Continuous-Time and Discrete Multivariable 2DOF Controllers

Marek Kubalcik, Vladimír Bobal

Abstract— The paper presents a design and implementation of a 2DOF (two degree of freedom) multivariable controller. The controller was designed in both discrete and continuous-time versions. The control algorithm is based on polynomial theory and pole – placement. The controller integrates an on – line identification of an ARX model of a controlled system and a control synthesis on the basis of the identified parameters. The model parameters are recursively estimated using the recursive least squares method. In case of the continuous-time control loop derivatives of the input and output variables of the continuous – time systems can not be directly measured. Therefore differential filters and filtered variables are established to substitute primary variables. The filtered variables are then used in the recursive identification procedure.

Keywords— multivariable control, control algorithms, adaptive control, polynomial methods, pole assignment, recursive identification.

I. INTRODUCTION

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. There are many different methods of controlling MIMO (multi input – multi output) systems [1]. Several of these use decentralized PID controllers [2], others apply single input-single-output (SISO) methods extended to cover multiple inputs [3]. The classical approach to the control of multi-input–multi-output (MIMO) systems is based on the design of a matrix controller to control all system outputs at one time. The basic advantage of this approach is its ability to achieve optimal control performance because the controller can use all the available information about the controlled system. Controllers are based on various approaches and various mathematical models of controlled processes. A standard technique for MIMO control systems uses polynomial methods [4], [5], [6], [7], [8] and is also used in this paper. Controller synthesis is reduced to the solution of linear Diophantine equations [9].

Marek Kubalčík is with the Tomas Bata University in Zlín, Faculty of Applied Informatics, Nám. T. G. Masaryka 5555, 760 05 Zlín (corresponding author to provide phone: +420 57-603-5198; e-mail: kubalcik@ fai.utb.cz).

Vladimír Bobál is with the Tomas Bata University in Zlín, Faculty of Applied Informatics, Nám. T. G. Masaryka 5555, 760 05 Zlín (e-mail: bobal@ fai.utb.cz).

One controller, which enables control of TITO (two input-two output) systems, is presented. The proposed control algorithm is based on the 2DOF (two degree of freedom) configuration [10]. The controller was realized both in discrete and continuous-time versions. Both versions of the controller were realized both with fixed parameters and as self-tuning controllers [11], [12] with recursive identification of a model of the controlled system. The recursive least squares method is used in the identification part.

In case of the continuous-time control loop input and output derivatives of a system can not be directly measured, the differential filters and filtered variables are established to substitute primary variables. This approach is described in detail in [16], [17], [18]. The filtered variables are then used in the recursive identification procedure, where the classical recursive least squares method is used to identify the parameters. This approach enables fast sampling. The value of the sampling period is then dependent only on capabilities of used hardware and software. The used software must enable realization of filters by differential equations.

II. MATHEMATICAL MODEL OF THE CONTROLLED PROCESS

A general transfer matrix of a two-input–two-output system with significant cross-coupling between the control loops is expressed as (for continuous-time systems $q = s$ as the derivative operator and for discrete systems $q = z^{-1}$ as the delay operator)

$$\boldsymbol{G}(q)=\begin{bmatrix} G_{11}(q) & G_{12}(q) \\ G_{21}(q) & G_{22}(q) \end{bmatrix} \tag{1}$$

$$\boldsymbol{Y}(q)=\boldsymbol{G}(q)\boldsymbol{U}(q) \tag{2}$$

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

$$\boldsymbol{Y}(q)=\left[y_1(q), y_2(q)\right]^T \ \boldsymbol{U}(q)=\left[u_1(q), u_2(q)\right]^T \tag{3}$$

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

$$\boldsymbol{G}(q)=\boldsymbol{A}^{-1}(q)\boldsymbol{B}(q)=\boldsymbol{B}_1(q)\boldsymbol{A}_1^{-1}(q) \tag{4}$$

where the polynomial matrices $\boldsymbol{A}\in R_{22}[q]$, $\boldsymbol{B}\in R_{22}[q]$ represent the left coprime factorization of matrix $\boldsymbol{G}(q)$ and the matrices $\boldsymbol{A}_1\in R_{22}[q]$, $\boldsymbol{B}_1\in R_{22}[q]$ represent the right coprime factorization of $\boldsymbol{G}(q)$. The further described algorithms are based on a model with polynomials of second order. This model proved to be effective for control of several TITO laboratory processes [13], where controllers based on a model with polynomials of the first order failed.

A. Discrete Model

Polynomial matrices of the discrete model are given by following exressions

$$\boldsymbol{A}\left(z^{-1}\right)=\begin{bmatrix} 1+a_1 z^{-1}+a_2 z^{-2} & a_3 z^{-1}+a_4 z^{-2} \\ a_5 z^{-1}+a_6 z^{-2} & 1+a_7 z^{-1}+a_8 z^{-2} \end{bmatrix} \tag{5}$$

$$\boldsymbol{B}\left(z^{-1}\right)=\begin{bmatrix} b_1 z^{-1}+b_2 z^{-2} & b_3 z^{-1}+b_4 z^{-2} \\ b_5 z^{-1}+b_6 z^{-2} & b_7 z^{-1}+b_8 z^{-2} \end{bmatrix} \tag{6}$$

The matrices can be converted to difference equations

$$\begin{aligned} y_1(k)=&-a_1 y_1(k-1)-a_2 y_1(k-2)-a_3 y_2(k-1)-a_4 y_2(k-2)+ \\ &+b_1 u_1(k-1)+b_2 u_1(k-2)+b_3 u_2(k-1)+b_4 u_2(k-2) \end{aligned} \tag{7}$$

$$\begin{aligned} y_2(k)=&-a_5 y_1(k-1)-a_6 y_1(k-2)-a_7 y_2(k-1)-a_8 y_2(k-2)+ \\ &+b_5 u_1(k-1)+b_6 u_1(k-2)+b_7 u_2(k-1)+b_8 u_2(k-2) \end{aligned} \tag{8}$$

B. Continuous-Time Model

Polynomial matrices of the continuous-time model are defined as follows

$$\boldsymbol{A}(s)=\begin{bmatrix} s^2+a_1 s+a_2 & a_3 s+a_4 \\ a_5 s+a_6 & s^2+a_7 s+a_8 \end{bmatrix} \tag{9}$$

$$\boldsymbol{B}(s)=\begin{bmatrix} b_1 s+b_2 & b_3 s+b_4 \\ b_5 s+b_6 & b_7 s+b_8 \end{bmatrix} \tag{10}$$

Differential equations describing dynamical behavior of the system are

$$y_1'' + a_1 y_1' + a_2 y_1 + a_3 y_2' + a_4 y_2 = b_1 u_1' + b_2 u_1 + b_3 u_2' + b_4 u_2 \tag{11}$$

$$y_2'' + a_5 y_1' + a_6 y_1 + a_7 y_2' + a_8 y_2 = b_5 u_1' + b_6 u_1 + b_7 u_2' + b_8 u_2 \tag{12}$$

III. Design of 2DOF Controllers

The 2DOF configuration of the closed loop system is depicted in Fig. 1. It was presented in [10] for SISO control loop.

