begin{aligned} \boldsymbol{r}_d &= \boldsymbol{G}\boldsymbol{x} + \boldsymbol{g} - \boldsymbol{A}\boldsymbol{\lambda} \\ \boldsymbol{r}_p &= \boldsymbol{s} - \boldsymbol{A}^T\boldsymbol{x} + \boldsymbol{b} \\ \boldsymbol{r}_{s\lambda} &= \boldsymbol{S}\boldsymbol{\Lambda}\boldsymbol{e} \end{aligned} \tag{29}$$

$$\mu = \frac{\boldsymbol{s}^T\boldsymbol{\lambda}}{m} \tag{30}$$

V. Results

This section shows the simulation results of the magnetic levitation control process. The parameters of the system are show in the Table I. and the mathematical model of the simulated system is shown in the equation (1). This nonlinear model was linearized about five operating points (y^s, u^s) = [(0, 0.2721), (0.25, 0.2177), (0.5, 0.1634), (0.75, 0.1090), (1, 0.0547)]. The final linear model that was used for the output values prediction is calculated as the linear combination of two out of these five models depending of the current output value.

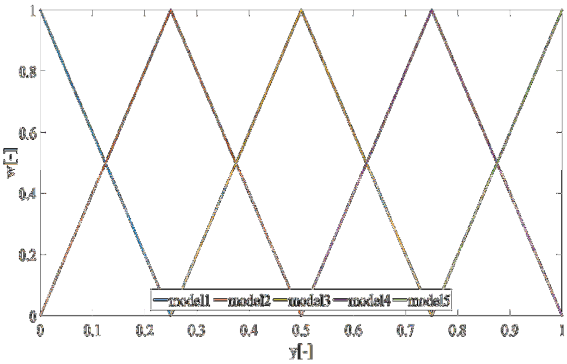

Fig. 2. Final linear modelselection

The vertical axis of the figure 2 represents the weight of the model and the horizontal axis represents the output value.

The results show a comparison between three methods of the cost function minimization: Matlab function quadprog, simple fast gradient method and the presented predictor-corrector method. These methods were also compared by their computation time which were measured using Matlab tic(), toc() functions. The simulations were done with the sampling period T_0 = 5ms, prediction horizon N = 20 steps and the weighting coefficients λ = 1 and δ = 1. The figures 3 and 4 show the output and input signals for simulations with a step change of the reference signal.

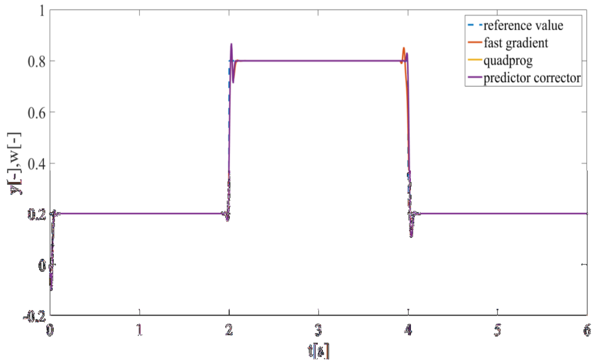

Fig. 3. System output signals

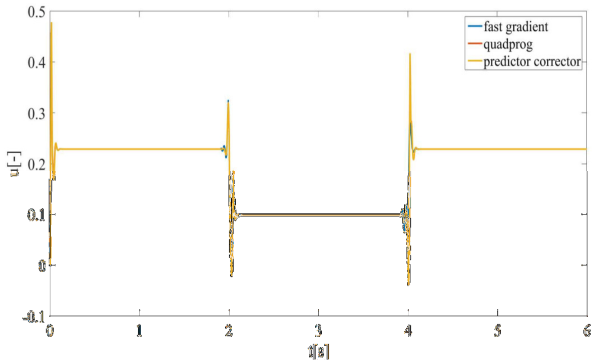

Fig. 4. System input signals

The simulations were also compared by two quadratic criterions for analysis of the control quality. The first criterion, described by equation

$$S_u = \frac{1}{N}\sum_{k=1}^{N}\Delta u^2(k) \tag{31}$$

compares the control increments made in every step and the second criterion, described by equation

$$S_e = \frac{1}{N}\sum_{k=1}^{N}\left[w(k) - y(k)\right]^2 \tag{32}$$

compares a difference between the reference value and the output value.

The Table II. shows results of these criterions of the individual simulations as well as a mean computation time of the one control step.

