Self-tuning Predictive Control of a Coupled Drives Process

Marek Kubalčík, Vladimír Bobál

***Abstract*—** This paper is focused on an application of a self – tuning predictive controller for real – time control of a coupled drives apparatus laboratory model, which models a multivariable non-linear system. The controller integrates an on – line identification of an ARX model of a controlled system and a predictive control synthesis on the basis of the identified parameters. The model parameters are recursively estimated using the recursive least squares method with the directional forgetting. The control algorithm is based on the Generalised Predictive Control (GPC) method. The optimization was realized by minimization of a quadratic objective function. The predictive controllers design is based on a multivariable CARIMA model. Results of real-time experiments are also included.

***Keywords*—** Predictive control, adaptive control, multivariable systems, recursive identification, nonlinear systems.

I. Introduction

TYPICAL technological processes require the simultaneous control of several variables related to one system. Each input may influence all system outputs. The coupled drives apparatus (Fig. 1) is a typical multivariable nonlinear system with significant cross – coupling. The design of a controller for such a system must be quite sophisticated if the system is to be controlled adequately. Simple decentralized PI or PID controllers largely do not yield satisfactory results. There are many different advanced methods of controlling multi-input–multi-output (MIMO) systems. The problem of selecting an appropriate control technique often arises. Perhaps the most popular way of controlling MIMO processes is by designing decoupling compensators to suppress the interactions (e.g. [1]) and the designing multiple SISO controllers (e.g. [2]). This requires determining how to pair the controlled and manipulated variables and that the plant has the same number of inputs and outputs. One of the most effective approaches to control of multivariable systems is model predictive control (MPC) [3], [4], [5], [6], [7], [8], [9] . An advantage of model predictive control is that multivariable systems can be handled in a straightforward manner. When using most of other approaches, the control actions are taken based on past errors. MPC uses also future values of the reference signals.

The aim of this contribution is implementation of the adaptive predictive controller handling constraints of the manipulated variable for control of the coupled drives apparatus laboratory model. The objective laboratory model is a nonlinear system with variable parameters. Self-tuning controllers [10], [11] are a possible approach to the control of this kind of system. The controller is then realized as self – tuning controller with recursive identification of the model of the process. The recursive least squares method with the directional forgetting is used in the identification part.

Dynamic behaviour of the system is described in the neighbourhood of a steady state by a discrete linear model in the form of matrix fraction, which represents a MIMO transfer function model. It is an input – output model (“black box model”) which does not take into consideration an internal structure of the system. It is a model of the system behaviour and its parameters do not have any particular physical denotation. The model is used to generate system predictions. The simplest possible model which gives accurate enough predictions is used.

The Generalised Predictive Control (GPC) method [12], [13] was applied. In the optimization part of the algorithm a quadratic cost function was used. A recursive algorithm which enables computation of predictions for arbitrary horizons was designed.

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).

II. Description of the Apparatus

The coupled-drives experimental laboratory model was designed to demonstrate simultaneous control of the tension and speed of material in a continuous process. It is based on experience with industrial control applications. Industrial coupled-drives systems are basic components of production lines, where material is manufactured in the form of a continuous strip. The material passes workstations, where its speed and tension are measured. The material speed and tension must be controlled within the defined limits. Practical examples are in the paper-making industry, strip metal and wire manufacturing. Electrical drives can be coupled together in many ways. The coupled-drives laboratory model represents the standard coupled-drives system, shown in Fig. 1