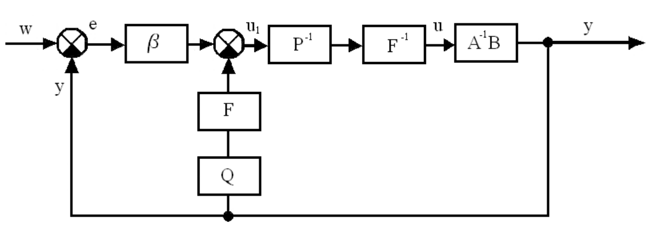

Fig. 1 Block diagram of 2DOF configuration

The vector of input reference signals is defined as

$$\boldsymbol{W}(q)=\boldsymbol{F}_w^{-1}(q)\boldsymbol{h}(q) \tag{13}$$

The operator $q = s$ as the derivative operator for continuous time systems and $q = z^{-1}$ as the delay operator for discrete systems. Further, the reference signals are considered as step functions. In this case $\boldsymbol{h}(q)$ is a vector of constants and $\boldsymbol{F}_w(q)$ is in the case of the discrete system expressed as

$$\boldsymbol{F}_w\left(z^{-1}\right)=\begin{bmatrix} 1-z^{-1} & 0 \\ 0 & 1-z^{-1} \end{bmatrix} \tag{14}$$

and in the case of the continuous-time system as

$$\boldsymbol{F}_w(s)=\begin{bmatrix} s & 0 \\ 0 & s \end{bmatrix} \tag{15}$$

The compensator $\boldsymbol{F}(q)$ is a component formally separated from the controller. It has to be included in the controller to fulfil the requirement on the asymptotic tracking. If the reference signals are step functions, then $\boldsymbol{F}(q)$ is an integrator.

It is possible to derive the following equation for the system output (operator q will be omitted from some operations for the purpose of simplification)

$$\boldsymbol{Y}=\boldsymbol{A}^{-1}\boldsymbol{B}\boldsymbol{U}=\boldsymbol{A}^{-1}\boldsymbol{B}\boldsymbol{F}^{-1}\boldsymbol{P}^{-1}\boldsymbol{U}_1 \tag{16}$$

Where

$$\boldsymbol{U}_1=\boldsymbol{\beta}(\boldsymbol{W}-\boldsymbol{Y})-\boldsymbol{QFY} \tag{17}$$

The corresponding equation for the controller's output, as shown in the block diagram in Fig. 1, follows as

$$\boldsymbol{U}=\boldsymbol{F}^{-1}\boldsymbol{P}^{-1}\boldsymbol{U}_1 \tag{18}$$

The substitution of $\boldsymbol{U}_1$ and $\boldsymbol{Y}$ results in

$$\boldsymbol{U}=\boldsymbol{F}^{-1}\boldsymbol{P}^{-1}\left[\boldsymbol{\beta}\left(\boldsymbol{W}-\boldsymbol{A}^{-1}\boldsymbol{BU}\right)-\boldsymbol{QFA}^{-1}\boldsymbol{BU}\right] \tag{19}$$

The equation (12) can be modified using the right matrix fraction of the controlled system into the form

$$\boldsymbol{U}=\boldsymbol{A}_1\left[\boldsymbol{PFA}_1+(\boldsymbol{\beta}+\boldsymbol{FQ})\boldsymbol{B}_1\right]^{-1}\boldsymbol{\beta W} \tag{20}$$

The determinant of the matrix in the denominator $\boldsymbol{PFA}_1+(\boldsymbol{\beta}+\boldsymbol{FQ})\boldsymbol{B}_1$ is the characteristic polynomial of the MIMO system. The roots of this polynomial matrix determine the behaviour of the closed loop system. They must be placed on the left side of the Gauss complex plane for the system to be stable. Conditions of BIBO stability can be defined by the following Diophantine matrix equation:

$$\boldsymbol{PFA}_1+(\boldsymbol{\beta}+\boldsymbol{FQ})\boldsymbol{B}_1=\boldsymbol{M} \tag{21}$$

where $\boldsymbol{M}\in R_{22}[q]$ is a stable diagonal polynomial matrix. If the system has the same number of inputs and outputs, matrix $\boldsymbol{M}$ can be chosen as diagonal, which allows easier computation of the controller parameters. Correct pole placement of the matrix $\boldsymbol{M}$ is very important for good control performance.

For the continuous-time case the matrix $\boldsymbol{M}$ takes the following form

$$\boldsymbol{M}(s)=\begin{bmatrix} s^4+m_1s^3+ & 0 \\ +m_2s^2+m_3s+m_4 & \\ 0 & s^4+m_5s^3+m_6s^2+ \\ & +m_7s+m_8 \end{bmatrix} \tag{22}$$

and for the discrete system it takes the form

$$\boldsymbol{M}(z^{-1})=\begin{bmatrix} 1+m_1z^{-1}+m_2z^{-2}+ & 0 \\ +m_3z^{-3}+m_4z^{-4} & \\ 0 & 1+m_1z^{-1}+m_2z^{-2}+ \\ & +m_3z^{-3}+m_4z^{-4} \end{bmatrix} \tag{23}$$

A. Design of Discrete Controller

The degree of the controller polynomial matrices depends on the internal properness of the closed loop. The structures of matrices $\boldsymbol{P}$, $\boldsymbol{Q}$ and $\boldsymbol{\beta}$ were chosen so that the number of unknown controller parameters equals the number of algebraic equations resulting from the solution of the Diophantine equation (21) using the method of uncertain coefficients:

$$\boldsymbol{P}(z^{-1})=\begin{bmatrix} 1+p_1z^{-1} & p_2z^{-1} \\ p_3z^{-1} & 1+p_4z^{-1} \end{bmatrix} \tag{24}$$

$$\boldsymbol{Q}(z^{-1})=\begin{bmatrix} q_1+q_2z^{-1} & q_3+q_4z^{-1} \\ q_5+q_6z^{-1} & q_7+q_8z^{-1} \end{bmatrix} \tag{25}$$

$$\boldsymbol{\beta}(z^{-1})=\begin{bmatrix} \beta_1 & \beta_2 \\ \beta_3 & \beta_4 \end{bmatrix} \tag{26}$$

