Slovak University of Technology in Bratislava
Institute of Information Engineering, Automation, and Mathematics

PROCEEDINGS

17$^{\mathrm{th}}$ International Conference on Process Control 2009

Hotel Baník, Štrbské Pleso, Slovakia, June 9 – 12, 2009

ISBN 978-80-227-3081-5

http://www.kirp.chtf.stuba.sk/pc09

Editors: M. Fikar and M. Kvasnica

Ogonowski, Z.: Fast Linearization Algorithm for Predictive Control, Editors: Fikar, M., Kvasnica, M., In *Proceedings of the 17th International Conference on Process Control '09*, Štrbské Pleso, Slovakia, 314–322, 2009.

Full paper online: `http://www.kirp.chtf.stuba.sk/pc09/data/abstracts/078.html`

FAST LINEARIZATION ALGORITHM FOR PREDICTIVE CONTROL

Zbigniew Ogonowski *

* *Institute of Automatic Control*
Silesian University of Technology, Gliwice, Poland
fax : +48 32 237-10-84, e-mail : zbigniew.ogonowski@polsl.pl

Abstract: The applicability of nonlinear predictive control algorithms is limited by the necessity of on-line solving an optimization problem. Complexity follows from non-linearity of the model and from the extension of the prediction horizon, which result in non-convex constrained optimization problem. To overcome these barriers, different nonlinear optimization schemes have been proposed as e.g. in recent Zavada and Biegler (2009) where nonlinear programming has been extended through a simple reformulation of the nonlinear predictive control problem into advanced-step controller. However still, the heuristic simplification of nonlinear problem namely successive linearization or linearization along predicted trajectory is very competitive. The most important advantage is the constant processing time. The only problem arises if short sampling period is used and processing time has to be reduced. In this case numerical complexity follows from time-consuming linearization procedure. The paper proposes a new and fast linearization algorithm which uses identification procedure. It is assumed that non-linear noise-free model is given. Data for identification are created with the model impulse response. This allows for more flexible linearization where the vicinity of the operating point is discussed rather then point-sensitive linearization provided by the standard procedures. There are two key-simplifications which make the algorithm fast. The generic non-linear model is usually given in continuous time. The lack of the nonlinear discrete counterpart causes that the linearization has to be done first and discretization afterwards. The method proposed in the paper coupled these two operations. The second simplification comes from the Toeplitz-type of the matrix being inverted in the identification Least-Mean-Square algorithm. It is shown in the paper that the number of matrix elements is reduced usually 4-5 times. Then fast algorithms can be used to invert the final general-Toeplitz matrix (e.g. Martinsson-Rokhlin-Tygert, 2005). The efficiency of the resulting algorithm is illustrated in the paper by comparison of the computation-time with standard linearization procedure, which bases on perturbation algorithm and discretizes obtained continuous-time linear model using modified scaling and squaring method.

Keywords: suboptimal predictive control, successive linearization, discretization, identification.

1. INTRODUCTION

The general results concerning stability, performance and robustness of the non-linear predictive control have been developed in the late nineties (see e.g. Algöwer and Zheng (2000)). It became clear that this control concept constituted one of the very few general synthesis method of non-linear control systems. The concept, however, gives rise to the collection of different control algorithms which applicability depends on number of terms. The most crucial is the numerical complexity versus limited resources of a digital environment. Stability of the closed-loop predictive control systems can serve as an illustration. It has been shown in Bitmead et al. (1990) that the closed-loop stability of receding horizon control system in the linear case is not guaranteed for generic cost function thus modifications of the cost function are necessary as was firstly proposed in Kwon and Peason (1978), by means of terminal zero-state constraint. This result has been generalized onto nonlinear case in e.g. Mayne and Michalska (1990). Number of proposals of how to construct the cost function has been presented and stability has been proved; for example terminal penalty has been introduced in Rawlings and Muske (1993), or combination of a terminal penalty with a terminal inequality constraints, Chen and Algöwer (1998). Proofs of the stability base on the monotonicity of the optimal value of the cost function, thus the solution of the optimization problem has to be known. In other words, to keep stability the control algorithm has to be optimal i.e. in every sampling period the control value has to follow from the solution of the optimization problem which minimizes cost function with state and control constraints.

Optimization problem which is formulated to design a general non-linear predictive controller constitutes considerably complex task, due to non-convexity, large number of decision variables and constraints. (Keerthi and Gilbert (1988)) discusses examples of infinite-dimensional problems; number of decision variables in optimization problem is infinite in this case. Obviously, only approximative solution can be applied. Approximate cost function is obtained by truncation of infinite series. This leads to hard optimization problem in a large dimensional space. Another way that allows for simpler calculations and still prevents stability is sub-optimal approach, firstly proposed in Scokaert et al. (1999). More recent interesting modification has been proposed in Limon et al. (2006). Suboptimal approach assumes control signal to be in a constrained region which also contains the optimal solution. The crucial problem is, however, that in order to determine that region heavy computations has to be curried out as well. Still the question of applicability of the resulting predictive control algorithm retains.