TABLE II. SIMULATION RESULTS

	Fast gradient	Quadprog	Predictor corrector
S_e [-]	$6.51 \cdot 10^{-4}$	$8.35 \cdot 10^{-4}$	$8.96 \cdot 10^{-4}$
S_u [-]	$1.19 \cdot 10^{-4}$	$1.97 \cdot 10^{-4}$	$2.25 \cdot 10^{-4}$
Computation time [ms]	9.505	12.737	0.687

The figures 5 and 6 show the output and input signals for simulations with a linear change of the reference signal.

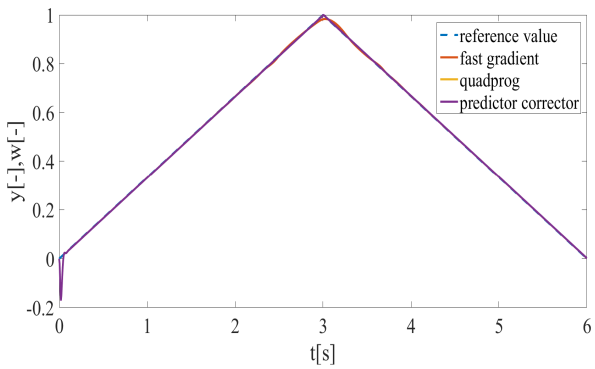

Fig. 5. System output signals

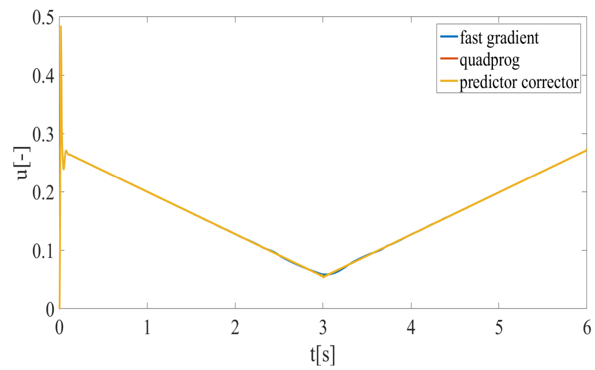

Fig. 6. System input signals

TABLE III. SIMULATION RESULTS

	Fast gradient	Quadprog	Predictor corrector
S_e [-]	$10.3 \cdot 10^{-5}$	$9.52 \cdot 10^{-5}$	$9.59 \cdot 10^{-5}$
S_u [-]	$8.54 \cdot 10^{-5}$	$8.54 \cdot 10^{-5}$	$8.61 \cdot 10^{-5}$
Computation time [ms]	9.004	12.667	1.384

The figures 7 and 8 show the output and input signals for simulations with a sinus change of the reference signal.

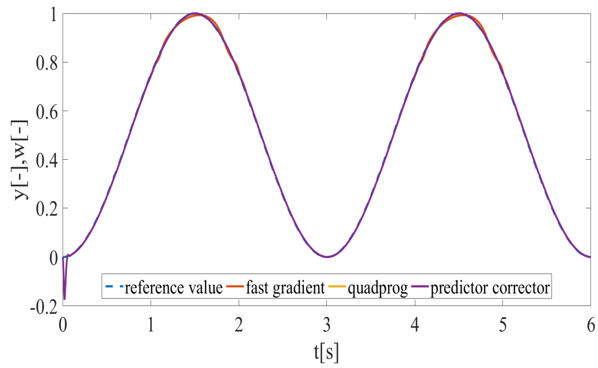

Fig. 7. System output signals

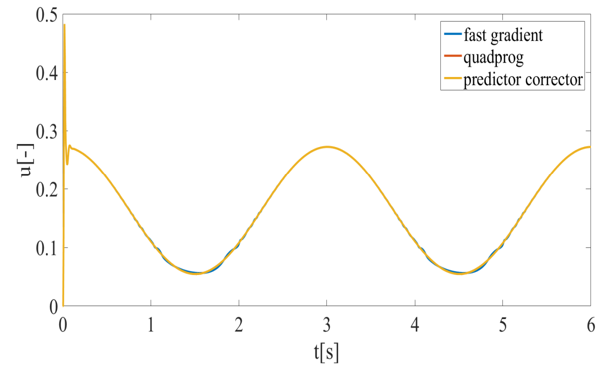

Fig. 8. System input signals

TABLE IV. SIMULATION RESULTS

	Fast gradient	Quadprog	Predictor corrector
S_e [-]	$1.31 \cdot 10^{-4}$	$1.03 \cdot 10^{-4}$	$1.03 \cdot 10^{-4}$
S_u [-]	$8.57 \cdot 10^{-5}$	$8.56 \cdot 10^{-5}$	$8.57 \cdot 10^{-5}$
Computation time [ms]	8.729	12.374	1.610