The solution of the Diophantine equation results in a set of algebraic equations with unknown controller parameters.

$$\begin{bmatrix} 1 & 0 & b_9 & 0 & b_{13} & 0 & b_9 & b_{13} \\ a_9-1 & a_{13} & b_{10}-b_9 & b_9 & b_{14}-b_{13} & b_{13} & b_{10} & b_{14} \\ a_{10}-a_9 & a_{14}-a_{13} & -b_{10} & b_{10}-b_9 & -b_{14} & b_{14}-b_{13} & 0 & 0 \\ -a_{10} & -a_{14} & 0 & -b_{10} & 0 & -b_{14} & 0 & 0 \\ 0 & 1 & b_{11} & 0 & b_{15} & 0 & b_{11} & b_{15} \\ a_{11} & a_{15}-1 & b_{12}-b_{11} & b_{11} & b_{16}-b_{15} & b_{15} & b_{12} & b_{16} \\ a_{12}-a_{11} & a_{16}-a_{15} & -b_{12} & b_{12}-b_{11} & -b_{16} & b_{16}-b_{15} & 0 & 0 \\ -a_{12} & -a_{16} & 0 & -b_{12} & 0 & -b_{16} & 0 & 0 \end{bmatrix}\begin{bmatrix} p_1 \\ p_2 \\ q_1 \\ q_2 \\ q_3 \\ q_4 \\ \beta_1 \\ \beta_2 \end{bmatrix}=\begin{bmatrix} m_1-a_9+1 \\ m_2+a_9-a_{10} \\ m_3+a_{10} \\ m_4 \\ -a_{11} \\ a_{11}-a_{12} \\ a_{12} \\ 0 \end{bmatrix} \tag{27}$$

$$\begin{bmatrix} 1 & 0 & b_9 & 0 & b_{13} & 0 & b_9 & b_{13} \\ a_9-1 & a_{13} & b_{10}-b_9 & b_9 & b_{14}-b_{13} & b_{13} & b_{10} & b_{14} \\ a_{10}-a_9 & a_{14}-a_{13} & -b_{10} & b_{10}-b_9 & -b_{14} & b_{14}-b_{13} & 0 & 0 \\ -a_{10} & -a_{14} & 0 & -b_{10} & 0 & -b_{14} & 0 & 0 \\ 0 & 1 & b_{11} & 0 & b_{15} & 0 & b_{11} & b_{15} \\ a_{11} & a_{15}-1 & b_{12}-b_{11} & b_{11} & b_{16}-b_{15} & b_{15} & b_{12} & b_{16} \\ a_{12}-a_{11} & a_{16}-a_{15} & -b_{12} & b_{12}-b_{11} & -b_{16} & b_{16}-b_{15} & 0 & 0 \\ -a_{12} & -a_{16} & 0 & -b_{12} & 0 & -b_{16} & 0 & 0 \end{bmatrix}\begin{bmatrix} p_3 \\ p_4 \\ q_5 \\ q_6 \\ q_7 \\ q_8 \\ \beta_3 \\ \beta_4 \end{bmatrix}=\begin{bmatrix} -a_{13} \\ a_{13}-a_{14} \\ a_{14} \\ 0 \\ m_5-a_{15}+1 \\ m_6+a_{15}-a_{16} \\ m_7+a_{16} \\ m_8 \end{bmatrix} \tag{28}$$

The controller parameters are obtained by solving these equations. The parameters are then used for computation of the control law. The control law is defined as:

$$\boldsymbol{FPU}=\boldsymbol{\beta E}-\boldsymbol{FQY} \tag{29}$$

where $\boldsymbol{E}$ is a vector of control errors. This matrix equation can be transcribed to the difference equations of the controller

$$\begin{aligned} u_1(k)&=\beta_1e_1(k)+\beta_2e_2(k)-q_1y_1(k)-(q_2-q_1)y_1(k-1)+ \\ &q_2y_1(k-2)-q_3y_2(k)-(q_4-q_3)y_2(k-1)+q_4y_2(k-2)- \\ &-(p_1-1)u_1(k-1)+p_1u_1(k-2)-p_2u_2(k-1)+p_2u_2(k-2) \end{aligned} \tag{30}$$

$$\begin{aligned} u_2(k)&=\beta_3e_1(k)+\beta_4e_2(k)-q_5y_1(k)-(q_6-q_5)y_1(k-1)+ \\ &q_6y_1(k-2)-q_7y_2(k)-(q_8-q_7)y_2(k-1)+q_8y_2(k-2)- \\ &-(p_4-1)u_2(k-1)+p_4u_2(k-2)-p_3u_1(k-1)+p_3u_1(k-2) \end{aligned} \tag{31}$$

B. Design of Continuous-Time Controller

Polynomial matrices of the continuous-time controller are as follows:

$$\boldsymbol{P}=\begin{bmatrix} s+p_1 & p_2 \\ p_3 & s+p_4 \end{bmatrix} \tag{32}$$

$$\boldsymbol{Q}=\begin{bmatrix} q_1s+q_2 & q_3s+q_4 \\ q_5s+q_6 & q_7s+q_8 \end{bmatrix} \tag{33}$$

$$\boldsymbol{\beta}=\begin{bmatrix} \beta_1 & \beta_2 \\ \beta_3 & \beta_4 \end{bmatrix} \tag{34}$$

The solution of the Diophantine equation results in a set of algebraic equations with unknown controller parameters. Using matrix notation, the algebraic equations are expressed in the following form.