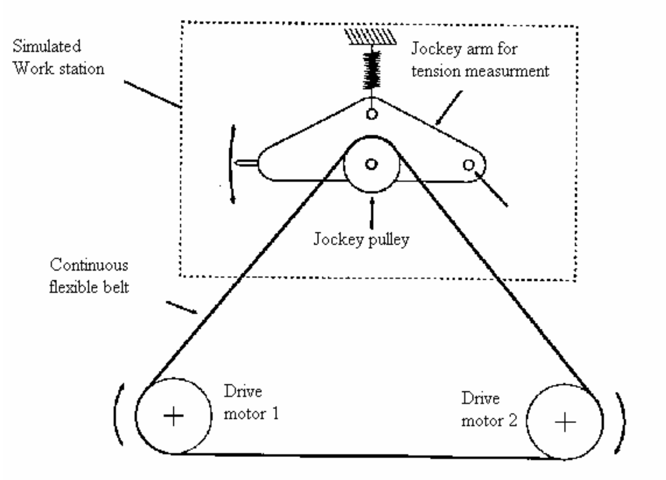


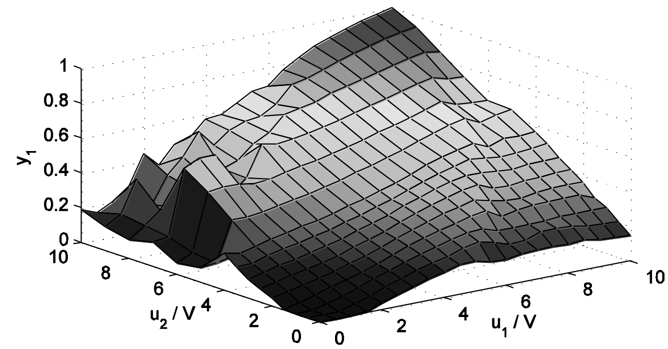
Fig. 1 Principles of the coupled-drives apparatus

The apparatus consists of three pulleys mounted in a vertical panel to form a triangle. The two base pulleys are directly mounted on the shafts of two nominally identical drive motors (motor 1 and motor 2) and the apparatus is controlled by manipulating the drive torques of these motors. The third pulley, the jockey, rotates freely and is mounted on a pivoted arm. The drive motors are coupled by a continuous flexible belt, which also passes over the pivoted arm. The jockey pulley assembly, which simulates a material workstation, is instrumented to allow measurement of the belt speed and tension. The jockey pulley's angular velocity and the belt tension are the system outputs. The belt tension is measured indirectly by monitoring the angular deflection of the pivoted tension arm to which the jockey pulley is attached. The deflection of the arm is then a measure of the tension in the belt.

The continuous flexible belt couples the actions of motor 1 and motor 2. If a drive voltage to motor 1's drive input is applied, then the speed and the tension in the belt will be changed and motor 2 will be rotated by the drag from motor 1. A similar result is achieved if a drive voltage is applied to motor 2's drive input. Both motors change both outputs. This is the coupling. The system inputs and outputs interact and the whole system is a multivariable system. The manipulated variables are the inputs to the servomotors and the controlled variables are the belt tension and the angular velocity of the jockey pulley. The apparatus can be considered as a two-input–two-output (TITO) system.

The range of the input voltage of the motors is 0–10 V, the range of the angle of the jockey arm is –10° to 10°.

The static characteristics of the apparatus were measured experimentally to determine the system linearity ranges. All the characteristics show non-linear behaviour: the belt tension characteristic is non-linear over the whole range because of belt oscillations. The static characteristics are shown in Fig. 2. The variable y_1 denotes the angular velocity and the variable y_2 the tension of the belt. The variables u_1 and u_2 are the voltage inputs of the left (motor 1) and right (motor 2) drive motors. The non-linear behaviour is caused by slipping and oscillation of the belt. From Fig. 2, it is obvious that as the difference in motor speeds increases, the slipping and the oscillations become more apparent.



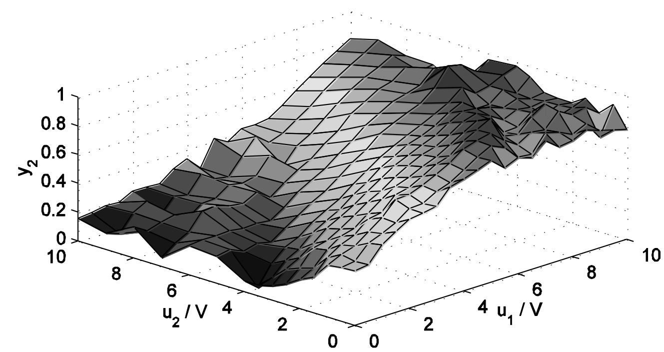


Fig. 2 Static characteristics of the coupled-drives apparatus

The step responses of the system were measured. The system was stabilized with 50% of the maximum drive voltages applied to both electric motors. Then, steps of 10% of the maximum drive voltage were applied to each motor separately. The responses are shown in Fig. 3 (a). The graphs in the first column are step responses of the angular velocity (y_1) and the belt tension (y_2) to the step applied to motor 1. Analogously, the second column shows responses to the step applied to motor 2. The shapes of the step responses are indeterminate. They were not then a ruling factor for assessing the model order, which is elaborated in later sections.