Suboptimal predictive control arises whenever the solution of the optimization problem is found in an approximative way. It is happen if optimization is stopped before the solution is found. The same effect can also be achieved if approximative model is used. This leads to approximative prediction of the output and even optimal solution of the optimization problem is found the control algorithm is still suboptimal. Well-known approach following that idea is successive linearization (Henson (1998)). As the suboptimal approach, successive linearization is the simplest. Nonlinear control system inherits all advantages of linear predictive control in this case. In every sampling period a linearization of the nonlinear model is computed, and then, linear predictive controller is designed to determine the control signal. The idea has been applied in number of real processes, as chemical batch processes (Cueli and Bordons (2008), Tiagounov and Weiland (2003)); vehicle and transportation problems (Falcone et al. (2007), Raffo et al. (2009)) to mention only a few. However, the authors emphasize the necessity of fast microprocessor environment to be used. Still there are a lot of examples of fast systems to be controlled where sampling period is to short to make the numerical control algorithm realizable. Representative examples can be found in mechatronics (e.g. Ogonowski and Plaza (2006)).

Another approach is linearization along predicted state trajectory (e.g. van der Veen et al. (1999)). Here linearization should be computed many times during every sampling period according to the length of the prediction horizon.

The conclusion following from the above resume is that any improvement which makes the predictive control algorithm faster is of the great value. The paper addresses an one of the most important element of predictive control algorithms – the linearization task. It is assumed that nonlinear model of the plant is given. Numerical standard linearization (sec. 2) uses sensitivity response. After obtaining a continuous linear model a discrete counterpart is determined usually by scaling and squaring method. The proposed method (sec. 3) uses identification procedure. The model is pick-up using Least-Mean-Square (LMS) method with data obtained from the impulse response of the nonlinear model. Reduction of the proposed algorithm complexity follows from special feature of the LMS measurement matrix entries. Sec. 4 presents numerical examples illustrating the efficiency of the method.

2. STANDARD LINEARIZATION AND DISCRETIZATION METHODS

Assume the following nonlinear continuous in time model of the plant

$$\begin{cases} \dot{\mathbf{x}}(t) = f(\mathbf{x}(t), u(t), t), \\ y(t) = g(\mathbf{x}(t), u(t), t) \end{cases} \tag{1}$$

where $\mathbf{x} \in \Re^n$ represents state of the system, $u \in \Re$ is the control signal and $y \in \Re$ is the output (measured) signal. Extension of the results presented in this paper onto multi-input and multi-output case is straightforward. Function f and g are assumed to be smooth, however, no special demands concerning these function are necessary to be stated because numerical algorithms will be discussed in this paper only.

Standard linearization concerns equilibrium point

$$\begin{cases} f(\mathbf{x}_e, u_e) = 0 \\ y_e(t) = g(\mathbf{x}_e, u_e) \end{cases} \tag{2}$$

where subscript e points a steady state. The notion 'equilibrium point', is misleading if successive linearization is concerned because the point around which the linearization is computed follows form dynamic behavior of the system. It then happens, that actual state and input do not fit the steady state. Formally, one can determine the linear model for any point. There are, however, two consequences: first, it may happen that derivatives of f and g do not exist and, second, accuracy of linear model can be weak. These problems is discussed in sec. 3.

To determine linear model in numerical way usually perturbation algorithm is used. Assume, that: $t = t_0$, $\mathbf{x}(t_0) = \mathbf{x}_0$, $u(t_0) = u_0$, and $y(t_0) = g(\mathbf{x}_0, u_0, t_0) = y_0$ represents the local 'operating point'. By subtracting the operating point values from the states, inputs, and outputs one defines a set of variables centered around this point:

$$\begin{cases} \delta\mathbf{x} = \mathbf{x}(t) - \mathbf{x}_0, \\ \delta u(t) = u(t)(t) - u_0, \\ \delta y(t) = y(t)(t) - y_0. \end{cases} \tag{3}$$

The linearized model is valid for these new variables if they are small, i.e. if deviation from the 'operating point' is small. The linearized model usually takes The following form

$$\begin{cases} \delta\dot{\mathbf{x}}(t) = A\delta\mathbf{x}(t) + B\delta u(t), \\ \delta y(t) = C\delta\mathbf{x}(t) + D\delta u(t). \end{cases} \tag{4}$$