$$\begin{bmatrix} 1 & a_1 & a_2 & 0 & 0 & a_3 & a_4 & 0 \\ 0 & a_5 & a_6 & 0 & 1 & a_7 & a_8 & 0 \\ b_1 & b_2 & 0 & 0 & b_3 & b_4 & 0 & 0 \\ 0 & b_1 & b_2 & 0 & 0 & b_3 & b_4 & 0 \\ b_5 & b_6 & 0 & 0 & b_7 & b_8 & 0 & 0 \\ 0 & b_5 & b_6 & 0 & 0 & b_7 & b_8 & 0 \\ 0 & 0 & b_1 & b_2 & 0 & 0 & b_3 & b_4 \\ 0 & 0 & b_5 & b_6 & 0 & 0 & b_7 & b_8 \end{bmatrix}\begin{bmatrix} p_1 \\ p_2 \\ q_1 \\ q_2 \\ q_3 \\ q_4 \\ \beta_1 \\ \beta_2 \end{bmatrix}=\begin{bmatrix} m_1-a_9 \\ m_2-a_{10} \\ m_3 \\ m_4 \\ -a_{11} \\ -a_{12} \\ 0 \\ 0 \end{bmatrix} \tag{35}$$

$$\begin{bmatrix} 1 & a_1 & a_2 & 0 & 0 & a_3 & a_4 & 0 \\ 0 & a_5 & a_6 & 0 & 1 & a_7 & a_8 & 0 \\ b_1 & b_2 & 0 & 0 & b_3 & b_4 & 0 & 0 \\ 0 & b_1 & b_2 & 0 & 0 & b_3 & b_4 & 0 \\ b_5 & b_6 & 0 & 0 & b_7 & b_8 & 0 & 0 \\ 0 & b_5 & b_6 & 0 & 0 & b_7 & b_8 & 0 \\ 0 & 0 & b_1 & b_2 & 0 & 0 & b_3 & b_4 \\ 0 & 0 & b_5 & b_6 & 0 & 0 & b_7 & b_8 \end{bmatrix}\begin{bmatrix} p_3 \\ p_4 \\ q_5 \\ q_6 \\ q_7 \\ q_8 \\ \beta_3 \\ \beta_4 \end{bmatrix}=\begin{bmatrix} -a_{13} \\ -a_{14} \\ 0 \\ 0 \\ m_5-a_{15} \\ m_6-a_{16} \\ m_7 \\ m_8 \end{bmatrix} \tag{36}$$

The matrix equation (29) can be transcribed to the differential equations of the controller

$$\begin{aligned} &u_1''+p_1u_1'= \\ &=\beta_1e_1+\beta_2e_2-q_1y_1''-q_2y_1'-q_3y_2''-q_4y_2'-p_2u_2' \end{aligned} \tag{37}$$

$$u_2'' + p_4 u_2' = \\ = \beta_3 e_1 + \beta_4 e_2 - q_5 y_1'' - q_6 y_1' - q_7 y_2'' - q_8 y_2' - p_3 u_1' \tag{38}$$

For purposes of simulation, the controller was realized in the Matlab/Simulink environment as an S-function. It was then necessary to obtain its state equations. Further there it is introduced a conversion of the first differential equation (37) to the state equations. The second differential equation (38) was converted similarly. Equation (37) can be itemized as follows

$$\begin{aligned} u_{1A}'' + p_1 u_{1A}' &= e_1 \beta_1 \\ u_{1B}'' + p_1 u_{1B}' &= e_2 \beta_2 \\ u_{1C}'' + p_1 u_{1C}' &= -q_1 y_1'' - q_2 y_1' \\ u_{1D}'' + p_1 u_{1D}' &= -q_3 y_2'' - q_4 y_2' \\ u_{1E}'' + p_1 u_{1E}' &= -p_2 u_2' \end{aligned} \tag{39}$$

Equations (39) can be transcribed to transfer functions. It is also possible to establish auxiliary variables Z_1, Z_2, Z_3, Z_4 and Z_5.

$$\begin{aligned} G_1(s) &= \frac{\beta_1}{s^2 + p_1 s} = \frac{U_{1A}}{E_1} = \frac{U_{1A}}{Z_1}\frac{Z_1}{E_1} \\ G_2(s) &= \frac{\beta_2}{s^2 + p_1 s} = \frac{U_{1B}}{E_2} = \frac{U_{1B}}{Z_2}\frac{Z_2}{E_2} \\ G_3(s) &= \frac{-q_1 s^2 - q_2 s}{s^2 + p_1 s} = \frac{U_{1C}}{Y_1} = \frac{U_{1C}}{Z_3}\frac{Z_3}{Y_1} \\ G_4(s) &= \frac{-q_3 s^2 - q_4 s}{s^2 + p_1 s} = \frac{U_{1D}}{Y_2} = \frac{U_{1D}}{Z_4}\frac{Z_4}{Y_2} \\ G_5(s) &= \frac{-p_2 s}{s^2 + p_1 s} = \frac{U_{1E}}{U_2} = \frac{U_{1E}}{Z_5}\frac{Z_5}{U_2} \end{aligned} \tag{40}$$

By means of the variables Z_1, Z_2, Z_3, Z_4 and Z_5 it is possible to define following equations

$$\begin{aligned} \beta_1 z_1 &= u_{1A} \\ \beta_2 z_2 &= u_{1B} \\ -q_1 z_3'' - q_2 z_3' &= u_{1C} \\ -q_3 z_4'' - q_4 z_4' &= u_{1D} \\ -p_2 z_5' &= u_{1E} \end{aligned} \tag{41}$$

and

$$\begin{aligned} z_1'' + p_1 z_1' &= e_1 \\ z_2'' + p_1 z_2' &= e_2 \\ z_3'' + p_1 z_3' &= y_1 \\ z_4'' + p_1 z_4' &= y_2 \\ z_5'' + p_1 z_5' &= u_2 \end{aligned} \tag{42}$$

Equations (42) can be converted to a set of differential equations of the first order (state equations). Choice of the state variables is as follows

$$\begin{aligned} x_1 &= z_1 \\ x_2 &= z_1' \\ x_3 &= z_2 \\ x_4 &= z_2' \\ x_5 &= z_3 \\ x_6 &= z_3' \\ x_7 &= z_4 \\ x_8 &= z_4' \\ x_9 &= z_5 \\ x_{10} &= z_5' \end{aligned} \tag{43}$$

And the state equations are

$$\begin{aligned} x_1' &= x_2 \\ x_2' &= e_1 - p_1 x_2 \\ x_3' &= x_4 \\ x_4' &= e_2 - p_1 x_4 \\ x_5' &= x_6 \\ x_6' &= y_1 - p_1 x_6 \\ x_7' &= x_8 \\ x_8' &= y_2 - p_1 x_8 \\ x_9' &= x_{10} \\ x_{10}' &= u_2 - p_1 x_{10} \end{aligned} \tag{44}$$

On the basis of the state variables, which are substituted to equations (41), it is possible to derive particular parts of the manipulated variable u_1

$$\begin{aligned} u_{1A} &= \beta_1 x_1 \\ u_{1B} &= \beta_2 x_3 \\ u_{1C} &= -q_1(y_1 - p_1 x_6) - q_2 x_6 \\ u_{1D} &= -q_3(y_2 - p_1 x_8) - q_4 x_8 \\ u_{1E} &= -p_2 x_{10} \end{aligned} \tag{45}$$

The manipulated variable we obtain as sum (46)

$$u_1 = u_{1A} + u_{1B} + u_{1C} + u_{1D} + u_{1E} \tag{46}$$

An expression for computation of the manipulated variable u_2 is obtained similarly on the basis of differential equation (38).