Further measurements were performed to examine the degree of linearity of the dynamic characteristics (e.g., finding whether the time constants change with the input magnitude). An example of the results is shown in Fig. 3 (b). First, the system was stabilized at 20% of the maximum drive voltages of both servomotors and then steps of 20, 40 and 60% of the drive voltage were applied to motor 1. It was evident that the dominant time constant changed with the input magnitude in the different operational ranges. If 60% of the drive voltage was applied, the time constant was different from those for the two other step responses. Figure 3 (b) also proves a non-linear relation between the input and output.

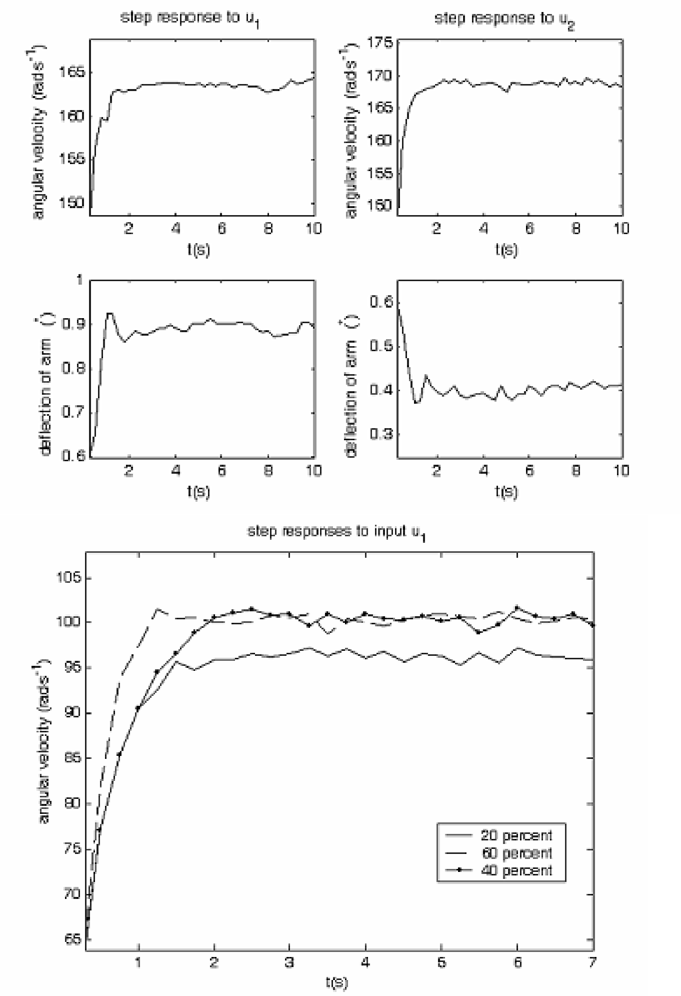


Fig. 3 Step responses of the coupled-drives apparatus

The measurements of the characteristics proved that the coupled-drives apparatus is a non-linear system with variable parameters. Self-tuning controllers are a possible approach to the control of this kind of system. The non-linear dynamics are described by a linear model in the neighbourhood of a steady state.

A suitable model of the real object for control with self-tuning controllers is an input–output model. This is a standard approach for self-tuning controllers. Instead of the often tedious construction of a model from first principles and then calculating its parameters from plant dimensions and physical constants, a general model is chosen and its parameters are identified from data. The advantages of this kind of model are its simplicity and accuracy in the operational range in which the input–output dependence is measured.

III. Mathematical Model of the Controlled Process

A simplified analytical model of the coupled-drives apparatus, based on physics and the equipment construction where all the parameters have physical interpretations, is presented in [14]. The main disadvantage of this model is its high complexity. Some simplifications were also required during its derivation and some assumptions with limited accuracy were used. The laboratory model is a nonlinear system, as it was mentioned above. Self-tuning controllers are a possible approach to the control of this kind of system. The nonlinear dynamics are described by a linear model in the neighbourhood of a steady state. A suitable model of the real object for control with self-tuning controllers is an input–output model. It is a model of the system behaviour and its parameters do not necessarily have physical interpretations. Of course, not all properties of the plant can be extracted from the data in this way, but when the dominant properties are modelled, the result is sufficient for controller design. The advantages of this kind of model are its simplicity and accuracy in the operational range in which the input–output dependence is measured.

It was necessary to determine a structure of the model in advance. The aim here was to find experimentally the simplest possible structure of the model. The parameters are identified during the process of recursive identification from the measured input and output signals.