The state-space matrices A, B, C, and D represent Jacobians of the system. Respect transfer function are more convenient for the prediction and can be directly computed using model (4). To compute the matrices of (4), the states and inputs are perturbed, and the response of the system to this perturbation is measured by computing $\delta\dot{\mathbf{x}}(t)$ and δy. The computations can be expressed as follows:

$$a(k) = \frac{\dot{\mathbf{x}}|_{\mathbf{x}_k^p} - \dot{\mathbf{x}}_0}{\mathbf{x}_i^p - \mathbf{x}_0}, \tag{5}$$

$$B = \frac{\dot{\mathbf{x}}|_{u^p} - \dot{\mathbf{x}}_0}{u^p - u_0}, \tag{6}$$

$$c(k) = \frac{y|_{\mathbf{x}_k^p} - y_0}{\mathbf{x}_k^p - \mathbf{x}_0}, \tag{7}$$

$$D = \frac{y|_{u^p} - y_0}{u^p - u_0}, \tag{8}$$

where $a(k)$ is k-th column of the matrix A, $c(k)$ is the k-th element of the (row vector) matrix C and:

$\mathbf{x}_k^p$ - the state vector whose k-th component is perturbed from the "operating point" value,
u^p - the perturbed input,
$\dot{\mathbf{x}}|_{\mathbf{x}_k^p}$ - the value of $\dot{\mathbf{x}}$ at $\mathbf{x}_k^p$ and u_0,
$\dot{\mathbf{x}}|_{u^p}$ - the value of $\dot{\mathbf{x}}$ at u^p and $\mathbf{x}_0$,
$y|_{\mathbf{x}_k^p}$ - the value of y at $\mathbf{x}_k^p$ and u_0,
$y|_{u^p}$ - the value of y at u^p and $\mathbf{x}_0$.

and division is made element-by-element. A perturbed value has to be a very small distanced from the operating point value. The default value is usually chosen as $10^{-8}|\mathbf{x}| \div 10^{-5}|\mathbf{x}|$.

Discretization of the linear approximation (4) bases on general solution of (4) and assumption concerning extrapolating element. Usually zero-order-hold extrapolation is applied and resulting problem concerns only calculation of fundamental matrix

$$\Phi = exp(AT_s) \tag{9}$$

where T_s represents the sampling period. There are number of different methods to calculate matrix (9) – see e.g. Moler and van Loan (2003). One of the very basic is scaling and squaring method developed by Lawson (1967). The method calculates Padé approximation as follows

$$exp(AT_s) \approx \left(r_{km} \left(exp\left(\frac{AT_p}{\sigma} \right) \right) \right)^{\sigma} \tag{10}$$

where scaling parameter

$$\sigma = 2^s, \quad s \text{ is a natural number} \tag{11}$$

and,

$$r_{km}(X) = \frac{n_{km}(X)}{d_{km}(X)} \tag{12}$$

where

$$n_{km}(X) = \sum_{j=0}^{k} \frac{(k+m-j)!k!}{(k+m)!(k-j)!j!}(X)^j, \tag{13}$$

$$d_{km}(X) = \sum_{j=0}^{m} \frac{(k+m-j)!m!}{(k+m)!(m-j)!j!}(-X)^j. \tag{14}$$

Scaling and squaring algorithm has been modified in different publications. The most recent Higham (2005) uses trigonometrical properties of the fundamental matrix, Koikari (2007) examines accuracy of the approximation (10).

Both methods, linearization trough perturbation and discretization trough scaling and squaring are used in many commercial computer programs as for example Matlab. The methods are still under research. The most important problem is complexity, which influences processing time and accuracy. Koikari (2007) has shown that the Matlab's discretization function causes to high error because of to low order of Padé approximation. He proposed yet another modification of scaling and squaring method to improve accuracy instead of increasing approximation orders.

3. IDENTIFICATION METHOD

Linearization and discretization as separate algorithms need complex operations that makes the whole process of input-output model determination unapplicable in many real-time control systems. There are also additional problems. Eventual discontinuities of the function f and g cause the sensitivity of the linearization to perturbation margins. The choice of the margin is then crucial and may change the result of linearization in a large range. These problems are detailed in Parson and Glass (2004).

Specific of application of the successive linearization or linearization along predicted output trajectory in predictive control adds yet another difficulty. The linearization considers initial conditions defined by the actual state of the system which are interpreted as the 'operating point'. The procedure calculates the approximation (4) and the respect transfer functions. However, one can expect that dynamic behavior of the predictive control system would excite the controlled plant much greater then the perturbation margin is. The range of this excitation can be easily evaluated. Thus much more valuable would be the linear approximation that takes into account this range of excitation or at least allows for controlled departure from 'operating point' then the perturbation margin does.

The identification method proposed below combines both linearization and discretization. Impulse response of the nonlinear plant provides data for Least-Mean-Square method of identification of the linear input-output discrete-time model. Amplitude of the excitation can be chosen according to the concrete demands.