IV. System Identification

The control algorithm was applied as a self-tuning controller. Self-tuning control is based on the online identification of a model of a controlled process. Each self – tuning controller consists of an on – line identification part and a control part.

Various discrete linear models are used to describe dynamic behaviour of controlled systems; see for example the overview in [14]. The most widely applied linear dynamic model is the ARX model. Usually the ARX model is tested first and more complex model structures are only examined if it does not perform satisfactorily. However, the ARX model matches the structure of many real processes. The parameters can be easily estimated by a linear least-squares technique.

A. Identification of Discrete Model

The ARX model describing the TITO process is defined as

$$\begin{aligned} y_1(k) &= \boldsymbol{\Theta}_1(k)\phi(k-1) + e_{s1}(k) \\ y_2(k) &= \boldsymbol{\Theta}_2(k)\phi(k-1) + e_{s2}(k) \end{aligned} \tag{47}$$

where $e_{s1}(k)$, $e_{s2}(k)$ are non-measurable disturbances.

Parameter vectors are specified as follows:

$$\boldsymbol{\Theta}_1^T(k) = [a_1, a_2, a_3, a_4, b_1, b_2, b_3, b_4] \tag{48}$$

$$\boldsymbol{\Theta}_2^T(k) = [a_5, a_6, a_7, a_8, b_5, b_6, b_7, b_8]$$

The data vector is

$$\begin{aligned} \phi^T(k-1) = [&y_1(k-1), y_1(k-2), y_2(k-1), \\ &y_2(k-2), u_1(k-1), u_1(k-2), u_2(k-1), u_2(k-2)] \end{aligned} \tag{49}$$

The aim of the identification is a recursive estimation of unknown model parameters $\boldsymbol{\Theta}$ on the basis of the inputs and the outputs considering the time moment k t_k, $\{y(i), u(i), i = k, k - 1, k - 2, ..., k_0\}$ (where k_0 is an initial time of the identification). We are looking for a vector $\hat{\boldsymbol{\Theta}}$ minimizing the criterion

$$J_k(\boldsymbol{\Theta}) = \sum_{i=k_0}^{k} e_s^2(i) \tag{50}$$

where

$$e_s(i) = y(i) - \boldsymbol{\Theta}^T \phi(i) = \begin{bmatrix} 1 & -\boldsymbol{\Theta}^T \end{bmatrix} \begin{bmatrix} y(i) \\ \phi(i) \end{bmatrix} \tag{51}$$

When using the least squares method, the influence of all measured input and output samples to the parameter estimates is the same. This is inconvenient for the identification of nonlinear systems, where changes in the identified parameters are expected. Tracking of changes of the parameters can be achieved using exponential forgetting. This technique ensues from the assumption that new data describe the dynamics of an object better than older data, which are multiplied by smaller weighting coefficients. However, if the identified plant is insufficiently activated, the input and output signals are steady (this situation is typical for closed control systems), and the exponential forgetting factor can cause numerical instability of the identification algorithm. A possible solution of this problem is the application of adaptive directional forgetting [15]. This technique changes the forgetting factor according to the level of information in the data. In view of the parameter changes in the nonlinear coupled-drives apparatus and the expected insufficient activation of the controlled system, the recursive least squares method with adaptive directional forgetting was applied. Then we minimize a modified criterion

$$J_k(\boldsymbol{\Theta}) = \sum_{i=k_0}^{k} \varphi^{2(k-i)} e_s^2(i) \tag{52}$$

where $0 \langle \varphi^2 \le 1$ is the exponential forgetting factor.

The vector of parameters is updated according to the following recursive expression

$$\hat{\boldsymbol{\Theta}}(k) = \hat{\boldsymbol{\Theta}}(k-1) + \frac{\boldsymbol{C}(k-1)\phi(k-1)}{1+\xi(k-1)} \hat{e}(k-1) \tag{53}$$

Where

$$\xi(k-1) = \phi^T(k-1)\boldsymbol{C}(k-1)\phi(k-1) \tag{54}$$

is an auxiliary scalar and

$$\hat{e}(k-1) = y(k) - \hat{\boldsymbol{\Theta}}^T(k-1)\phi(k-1) \tag{55}$$

is a prediction error. If $\xi(k-1) > 0$, then the square covariance matrix C is updated according to following expression

$$\boldsymbol{C}(k) = \boldsymbol{C}(k-1) - \frac{\boldsymbol{C}(k-1)\phi(k-1)\phi^T(k-1)\boldsymbol{C}(k-1)}{\varepsilon^{-1}(k) + \xi(k-1)} \tag{56}$$

Where

$$\varepsilon(k) = \varphi(k) - \frac{1-\varphi(k)}{\xi(k-1)} \tag{57}$$

If $\xi(k-1) = 0$ then

$$\boldsymbol{C}(k) = \boldsymbol{C}(k-1) \tag{58}$$

The directional forgetting factor is computed in each sampling period according to the expression

$$\varphi(k) = \left\{1 + (1+\rho)[\ln(1+\xi(k-1))] + \left[\frac{(\upsilon(k-1)+1)\eta(k-1)}{1+\xi(k-1)+\eta(k-1)} - 1\right]\frac{\xi(k-1)}{1+\xi(k-1)}\right\}^{-1} \tag{59}$$

Where

$$\eta(k) = \frac{\hat{e}^2(k)}{\lambda(k)} \tag{60}$$

$$\upsilon(k) = \varphi(k)[(\upsilon(k-1)+1] \tag{61}$$

$$\lambda(k) = \varphi(k)\left[\lambda(k-1) + \frac{\hat{e}^2(k-1)}{1+\xi(k-1)}\right] \tag{62}$$

are auxiliary variables.

B. Identification of Continuous-Time Model

It is not possible to measure directly input and output derivatives of a system in case of continuous – time control loop. One of the possible approaches to this problem is establishing of filters and filtered variables to substitute the primary variables. This approach is described in detail in [16],

[17], [18]. The filtered variables are then used in the recursive identification procedure.