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

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

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

where $\boldsymbol{U}(z)$ and $\boldsymbol{Y}(z)$ are vectors of the manipulated variables (inputs to the servomotors) and the controlled variables (tension and speed of the belt), respectively.

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

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

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

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

$$\boldsymbol{A}(z^{-1})\boldsymbol{Y}(z)=\boldsymbol{B}(z^{-1})\boldsymbol{U}(z) \tag{5}$$

The control algorithm was first designed for a model with polynomials of the first order. This model proved to be unsuitable for the coupled-drives process description and satisfactory control results were not achieved. Consequently, the algorithm was designed for a model with second-order polynomials. This model proved to be effective and sufficiently complex to describe the coupled-drives process, while enabling quite simple computation of the controller. The controller described below is based on this model. The model has 16 parameters:

$$\boldsymbol{A}\left(z^{-1}\right)=\begin{bmatrix}1+a_1z^{-1}+a_2z^{-2} & a_3z^{-1}+a_4z^{-2}\\ a_5z^{-1}+a_6z^{-2} & 1+a_7z^{-1}+a_8z^{-2}\end{bmatrix} \tag{6}$$

$$\boldsymbol{B}\left(z^{-1}\right)=\begin{bmatrix}b_1z^{-1}+b_2z^{-2} & b_3z^{-1}+b_4z^{-2}\\ b_5z^{-1}+b_6z^{-2} & b_7z^{-1}+b_8z^{-2}\end{bmatrix} \tag{7}$$

A widely used model in general model predictive control is the CARIMA model which we can obtain from the nominal model (5) by adding a disturbance model

$$\boldsymbol{A}\left(z^{-1}\right)\boldsymbol{Y}(z)=\boldsymbol{B}\left(z^{-1}\right)\boldsymbol{U}(z)+\boldsymbol{C}\left(z^{-1}\right)\boldsymbol{\Delta}^{-1}\boldsymbol{E}_S(z) \tag{8}$$

where $\boldsymbol{E}_s\left(z^{-1}\right)$ is a non-measurable random disturbance that is assumed to have zero mean value and constant covariance and the inverted operator delta is an integrator. The matrix $\boldsymbol{C}\left(z^{-1}\right)$ will be further considered as 2x2 identity matrix.

Application of this model enables to achieve an integral action.

IV. Design of the Controller

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

$$J(k)=\sum_{j=1}^{N}\boldsymbol{e}(k+j)^2+\lambda\sum_{j=1}^{N_u}\Delta\boldsymbol{u}(k+j)^2 \tag{9}$$

where e(k+j) is a vector of predicted control errors, Δu(k+j) is a vector of future increments of manipulated variables (for the system with two inputs and two outputs each vector has two elements), N is length of the prediction horizon, Nu is length of the control horizon and λ is a weighting factor of control increments.

A predictor in a vector form is given by

$$\hat{\boldsymbol{y}}=\boldsymbol{G}\Delta\boldsymbol{u}+\boldsymbol{y}_0 \tag{10}$$

Where is a vector of system predictions along the horizon of the length N, Δu is a vector of control increments, y0 is the free response vector. G is a matrix of the dynamics given as

$$\boldsymbol{G}=\begin{bmatrix}\boldsymbol{G}_0 & 0 & \cdots & \cdots & 0\\ \boldsymbol{G}_1 & \boldsymbol{G}_0 & 0 & \cdots & 0\\ \vdots & & \ddots & \ddots & \vdots\\ \vdots & & & \boldsymbol{G}_0 & 0\\ \boldsymbol{G}_{N-1} & \cdots & \cdots & \cdots & \boldsymbol{G}_0\end{bmatrix} \tag{11}$$

where sub-matrices Gi have dimension 2x2 and contain values of the step sequence.

The first task was computation of predictions for arbitrary prediction and control horizons. Dynamics of the coupled drives process requires horizons of length where it is not possible to compute predictions in the simple straightforward way. Recursive expressions for computation of the free response and the matrix G in each sampling period had to be derived. There are several different ways of deriving the prediction equations for matrix fraction models. Some papers make use of Diophantine equations to form the prediction equations (e.g. [15]). In [16] matrix methods are used to compute predictions. We derived a method for recursive computation of both the free response and the matrix of the dynamics by direct use of the CARIMA model. Its difference equations without the unknown term can be expressed as:

$$\begin{aligned}&y_1(k+1)=(1-a_1)y_1(k)+(a_1-a_2)y_1(k-1)+a_2y_1(k-2)-\\&-a_3y_2(k)+(a_3-a_4)y_2(k-1)+a_4y_2(k-2)+\\&+b_1\Delta u_1(k)+b_2\Delta u_1(k-1)+b_3\Delta u_2(k)+b_4\Delta u_2(k-1)\\&y_2(k+1)=(1-a_7)y_2(k)+(a_7-a_8)y_2(k-1)+a_8y_2(k-2)-\\&-a_5y_1(k)+(a_5-a_6)y_1(k-1)+a_6y_1(k-2)+\\&+b_5\Delta u_1(k)+b_6\Delta u_1(k-1)+b_7\Delta u_2(k)+b_8\Delta u_2(k-1)\end{aligned} \tag{12}$$

These equations can be written in a matrix form

$$\begin{aligned}&\boldsymbol{y}(k+1)=\boldsymbol{A}_1\boldsymbol{y}(k)+\boldsymbol{A}_2\boldsymbol{y}(k-1)+\boldsymbol{A}_3\boldsymbol{y}(k-2)+\\&+\boldsymbol{B}_1\Delta\boldsymbol{u}(k)+\boldsymbol{B}_2\Delta\boldsymbol{u}(k-1)\end{aligned} \tag{13}$$

where

$$\boldsymbol{A}_1=\begin{bmatrix}1-a_1 & -a_3\\ -a_5 & 1-a_7\end{bmatrix} \tag{14}$$

$$\boldsymbol{A}_2=\begin{bmatrix}a_1-a_2 & a_3-a_4\\ a_5-a_6 & a_7-a_8\end{bmatrix} \tag{15}$$

$$\boldsymbol{A}_3=\begin{bmatrix}a_2 & a_4\\ a_6 & a_8\end{bmatrix} \tag{16}$$

$$\boldsymbol{B}_1=\begin{bmatrix}b_1 & b_3\\ b_5 & b_7\end{bmatrix} \tag{17}$$

$$\boldsymbol{B}_2=\begin{bmatrix}b_2 & b_4\\ b_6 & b_8\end{bmatrix} \tag{18}$$

It was necessary to compute three step ahead predictions in a straightforward way by substituting of previous predictions to later predictions. The model order defines that computation of one step ahead prediction is based on three past values of the system output. It is possible to divide computation of the predictions to recursion of the free response and recursion of the matrix of the dynamics. The free response vector can be expressed as:

$$\boldsymbol{y}_0 = \left[\begin{array}{cc|cc|cc|cc} p(1,1) & p(1,2) & p(1,3) & p(1,4) & p(1,5) & p(1,6) & p(1,7) & p(1,8) \\ p(2,1) & p(2,2) & p(2,3) & p(2,4) & p(2,5) & p(2,6) & p(2,7) & p(2,8) \\ \hline p(3,1) & p(3,2) & p(3,3) & p(3,4) & p(3,5) & p(3,6) & p(3,7) & p(3,8) \\ p(4,1) & p(4,2) & p(4,3) & p(4,4) & p(4,5) & p(4,6) & p(4,7) & p(4,8) \\ \hline p(5,1) & p(5,2) & p(5,3) & p(5,4) & p(5,5) & p(5,6) & p(5,7) & p(5,8) \\ p(6,1) & p(6,2) & p(6,3) & p(6,4) & p(6,5) & p(6,6) & p(6,7) & p(6,8) \end{array}\right] \left[\begin{array}{c} y_1(k) \\ y_2(k) \\ \hline y_1(k-1) \\ y_2(k-1) \\ \hline y_1(k-2) \\ y_2(k-2) \\ \hline \Delta u_1(k-1) \\ \Delta u_2(k-1) \end{array}\right] =$$

$$= \begin{bmatrix} \boldsymbol{P}(1,1) & \boldsymbol{P}(1,2) & \boldsymbol{P}(1,3) & \boldsymbol{P}(1,4) \\ \boldsymbol{P}(2,1) & \boldsymbol{P}(2,2) & \boldsymbol{P}(2,3) & \boldsymbol{P}(2,4) \\ \boldsymbol{P}(3,1) & \boldsymbol{P}(3,2) & \boldsymbol{P}(3,3) & \boldsymbol{P}(3,4) \end{bmatrix} \begin{bmatrix} \boldsymbol{y}(k) \\ \boldsymbol{y}(k-1) \\ \boldsymbol{y}(k-2) \\ \Delta\boldsymbol{u}(k-1) \end{bmatrix} = \boldsymbol{P} \begin{bmatrix} \boldsymbol{y}(k) \\ \boldsymbol{y}(k-1) \\ \boldsymbol{y}(k-2) \\ \Delta\boldsymbol{u}(k-1) \end{bmatrix} \tag{19}$$