Assume the following transfer function to be identified

$$K(z^{-1}) = \frac{B(z^{-1})}{A(z^{-1})} = \sum_{j=0}^{\infty} g_j z^{-j} \qquad (15)$$

where z^{-1} is one sampling period back-shift operator, polynomial A of the order nA is monic and polynomial B of the order nB can have zero leading coefficients to include discrete delay-time. It is easy to verify that

$$g_j = \begin{cases} b_j - \sum\limits_{s=1}^{\max\{j,nA\}} a_j g_{j-s} & j = 0, 1, \ldots, nB \\ -\sum\limits_{s=1}^{nA} a_j g_{j-s} & j > nB \end{cases} \qquad (16)$$

where b_j and a_j are coefficients of the polynomials B and A respectively. It follows from (16) that

$$\begin{bmatrix} g_0 \\ g_1 \\ \vdots \\ g_{nA+nB} \end{bmatrix} = [I, G] \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_{nB} \\ a_1 \\ a_2 \\ \vdots \\ a_{nA} \end{bmatrix} \qquad (17)$$

where I is $nB + nA + 1 \times nB + 1$ matrix of the following entries:

$$I = \begin{bmatrix} 1 & 0 & \ldots & 0 \\ 0 & 1 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & 1 \\ 0 & 0 & \ldots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \ldots & 0 \end{bmatrix} \qquad (18)$$

and,

$$G = \begin{bmatrix} G_1 \\ G_2 \end{bmatrix} \qquad (19)$$

$$G_1 = \begin{bmatrix} 0 & 0 & \ldots & 0 \\ -g_0 & 0 & \ldots & 0 \\ -g_1 & -g_0 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ -g_{nB-1} & -g_{nB-2} & \cdots & g_0 \end{bmatrix} \qquad (20)$$

$$G_2 = \begin{bmatrix} -g_{nB} & -g_{nB-1} & \cdots & -g_1 \\ -g_{nB+1} & -g_{nB} & \cdots & -g_2 \\ \vdots & \vdots & \ddots & \vdots \\ -g_{nB+nA-1} & -g_{nB+nA-2} & \cdots & -g_{nA} \end{bmatrix} \qquad (21)$$

Note, that negative indices of matrix G_1 and G_2 entries mean that these entries not longer appear in the matrices, e.g. if $nB = 0$ matrix G_1 reduces to row zero-vector of the length nA.

The set of linear equation (17) is square and creates the minimal-size identification problem. Such problem can be stated because noise-free case is discussed. There is obvious possibility to enlarge the set (17) with succeeding elements of the impulse response to increase the accuracy of the method. Not till then Least-Mean-Square method can be used. If minimal-size identification task has to be solved then the inverse (instead of pseudo-inverse) of the matrix $[I, G]$ has to be calculated.

The most important coincidence should be now observed. Thanks to the special construction of the set (17) the only matrix G_2 has to be inverted. Knowing the solution for $a_1, a_2, \ldots, a_{nA}$ the coefficients b_j of the polynomial B can be obtained straightforward from the set (17). It is well known that discrete counterpart of the linear continuous model has equal orders $nA = nB$ or $nB = nA + 1$ (in the delay-free case or continuous delay-time is multiple sampling period). The complexity of the problem is then about twice reduced, or, number of entries of the matrix to be inverted is reduced 4-5 times.

The second coincidence is the special construction of the matrix G_2 namely general Toeplitz type. There are well known special algorithms that invert Toeplitz matrix much faster then the standard algorithms do. Methods for computing inversion of G_2 in $O(N^2)$ operations have been known since the 1960s, see for example, Trench (1964). If G_2 is positive-definite algorithms in $O(N \log 2N)$ operations are known, Ammar (1996). These algorithms base on algebraic properties of the Toeplitz matrix structure. Most recent work (Martinsson et al. (2005)) proposes four length-N FFTs plus an $O(N)$- operation algorithm.

4. ILLUSTRATIVE EXAMPLE

An example of fast sampled system is magnetic bearings for high-speed rotating machinery. One of the most challenging control problems arises in this unit. Mechanical construction of the magnetic suspension have reached its maturity and further improvement of the performance needs application of modern control algorithms rather then redesign of the construction. Usually design of the control system bases on the frequency analysis. The control system uses simple lead-lag compensators, which are almost always followed by additional cascaded biquad filters to shape the frequency response and to assure stability of all system bending modes. The stiffness of the system becomes however too small even if the control system is carefully tuned and stability is assured. Thus more sophisticated control system is needed. An example of such algorithm is proposed in Ogonowski and Plaza (2006). The algorithm has been tested on MCB500 magnetic suspension system (Fig.1) which consists of two active radial magnetic bearings supporting a steel shaft. The shaft can rotate freely due to being actively positioned in the radial directions at the shaft's ends and passively centered in the axial direction. Position of the shaft's ends constitutes 4 degrees of freedom. The position is measured in 4 axes using hallotrons. The system includes four linear current amplifiers and four linear lead-lag compensators, which control the radial bearing axes (PD controllers). MBC500 has been enhanced with pneumatic push-pull driver and angular velocity control unit. Fig. 2 presents the schematic of the unit. First-principle model of the MBC500 system is described in Morse et al. (1996) (the model and the parameters of the unit are briefly recalled in the Appendix).