The figures 3 up to 8 show the simulation of the output and input signals of the magnetic levitation model control. The control of this model is realized by the predictive control method which is based on the cost function minimization. This is an optimization problem and several method can be used to calculate the input signal. This section compares three optimization methods and tries to determine the suitable method for control of the chosen model. This model is relatively fast and the sampling period was chosen as

$T_0 = 5\text{ms}$. The tables of the simulation results compare the quadratic criterions of the control quality and the computation time of the chosen minimization method. The values of the quadratic criterions S_u and S_e for all of the methods in all of the presented simulations indicate that these methods should be applicable and able to follow the reference signal. However, the last and the most important examined parameter, the computation time, points out that it is not true. The fast gradient method and the quadprog function have the computation time of all of the simulations much higher than the sampling period $T_0 = 5\text{ms}$. That means, these methods are not useable for the real application of the magnetic levitation control. Only the presented predictor-corrector method has the computation time lower that the sampling period and so can be used for the real application.

VI. Conclusion

In this paper, the predictive controller based on the state-space CARIMA model was presented. The designed controller was used to simulation of the magnetic levitation model control. This model represents a nonlinear single input single output system with a short sampling period. These systems are very difficult to control due to their complexity and the speed of the output value. The mathematical model of the real magnetic levitation model CE 152 designed by TQ Education and Training Ltd was used as an exemplar system. The aim of this paper was to present a predictive controller capable of control of such system. Since the presented predictive control method works only with a linear models, the chosen nonlinear model linearization process is described. The final linear model is obtained by linear combination of two out of the five linear models depending on the current output value. The calculation of the input signal is done by the minimization of the cost function that minimize the differences between the output and the reference signals and the control signal increments. This minimization can be achieved by using different methods with different computation demands. One of these methods, the predictor-corrector method, is presented in this paper. The results section shows the differences between three minimization methods used to calculation of the control signal. While the presented figures of the simulation may indicate that all of the optimization methods reached almost the same level of the reference signal following, the computation time of these methods is very different. The mean computation time of the fast gradient method and the quadprog function exceeds the sampling period $T_0 = 5\text{ms}$ almost two times in the case of the fast gradient method and more than two times in the case of the quadprog Matlab function. Only the presented predictor-corrector method mean computation time is sufficient for the sampling period $T_0 = 5\text{ms}$.

Acknowledgment

This article was created with support of the Ministry of Education of the Czech Republic under grant IGA reg. n. IGA/FAI/2017/009.

References

[1] V. Bobál, Adaptive and predictive control. vol. 1. Zlin, Tomas Bata University in Zlin, 2008, in Czech.

[2] R. Bars, R. Haber and U. Schmitz. Predictive control in process engineering: From the basics to the applications. Weinhaim: Willey-VCH Verlag, 2011.

[3] F. Gazdoš, P. Dostál, R. Pelikán and V. Bobál. “Polynomial approach to control system design for a magnetic levitation system”. Proceedings of european control conference 2007, Kos, Greece, pp. 4561-4567,

[4] J. A. Rossiter, Model based predictive control: a practical approach, CRC Press, 2003.

[5] K. M. Hangos, J. Bokor and G. Szederkényi. Analysis and control of nonlinear process systems. Springer. London, 2004.

[6] K. M. Moudgalya, Digital control. Chichester: John Wiley, 2007.

[7] J. M. Maciejowski, Predictive control with constraints, Prentice Hall, London, 2002.

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

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

[10] E.F. Camacho, and C. Bordons, Model predictive control, Springer Verlag, London, 2004.

[11] J. Mikleš, and M. Fikar, Process modelling, optimisation and control, Springer-Verlag, Berlin, 2008.

[12] R.R. Bitmead, M. Gevers, V. Hertz, Adaptive optimal control. The thinking man’s GPC, Prentice Hall, Englewood Cliffs, New Jersey, 1990.

[13] L. Wang, Model predictive control system design and implementation using MATLAB, Springer Verlag, London, 2009.

[14] R. Haber, R. Bars and U. Schmitz. Predictive control in process engineering: From the basics to the applications. Weinhaim: Willey-VCH Verlag, 2011.

[15] J. Nocedal and S. Wright. Numerical optimisation second edition. Springer, New York, 2000.

[16] S. Wright. Primal-dual interior point methods. Philadelphia: Society for Industrial and Applied Mathematics, 1997.