Let us consider a linear continuous – time ARX model in a form of differential equation

$$A(\sigma)y(t) = B(\sigma)u(t) + n(t) \tag{63}$$

where $n(t)$ is a random continuous – time variable and σ is the derivative operator. After the Laplace transform we obtain

$$A(s)Y(s) = B(s)U(s) + N(s) + O_1(s) \tag{64}$$

where the polynomial O_1 represents the Laplace transform of initial conditions. The output of the system is than given as

$$Y(s) = \frac{B(s)}{A(s)}U(s) + \frac{N(s)}{A(s)} + \frac{O_1(s)}{A(s)} \tag{65}$$

In order to obtain approximations of derivatives of the continuous – time variables it is necessary to establish filters using differential equations

$$C(\sigma)u_f(t) = u(t); \quad C(\sigma)y_f(t) = y(t) \tag{66}$$

where $C(\sigma)$ is a stable polynomial and u_f is a filtered input and y_f is a filtered output. After the Laplace transform we obtain

$$C(s)U_f(s) = U(s) + O_2(s); \quad C(s)Y_f(s) = Y(s) + O_3(s) \tag{67}$$

where O_2(s) is a polynomial of initial conditions for the filtered input and O_3(s) is a polynomial of initial conditions for the filtered output. The degree of the polynomial c must be greater or equal to the degree of the polynomial A (*deg C*(s)> *deg A*(*s*)). It is profitable to choose *deg C*(*s*) = *deg A*(*s*) (the lower is the degree of the polynomial *C*, the faster is the dynamics of the filter). Time constants of the filters must be lower than time constants of the plant. A right choice of the filter's constants makes convergence of the parameters faster.

After substitution of the filtered variables to the equation (66) we obtain

$$A\left[CY_f(s) - O_3\right] = B\left[CU_f - O_2\right] + N(s) + O_1 \tag{68}$$

After modification and substitution

$$AY_f(s) = BU_f(s) + \frac{O_1 - BO_2 + AO_3 + N(s)}{C} \tag{69}$$

and substitution

$$O = \frac{O_1 - BO_2 + AO_3}{C} \tag{70}$$

we obtain

$$Y_f(s) = \frac{B}{A}U_f(s) + \frac{O}{A} + \frac{1}{A}N(s) \quad \Rightarrow \quad G_f(s) = \frac{B}{A} = G(s) \tag{71}$$

Expression (73) proves that the transfer behaviour between the filtered and between the non – filtered variables is equivalent. Different are only initial conditions for the filtered and original variables. This fact enables to employ the filtered variables for the model parameter estimation.

After transformation to the time domain we obtain the following equation

$$A(\sigma)y_f(t) = B(\sigma)\, u_f(t) + n(t) \tag{72}$$

The filtered variables are taken in discrete time intervals *tk* = *kTs*, *k* = 0,1,2, …, where *Ts* is the sampling period. The equation (74) can be modified to the form suitable for the model parameters estimation

$$y^{(n)}{}_f(t_k) = -\sum_{i=0}^{n-1} a_i y_f^{(i)}(t_k) + \sum_{j=0}^{m} b_j u_f^{(j)}(t_k) + n(t_k) \tag{73}$$

The parameters of the model are estimated by the recursive method described in the previous section according to expressions (55) - (64). For the considered continuous – time model given by expressions (9) - (12) the equation (75) takes following form

$$\begin{aligned} y''_{1f}(t_k) &= -a_1 y'_{1f}(t_k) - a_2 y_{1f}(t_k) - a_3 y'_{2f}(t_k) - a_4 y_{2f}(t_k) + \\ &+ b_1 u'_{1f}(t_k) + b_2 u_{1f}(t_k) + b_3 u'_{2f}(t_k) + b_4 u_{2f}(t_k) + \varepsilon_1(t_k) \end{aligned} \tag{74}$$

$$\begin{aligned} y''_{2f}(t_k) &= -a_5 y'_{1f}(t_k) - a_6 y_{1f}(t_k) - a_7 y'_{2f}(t_k) - a_8 y_{2f}(t_k) + \\ &+ b_5 u'_{1f}(t_k) + b_6 u_{1f}(t_k) + b_7 u'_{2f}(t_k) + b_8 u_{2f}(t_k) + \varepsilon_2(t_k) \end{aligned} \tag{75}$$

The regression vector and the vector of parameters are

$$\begin{aligned} \phi_{1,2}{}^T(t_k) &= [-y'_{1f}(t_k), -y_{1f}(t_k), -y'_{2f}(t_k), -y_{2f}(t_k), \\ &-u'_{1f}(t_k), -u_{1f}(t_k), -u'_{2f}(t_k), -u_{2f}(t_k), 1] \end{aligned} \tag{76}$$

$$\boldsymbol{\Theta}_1{}^T(t_k) = \left[a_1, a_2, a_3, a_4, b_1, b_2, b_3, b_4, d_1\right] \tag{77}$$

$$\boldsymbol{\Theta}_2{}^T(t_k) = \left[a_5, a_6, a_7, a_8, b_5, b_6, b_7, b_8, d_2\right] \tag{78}$$

Considering the order of the system, the filters for both variables were chosen to have second order.

$$\begin{aligned} y''_{1f}(t) + c_1 y'_{1f}(t) + c_0 y_{1f}(t) &= y_1(t) \\ y''_{2f}(t) + c_1 y'_{2f}(t) + c_0 y_{2f}(t) &= y_2(t) \\ u''_{1f}(t) + c_1 u'_{1f}(t) + c_0 u_{1f}(t) &= u_1(t) \\ u''_{2f}(t) + c_1 u'_{2f}(t) + c_0 u_{2f}(t) &= u_2(t) \end{aligned} \tag{79}$$

A right choice of the coefficients of the filter's polynomials and choice of the sampling period are the ruling factors for the speed of the parameter's convergence. Time constants of the filters must be lower than time constants of the plant.

V. Simulation Verification

The proposed controllers were verified by simulation. Verification by simulation was carried out on a range of plants with various dynamics.

A. Simulation of Discrete Control

As a simulation example for the discrete controller it is shown control of a system which represents a linear model of a coupled drives process obtained by the recursive identification for a particular steady state [13].

$$\boldsymbol{A}\left(z^{-1}\right)=\begin{bmatrix} 1-0.5827z^{-1}+0.1745z^{-2} & -0.0220z^{-1}+0.1797z^{-2} \\ 0.0167z^{-1}-0.0886z^{-2} & 1-0.4564z^{-1}-0.0830z^{-2} \end{bmatrix} \quad (80)$$

$$\boldsymbol{B}\left(z^{-1}\right)=\begin{bmatrix} -0.0035z^{-1}+0.0955z^{-2} & 0.1484z^{-1}+0.2197z^{-2} \\ 0.2783z^{-1}+0.3107z^{-2} & -0.0371z^{-1}-0.3489z^{-2} \end{bmatrix} \quad (81)$$

The step response of the system is in Fig. 2.