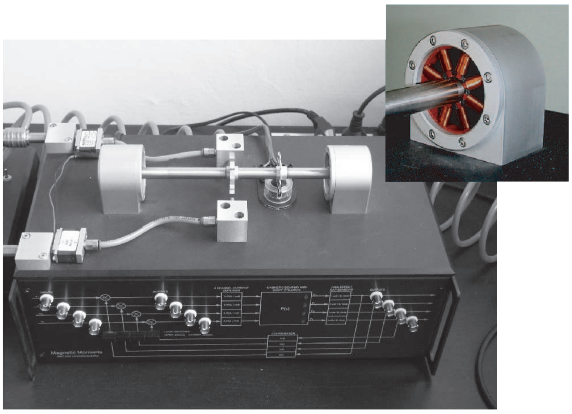

Fig. 1. Laboratory set-up MBC500.

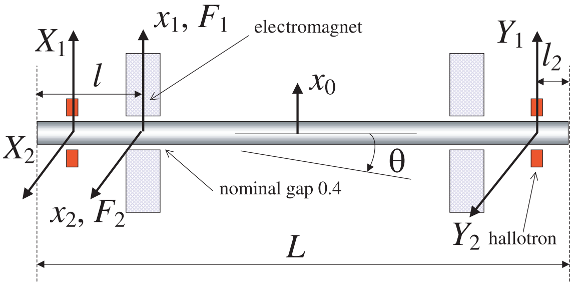

Fig. 2. Scheme of the MBC500 system.

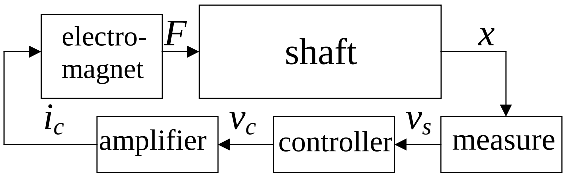

Fig. 3. Structure of the control system.

Fig. 3 presents structure of the control system. Front panel (Fig. 1) allows for replacing the built-in controller with the outer one (e.g. weighted minimumvariance realized by PC equipped with DSP card, as is presented in Ogonowski and Plaza (2006)). Significantly better vibration control can be achieved with internal model structure where the plant is pre-stabilized with built-in controllers (Ogonowski and Plaza (2007)).

To check the algorithm presented in this paper, it is proposed to find the linear approximation as the transfer function between control signal ν_c and displacement x_1. This constitutes single-input and single-output model, which is necessary to design internal model control as in Ogonowski and Plaza (2007). Adding a perturbation signal to ν_s, one can observe the response, and pick-up the linear model. This is realized numerically using nonlinear model (see Appendix) by both of the approach described above (sec. 2 and 3). The system itself responds as is presented in Fig. 4 (step input is equal 50% of ν_s range).

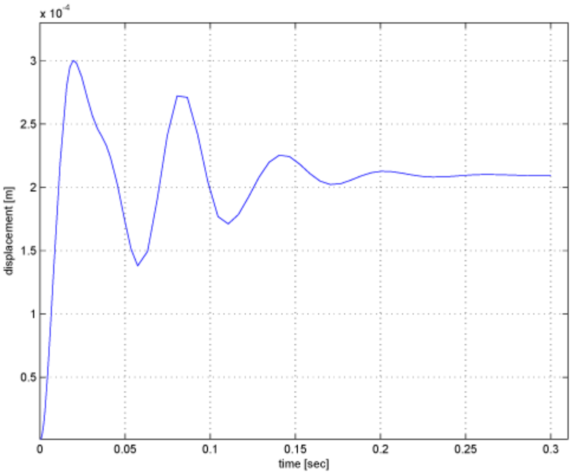

Fig. 4. System response.

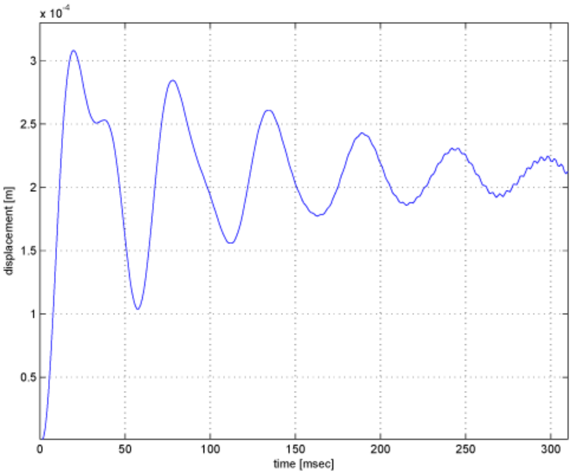

Fig. 5. Model response (standard linearization and discretization).