All the elements P(i,j) i=1…3, j=1…4 have to be directly computed to initialize the recursion. The next row of the matrix P is repeatedly computed on the basis of the three previous predictions until the prediction horizon is achieved. As an illustrative example it is given the computation of the next element of the first column:

$$\boldsymbol{P}(4,1) = \begin{bmatrix} p(7,1) & p(7,2) \\ p(8,1) & p(8,2) \end{bmatrix} = \boldsymbol{A}_1\boldsymbol{P}(3,1) + \boldsymbol{A}_2\boldsymbol{P}(2,1) + \boldsymbol{A}_3\boldsymbol{P}(1,1) \tag{20}$$

The recursion of the matrix G is similar. The next element of the first column is repeatedly computed and the remaining columns are shifted. This procedure is performed repeatedly until the prediction horizon is achieved. If the control horizon is lower than the prediction horizon a number of columns in the matrix is reduced. The technique is apparent from the equations (20) and (21).

$$\boldsymbol{G}\Delta\boldsymbol{u} = \left[\begin{array}{cc|cc} g(1,1) & g(1,2) & 0 & 0 \\ g(2,1) & g(2,2) & 0 & 0 \\ \hline g(3,1) & g(3,2) & g(1,1) & g(1,2) \\ g(4,1) & g(4,2) & g(2,1) & g(2,2) \\ \hline g(5,1) & g(5,2) & g(3,1) & g(3,2) \\ g(6,1) & g(6,2) & g(4,1) & g(4,2) \end{array}\right] \left[\begin{array}{c} \Delta u_1(k) \\ \Delta u_2(k) \\ \hline \Delta u_1(k+1) \\ \Delta u_2(k+1) \end{array}\right] = \tag{21}$$

$$= \begin{bmatrix} G(1,1) & 0 \\ G(2,1) & G(1,1) \\ G(3,1) & G(2,1) \end{bmatrix} \begin{bmatrix} \Delta\boldsymbol{u}(k) \\ \Delta\boldsymbol{u}(k+1) \end{bmatrix}$$

The criterion (9) can be written in a general vector form

$$J = (\hat{\boldsymbol{y}} - \boldsymbol{w})^T (\hat{\boldsymbol{y}} - \boldsymbol{w}) + \lambda \Delta\boldsymbol{u}^T \Delta\boldsymbol{u} \tag{22}$$

where w is a vector of the reference trajectory. The criterion can be modified using the expression (10) to

$$J = 2\boldsymbol{g}^T \Delta\boldsymbol{u} + \Delta\boldsymbol{u}^T \boldsymbol{H} \Delta\boldsymbol{u} \tag{23}$$

where the gradient g and the Hess matrix H are defined by following expressions

$$\boldsymbol{g}^T = \boldsymbol{G}^T (\boldsymbol{y}_0 - \boldsymbol{w}) \tag{24}$$

$$\boldsymbol{H} = \boldsymbol{G}^T \boldsymbol{G} \tag{25}$$

In case of the coupled drives apparatus, the actuators have a limited range of action. The voltages applied to the motors can vary between fixed limits. MPC can consider constrained input and output signals in the process of the controller design [17]. This is one of the major advantages of predictive control. General formulation of predictive control with constraints is then as follows

$$\min_{\Delta\boldsymbol{u}} 2\boldsymbol{g}^T \Delta\boldsymbol{u} + \Delta\boldsymbol{u}^T \boldsymbol{H} \Delta\boldsymbol{u} \tag{26}$$

owing to

$$\boldsymbol{A}\Delta\boldsymbol{u} \le \boldsymbol{b} \tag{27}$$

The inequality (27) expresses the constraints in a compact form. In our case of the constrained input signals particular matrices can be expressed as

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{T} \\ -\boldsymbol{T} \end{bmatrix} \quad \boldsymbol{b} = \begin{bmatrix} \boldsymbol{1u}_{\min} - \boldsymbol{1u}(k-1) \\ -\boldsymbol{1u}_{\max} + \boldsymbol{1u}(k-1) \end{bmatrix} \tag{28}$$