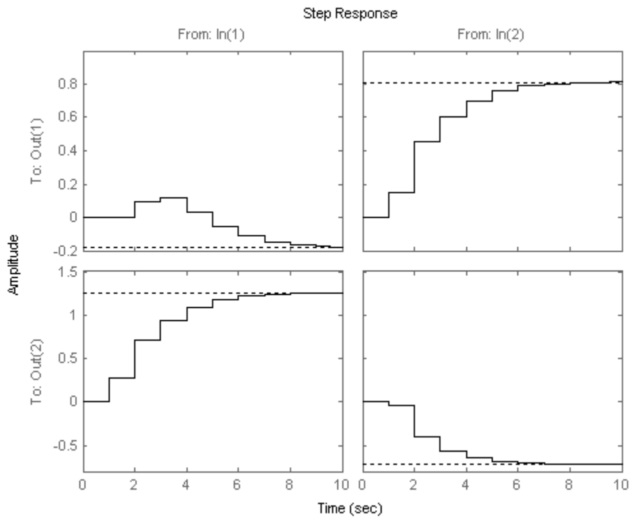

Fig. 2 Step response of the discrete system

The tuning parameter is the matrix $\boldsymbol{M}$. A suitable pole-placement (matrix $\boldsymbol{M}$) was chosen experimentally. At first, a multiple pole was chosen on the real axis. A suitable position of the multiple pole was chosen by experiments and comparison of control results. Then it was searched a suitable combination of various poles in the neighbourhood of the multiple pole.

$$\boldsymbol{M}\left(z^{-1}\right)=\begin{bmatrix} 1-0.9z^{-1}+0.19z^{-2}- \\ -0.009z^{-3}-0.002z^{-4} & 0 \\ 0 & 1-0.9z^{-1}+0.19z^{-2}- \\ & -0.009z^{-3}-0.002z^{-4} \end{bmatrix} \quad (82)$$

The time responses of the control are shown in Fig. 3-4

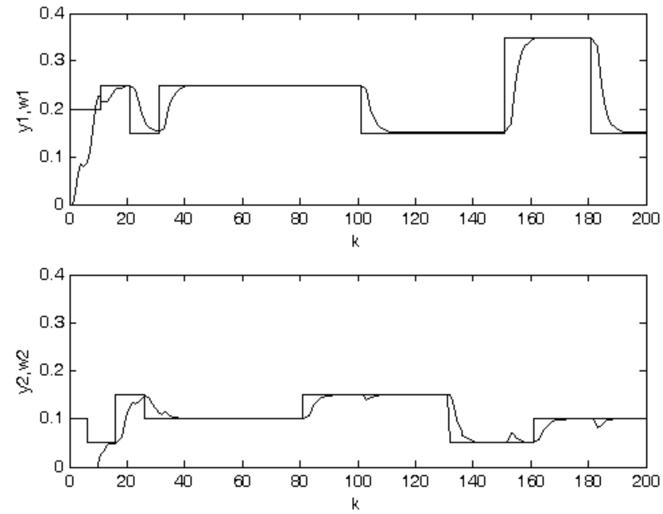

Fig. 3 Adaptive control with discrete controller

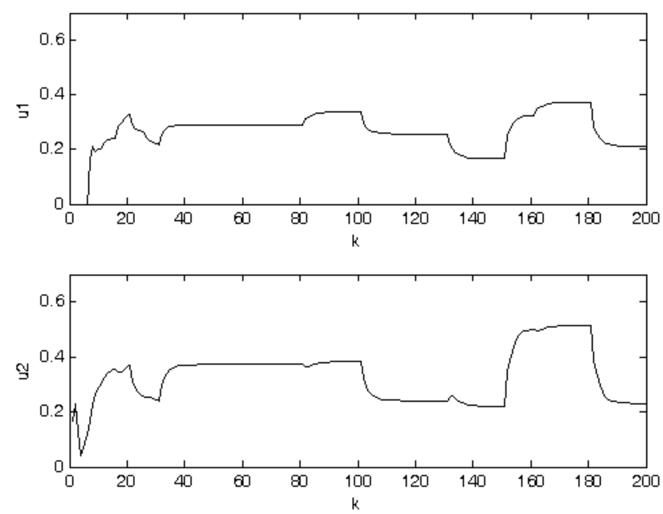

Fig. 4 Adaptive control with discrete controller-manipulated variables

B. *Simulation of Continuous-Time Control*

A continuous-time model in the form of the matrix fraction obtained by a possible conversion of the discrete model does not need to have the structure on which it is based the computation of the control law. The model obtained by this way would by then unusable.

It is shown control of the following continuous-time system

$$\boldsymbol{A}(s)=\begin{bmatrix} s^2+2s+0{,}7 & 0{,}2s+0{,}4 \\ -0{,}5s-0{,}1 & s^2+2s+0{,}7 \end{bmatrix} \quad (83)$$

$$\boldsymbol{B}(s)=\begin{bmatrix} 0{,}5s+0{,}2 & 0{,}1s+0{,}3 \\ 0{,}5s+0{,}1 & 0{,}3s+0{,}4 \end{bmatrix} \quad (84)$$

Fig. 5 shows the system‘s step response

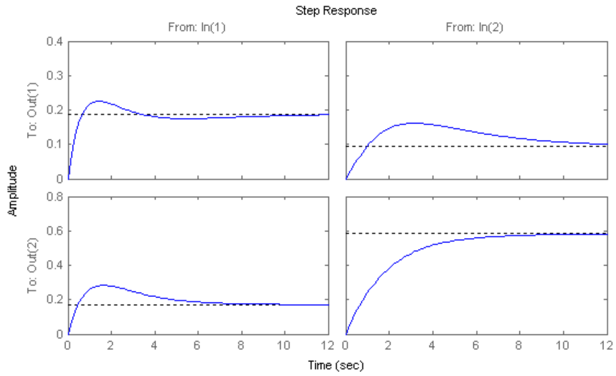

Fig. 5 Step response of the continuous-time system

The matrix $\boldsymbol{M}$ was obtained as follows

$$\boldsymbol{M}(s)=\begin{bmatrix} s^4+2s^3+ & \\ +9s^2+6s+2 & 0 \\ 0 & s^4+2s^3+ \\ & +9s^2+6s+2 \end{bmatrix} \tag{85}$$

The time responses of the control are shown in Fig. 6-7.