The system has got 14 states, thus linear transfer function has got the polynomials of the order $nB = nA = 14$, or $nB = 13$ and one sample time delay because $b_0 \equiv 0$ in this case. Using standard linearization with perturbation around zero in the range 10^{-8} one can obtain the transfer function with coefficients presented in Tab. 1.

Fig. 5 shows the model response to 50% input step. It is clear that nonlinearity of the plant causees significant difference when compare with Fig. 4.

Identification method applied to this example gives results as in Tab. 2 (impulse response has been calculated from step response to 10^{-4} input.)

Fig. 6 presents the response of the model obtained by the identification procedure. To be convinced that the obtained result is much more accurate, one can compare this result with the previous responses.

The second advantage of the proposed procedure is much less complexity. Comparison of the processing time shows about 8 times faster identification procedure (with classical Lawson algorithm for Toeplitz matrix inversion) when compare with standard linearization and dyscretization method.

j	b_j	a_j
0	0.00000000000000	1.00000000000000
1	0.02108903210439	-5.87597247692839
2	-0.05260212853356	15.80447629388848
3	0.03547767263475	-26.48106954418814
4	0.02478334904841	31.68055210048474
5	-0.08583591657239	-28.39355650062849
6	0.12436310069233	18.40281830654936
7	-0.09781171676396	-7.78405051479949
8	0.03061405579883	1.80088835658447
9	0.00064046070909	-0.15942172493736
10	-0.00029644902319	0.00560974342229
11	0.00001156921562	-0.00006798079989
12	-0.00000008196529	0.00000026232927
13	-0.00000000000000	-0.00000000000000
14	0.00000000000000	-0.00000000000000

Tab. 1 Results of standard linearization and discretization.

j	b_j	a_j
0	0	1.00000000000000
1	-0.00000148218697	-0.89089020582183
2	0.00000770973566	-0.47727814344222
3	0.00000706155007	-0.06927136352096
4	0.00000195047562	0.26033330278855
5	-0.00000183614572	0.34679163055312
6	-0.00000384374566	-0.12414894868296
7	0.00000000263544	0.00340902828028
8	0.00000000002342	0.00007947044490
9	-0.00000000000067	-0.00000563363629
10	0.00000000000000	-0.00023828029923
11	0.00000000000000	0.00003536357282
12	0.00000000000000	0.00000736335536
13	0.00000000000000	0.00000000006373
14	0.00000000000000	-0.00000000000000

Tab. 2 Results of identification method.

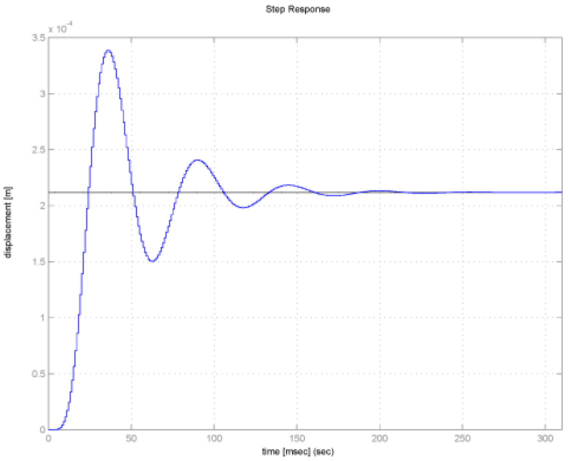

Fig. 6. Model response (proposed identification method).

5. CONCLUSION

Identification method instead of standard linearization and discretization method makes the process of linear-discrete model determination much faster and allows for departure from ”operating point” to be 'controlled'. This leads to improvement of the accuracy of the linearization in the context of predictive control policy. One can design an iterative predictive algorithm, where the increment of the control signal to be applied in the next sampling period would be the step excitation to establish impulse response of the nonlinear model. Thus the procedure of the linearization can be repeated. Good properties of such algorithm could be expected especially if hard nonlinearities or non-symmetrical response of the plant is concerned.

An interesting extension of the identification method can be formulated if more data are included and non-minimal-size identification problem is studied. Pseudo-inverse of the matrix $[I, G]$ needs to be calculated. Further research will be devoted to these problems especially the issues of matrix inversion. The key-issue is that the problem loses Toeplitz-type, however, the matrix to be inverted becomes symmetrical and positive-definite and another fast procedures can be applied.