Forms of the matrices for an arbitrary control horizon are as follows

$$\begin{bmatrix} 1 & 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 1 & 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ 1 & 0 & 1 & 0 & \cdots & 1 & 0 \\ 0 & 1 & 0 & 1 & \cdots & 0 & 1 \\ -1 & 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & -1 & 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ -1 & 0 & -1 & 0 & \cdots & -1 & 0 \\ 0 & -1 & 0 & -1 & \cdots & 0 & -1 \end{bmatrix} \begin{bmatrix} \Delta u_1(k) \\ \Delta u_2(k) \\ \Delta u_1(k+1) \\ \Delta u_2(k+1) \\ \vdots \\ \Delta u_1(k+N_u) \\ \Delta u_2(k+N_u) \end{bmatrix} \ge \tag{29}$$

$$\ge \begin{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ \vdots & \vdots \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_{1\min} \\ u_{2\min} \end{bmatrix} - \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ \vdots & \vdots \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_1(k-1) \\ u_2(k-1) \end{bmatrix} \\ -\begin{bmatrix} 1 & 0 \\ 0 & 1 \\ \vdots & \vdots \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_{1\max} \\ u_{2\max} \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ \vdots & \vdots \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_1(k-1) \\ u_2(k-1) \end{bmatrix} \end{bmatrix}$$

The optimization problem is then solved numerically by quadratic programming in each sampling period. The first element of the resulting vector is then applied as the increment of the manipulated variable.

V. System Identification

The control algorithm was applied as a self-tuning controller (as discussed in sections 1 and 3). 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 [18]. 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. It is suitable also for the proposed predictive controller, because the parameters of the incremental CARIMA model are equal to the parameters of the ARX model in our case when the matrix is the identity matrix.

The ARX model describing the TITO process is defined as

$$\begin{aligned} y_1(k) &= \boldsymbol{\Theta}_1(k)\phi(k-1)+e_{s1}(k) \\ y_2(k) &= \boldsymbol{\Theta}_2(k)\phi(k-1)+e_{s2}(k) \end{aligned} \tag{30}$$

where $e_{s1}(k)$, $e_{s2}(k)$ are non-measurable disturbances.
Parameter vectors are specified as follows:

$$\begin{aligned} \boldsymbol{\Theta}_1^T(k) &= [a_1,a_2,a_3,a_4,b_1,b_2,b_3,b_4] \\ \boldsymbol{\Theta}_2^T(k) &= [a_5,a_6,a_7,a_8,b_5,b_6,b_7,b_8] \end{aligned} \tag{31}$$

The data vector is

$$\begin{aligned} \phi^T(k-1) = [&y_1(k-1), y_1(k-2), y_2(k-1), \\ &y_2(k-2), u_1(k-1), u_1(k-2), u_2(k-1), u_2(k-2)] \end{aligned} \tag{32}$$

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

$$J_k(\boldsymbol{\Theta}) = \sum_{i=k_0}^{k} e_s^2(i) \tag{33}$$

where

$$e_s(i) = y(i) - \boldsymbol{\Theta}^T\phi(i) = \begin{bmatrix} 1 & -\boldsymbol{\Theta}^T \end{bmatrix} \begin{bmatrix} y(i) \\ \phi(i) \end{bmatrix} \tag{34}$$

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 [19]. 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

$$J_k(\boldsymbol{\Theta}) = \sum_{i=k_0}^{k} \varphi^{2(k-i)} e_s^2(i) \tag{35}$$

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

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

$$\hat{\boldsymbol{\Theta}}(k) = \hat{\boldsymbol{\Theta}}(k-1) + \frac{\boldsymbol{C}(k-1)\phi(k-1)}{1+\xi(k-1)}\hat{e}(k-1) \tag{36}$$

Where

$$\xi(k-1) = \phi^T(k-1)\boldsymbol{C}(k-1)\phi(k-1) \tag{37}$$

is an auxiliary scalar and

$$\hat{e}(k-1) = y(k) - \hat{\boldsymbol{\Theta}}^T(k-1)\phi(k-1) \tag{38}$$

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

$$\boldsymbol{C}(k) = \boldsymbol{C}(k-1) - \frac{\boldsymbol{C}(k-1)\phi(k-1)\phi^T(k-1)\boldsymbol{C}(k-1)}{\varepsilon^{-1}(k)+\xi(k-1)} \tag{39}$$