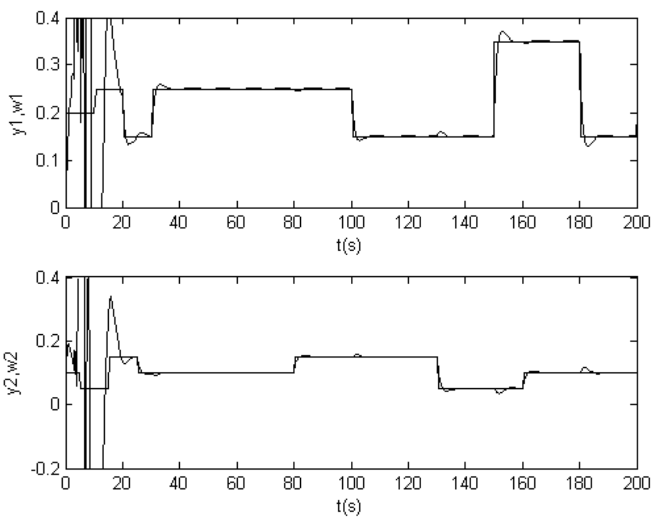

Fig. 6 Adaptive control with continuous-time controller

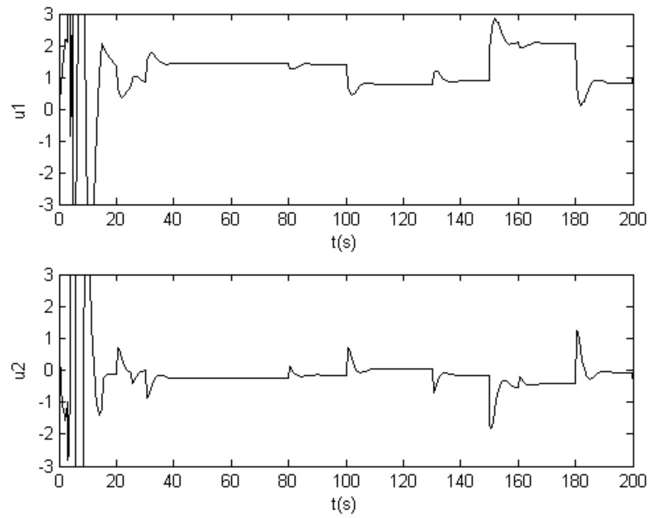

Fig. 7 Adaptive control with continuous-time controller-manipulated variables

From the courses of the variables in Fig.4-7 it is obvious that the basic requirements on control were satisfied. The system was stabilized and the asymptotic tracking of the reference signals was achieved.

VI. Conclusion

The main content of this paper is utilization of algebraic methods for design of multivariable control system for the system with two inputs and two outputs. System with two inputs and two outputs is the most simple and the most common case of a multivariable system. Many industrial processes have character of a system with two inputs and two outputs. The design of the controller is based on a linear model of the controlled system in the form of matrix fraction. The structure of the model is chosen in advance. The method of pole assignment was used. The method of polynomial synthesis was utilized which lead to solution of the Diophantine matrix equation.

Algorithms based on the apparatus and terms of linear algebra have a number of advantages and are widely used. They are simple and easily programmable. The synthesis lies in the formulation of the Diophantine equation. A great advantage of this method is its simple applicability for multivariable systems. In the framework of this paper were designed both continuous-time and discrete versions of the control system based on 2DOF configuration.

In the identification part of the self-tuning controller recursive least squares method with the directional forgetting was applied. In the continuous-time version the method was modified for estimation of a continuous-time model using filtering of continuous-time variables. The regression vector was filled with the filtered values as derivatives of the controlled and manipulated variables are not possible to be directly obtained. The designed controller was implemented in the Matlab/Simulink environment and verified by simulation.

REFERENCES

[1] T. Tsay, "Stability Analyses of Nonlinear Multivariable Feedback Control Systems", WSEAS Transactions on Systems, Vol. 11, Issue 4, 2012, pp. 140-151.

[2] P.R. Krishnawamy, et al., "Reference System Decoupling for Multivariable Control". *Ind. Eng. Chem. Res.*, 30, 1991, pp. 662-670.

[3] I.L. Chien, D.E.Seborg and D. A. Mellichamp, "Self-Tuning Control with Decoupling", *AIChE J.,* Vol. 33, No. 7, 1987, pp. 1079 – 1088.

[4] V. Kučera," Stochastic multivariable control: a polynomial approach", *IEEE Trans. of Automatic Control*, 5, pp. 913–919,1980.

[5] V. Kučera, V., *Analysis and Design of Discrete Linear Control Systems*. Prentice Hall, Englewood Cliffs, New Jersey, 1991.

[6] Vidyasagar, M., *Control System Synthesis: A Factorization Approach*. MIT Press, Cambridge MA, 1985.

[7] J. Vojtesek, P. Dostal, "Pole-placement and LQ Hybrid Adaptive Control Applied Isothermal Continuous Stirred Tank Reactor", WSEAS Transactions on Systems and Control, Vol. 8, Issue 3, 2014, pp. 63-72.

[8] S.K. Choudhary, "Robust Feedback Control Analysis of Magnetic Levitation System", WSEAS Transactions on Systems, Vol. 13, 2014, pp. 285-291.

[9] V. Kučera, "Diophantine Equations in Control – a Survey". *Automatica*, 29, 1993, pp. 1361 - 1375.

[10] R. Ortega, and R. Kelly, PID self – tuners: Some theoretical and practical aspects, IEEE Trans. Ind. Electron., IE - 3, pp. 332 – 338, 1984.

[11] I. D. Landau, R. Lozano, M. M'Saad, *Adaptive Control* (Springer - Verlag, Berlin, 1998).

[12] V. Bobal, J., Bőhm , J., Fessl, J., Machacek, *Digital Self-Tuning Controllers* (Springer - Verlag, London, 2005).

[13] 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, pp. 641-654.

[14] O. Nelles, *Nonlinear System Identification* (Springer-Verlag, Berlin, 2001).

[15] R. Kulhavý, "Restricted exponential forgetting in real time identification", *Automatica*, 23, 1987, 586-600.

[16] P. J. Gawthrop, *Continuous – Time Self – Tuning Control*, John Willey, Chichester, 1987.

[17] H. Unbehauen, G.P. Rao, *Identification of Continuous Systems*, North Holland, Amsterdam, 1987.

[18] B. Wahlberg, B. On the identification of continuous – time dynamical systems, *Report LiTH-ISY-I-0905*, Linköping, 1990.