APPENDIX

According to the notation showed in Figs. 2 and 3 the following analytical (first principle) model is derived [1]. The measurement element is nonlinear according to the following equation

$$\nu_s = 5\left[\frac{\mathrm{V}}{\mathrm{mm}}\right] X_1 + 24\left[\frac{\mathrm{V}}{\mathrm{mm}^3}\right] X_1^3 \qquad (22)$$

where ν_s is the sensor output and X_1 is the displacement in the first X-axis. Built-in compensators are designed as lead-lag according to the following relation

$$\nu_c = \frac{1.45(1 + 0.9 \cdot 10^{-3}}{(1 + 3.3 \cdot 10^{-4}s)(1 + 2.2 \cdot 10^{-5})s}\nu_s \qquad (23)$$

The compensator output is then transformed into the current:

$$i_c = \frac{0.25}{1 + 2.2 \cdot 10^{-4}s}\left[\frac{\mathrm{A}}{\mathrm{V}}\right]\nu_c \qquad (24)$$

and then again in the nonlinear manner the signal is transformed into the force:

$$F_1 = k\frac{(i_c + 0.5)^2}{(x_1 - 0.0004)^2} - k\frac{(i_c - 0.5)^2}{(x_1 - 0.0004)^2} \qquad (25)$$

where displacement x concerns the point of electromagnets rather then the point of Hall-sensor as in the X case. 0.5 [A] current appearing in equation (25) refers to bearing bias upon which a control signal is superimposed.

Mechanical part of the system has to be considered as two-input and two output. It follows form the notation shown in Fig.2 that:

$$\begin{aligned} x_1 &= x_0 - \left(\frac{L}{2} - l\right) sin\theta \\ x_2 &= x_0 - \left(\frac{L}{2} - l\right) sin\theta \qquad (26) \\ X_1 &= x_0 - \left(\frac{L}{2} - l_2\right) sin\theta \\ X_2 &= x_0 - \left(\frac{L}{2} - l_2\right) sin\theta \end{aligned}$$

$$(27)$$

It follows from the force balance that

$$\sum \overrightarrow{F} = m\overrightarrow{a} \qquad (28)$$

where $\sum \overrightarrow{F}$ is the summation of all external forces applied to the system, m is the rotor mass, and $\overrightarrow{a}$ is the acceleration of the center of gravity of the system. The moment balance

$$\sum \overrightarrow{M} = I\overrightarrow{\alpha} \qquad (29)$$

where $\sum \overrightarrow{M}$ is the summation of all external moments applied to the system, I is the rotational moment of inertia of the system about the axis trough the center of gravity and in the direction of rotation, and $\overrightarrow{\alpha}$ is the angular acceleration of the system. The equation of the motion is then as follows:

$$\begin{aligned}\sum F &= m\ddot{x}_0 = F_1 + F_2 \\ \sum M &= I_0\ddot{\theta} = F_2\left(\frac{L}{2} - l\right)cos\theta - \\ &\quad F_1\left(\frac{L}{2} - l\right)cos\theta\end{aligned} \tag{30}$$

With F_1 and F_2 as input and X_1 and X_2 as output variables the linear state-space model of the system can be formulated as follows

$$\begin{bmatrix}\dot{x}_0\\ \ddot{x}_0\\ \dot{\theta}\\ \ddot{\theta}\end{bmatrix} = \begin{bmatrix}0&1&0&0\\0&0&0&0\\0&0&0&1\\0&0&0&0\end{bmatrix}\begin{bmatrix}x_0\\ \dot{x}_0\\ \theta\\ \dot{\theta}\end{bmatrix} + \begin{bmatrix}0 & 0\\ \frac{1}{m} & \frac{1}{m}\\ 0 & 0\\ -\frac{1}{I_0}\left(\frac{L}{2}-l\right) & \frac{1}{I_0}\left(\frac{L}{2}-l\right)\end{bmatrix}\begin{bmatrix}F_1\\ F_2\end{bmatrix} \tag{31}$$

$$\begin{bmatrix}X_1\\ X_2\end{bmatrix} = \begin{bmatrix}1 & 0 & -\left(\frac{L}{2}-l_2\right) & 0\\ 1 & 0 & \left(\frac{L}{2}-l_2\right) & 0\end{bmatrix}\begin{bmatrix}x_0\\ \dot{x}_0\\ \theta\\ \dot{\theta}\end{bmatrix} \tag{32}$$

Parameters of MBC500 bearing system are as follows:

- Total length of the shaft 26.9 cm.
- Distance from each bearing to the end of the shaft 2.4 cm.
- Distance from each Hall-effect sensor to the end of the shaft 2.8mm.
- Shaft's moment of inertia with respect to rotation about an axis in the y direction $1.588 \cdot 10^{-3}$ kg m^2.
- Mass of the rotor 0.2629 kg.

ACKNOWLEDGMENTS

The work has been supported by the Polish Ministry of Science and Higher Education, grant of Institute of Automatic Control, Silesian University of Technology. This support is very gratefully acknowledged.

References

F. Allgöwer, A.Zheng (Eds.) *Nonlinear Model Predictive Control.* Birkhäuser, 2000.

D. Mayne, H. Michalska. Receding horizon control of nonlinear systems. *IEEE Trans. on Automatic Control*, volume 35, pages 814–824, 1990.

R. Bitmead, M. Gevers and V. Wertz. *Adaptive Optimal Control: The Thinking Man's GPC.* Prentice Hall, 1990.

W. Kwon, and A. Pearson. On Feedback Stabilization of Time-Varying Discrete-Linear Systems. *IEEE Trans. on Automatic Control*, volume 23, pages 479–481, 1978.

J. Rawlings, K. Muske The stability of constrained receding horizon control. *IEEE Trans. on Automatic Control*, volume 38, pages 1512–1516, 1993.

H. Chen, F. Algöwer. A quasi-infinite horizon nonlinear predictive control scheme with guaranteed stability. *Automatica*, volume 34, no 10, pages 1205–1217, 1998.

S. Keerthi, E. Gilbert. Optimal, infinite-horizon feedback laws for a general class of constrained discrete-time systems. *J. Optimiz. Th. Appl.*, volume 57, pages 265–293, 1988.

P.O.M. Scokaert, D.Q. Mayne, J.B. Rawlings. Suboptimal model predictive control (feasibility implies stability. *IEEE Trans. on Automatic Control*, volume 44, pages 648–654, 1999.

D. Limon, T. Alamos, F. Salas, E.F. Camacho. On the stability of constrined MPC without terminal constraints. *IEEE Trans. on Automatic Control*, volume 51, pages 832–836, 2006.

M.A. Henson. Nonlinear model predictive control: current status and future directions. *Computes and Chemical Engineering*, volume 23, pages 187–202, 1998.

M.A. Henson. Iterative nonlinear model predictive control. Stability, robustness and applications. *Control Engineering Practice*, volume 16, pages 1023–1034, 2008.

A.A. Tiagounov, S. Weiland. Model Predictive Control Algorithm for Nonlinear Chemical Processes. *Proc. Int. Conf. Physics and Control*, Petersburg, Russia, 2003.

P. Falcone, F. Borrelli, J. Asgari, H.E. Tseng, D. Hrovat. Predictive Active Steering Control for Autonomous Vehicle Systems. *IEEE Trans. on Control Systems Technology*, volume 15, pages 566–580, 2007.

G.V. Raffo, G.K. Gomes, J.E. Normey-Rico, C.R. Kelber, L.B. Becker. A Predictive Controller for Autonomous Vehicle Path Tracking. *IEEE Trans. on Intelligent Transportation Systems*, in press, 2009.

Z. Ogonowski, K. Plaza. Adaptive vibrations control of a turbine with magnetic bearings. *Proc. Int. Conf. on Noise and Vibration Engineering, ISMA 2006*, Leuven, Belgium, 2006.

Z. Ogonowski, K. Plaza. Internal model control of a magnetic levitation system. *Proc. of the AI-Meth Symposium on Artificial Intelligence Methods*, Gliwice, Poland, pages 155-162, 2007.

P.J. van der Veen, R Babuska, H.B. Verbruggen. Comparison of nonlinear predictive control methods for a waste-water treatment benchmark. *Proc. 5th European Control Conference, ECC'99*, Karlsruhe, Germany, 1999.

L. Parson, J. Glass. Matlab Digest - July 2004. http://www.mathworks.com, The MathWorks, 2004.

J.D. Lawson Generalized Runge-Kutta processes for stable systems with large Lipschitz constants. *SIAM J. Num. Anal.*, volume 4, pages 372-380, 1967.

N.J. Higham. The scaling and squaring method for the matrix exponential revisited. *SIAM J. Matrix Anal. Appl.*, volume 26, 2005.

N.J. Higham. An error analysis of the modified scaling and squaring method. *Comp. and Math. with Appl.*, volume 53, 2007.

W.R. Trench. An algorithm for the inversion of finite Toeplitz matrices. *J. Soc. lndust. AppL Math.*, volume 12, pages 515-522, 1964.

G. Ammar. Classical foundations of algorithms for solving positive definite Toeplitz equations. *Calcolo*, volume 33, pages 99-113, 1996.

P.G. Martinsson, V. Rokhlin, M. Tygert. A fast algorithm for the inversion of general Toeplitz matrices. *Comp. and Math. with Appl.*, volume 50, 2005.

N. Morse, R. Smith, B. Paden. Magnetic bearing Lab# 1: Analytical modeling of a magnetic bearing system. *Comp. and Math. with Appl.*, Magnetic Moments LLC, Goleta, USA, 1996.

V.M. Zavala, L.T. Biegler The advanced-step NMPC controller: Optimality, stability and robustness. *Automatica*, volume 45, issue 1, pages 86-93, 2009.

C. Moler, C. van Loan Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later. *SIAM Review*, volume 45, issue 1, pages 3-49, 2003.