Where

$$\varepsilon(k) = \varphi(k) - \frac{1-\varphi(k)}{\xi(k-1)} \tag{40}$$

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

$$\boldsymbol{C}(k) = \boldsymbol{C}(k-1) \tag{41}$$

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

$$\varphi(k) = \left\{1 + (1+\rho)[\ln(1+\xi(k-1))] + \left[\frac{(\upsilon(k-1)+1)\eta(k-1)}{1+\xi(k-1)+\eta(k-1)} - 1\right]\frac{\xi(k-1)}{1+\xi(k-1)}\right\}^{-1} \tag{42}$$

Where

$$\eta(k) = \frac{\hat{e}^2(k)}{\lambda(k)} \tag{43}$$

$$\upsilon(k) = \varphi(k)[(\upsilon(k-1)+1] \tag{44}$$

$$\lambda(k) = \varphi(k)\left[\lambda(k-1) + \frac{\hat{e}^2(k-1)}{1+\xi(k-1)}\right] \tag{45}$$

are auxiliary variables.

VI. Experimental Examples

The model was connected with a PC equipped with a control and measurement PC card. Matlab and Real Time Toolbox were used to control the system.

An approximate sampling period was found on the basis of measured step responses so that 10 samples would cover the important part of the step response. The sampling period was tuned experimentally and the best value was T0 = 0.25 s.

The tuning parameters that are lengths of the prediction and control horizons and the weighting coefficient λ were also tuned experimentally. There is a lack of clear theory relating to the closed loop behavior to design parameters. The length of the prediction horizon, which should cover the important part of the step response, was set to N=15. The length of the control horizon was also set to Nu=15. The coefficient λ was taken as equal to 2.

Figures 4 and 5 show time responses of the control when the initial parameter estimates were chosen without any a priori information. The reference trajectories contain frequent step changes in the beginning of experiments to activate input and output signals and improve the identification. The manipulated variables u1 and u2 are the inputs to the drive motors 1 and 2. The output y1 is the angular velocity and the output y2 is the tension of the belt.

In subsequent experiments, the initial parameter estimates were set to the values obtained at the end of the previous experiment. The initial conditions of the recursive identification were also modified by reducing the diagonal elements of the square covariance matrix that represent variances of the identified parameters. The reference trajectories were chosen to have the same values at the beginning as they had at the end of the previous experiments. This is because the system is nonlinear and the identified parameters were valid only for particular steady states. Time responses of these experiments are shown in Figs 6 and 7.

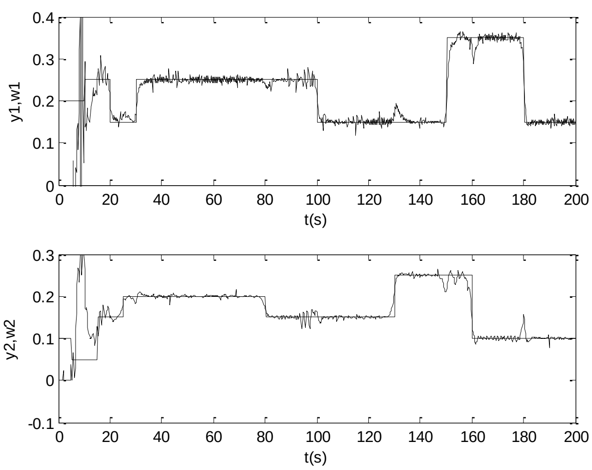


Fig. 4 Control of the coupled drives apparatus

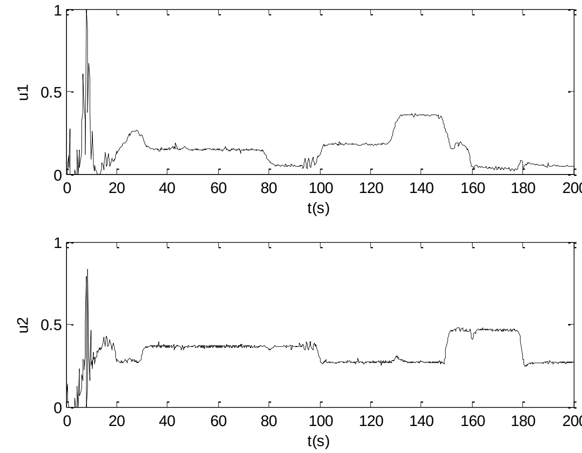


Fig. 5 Control of the coupled drives apparatus– manipulated variable

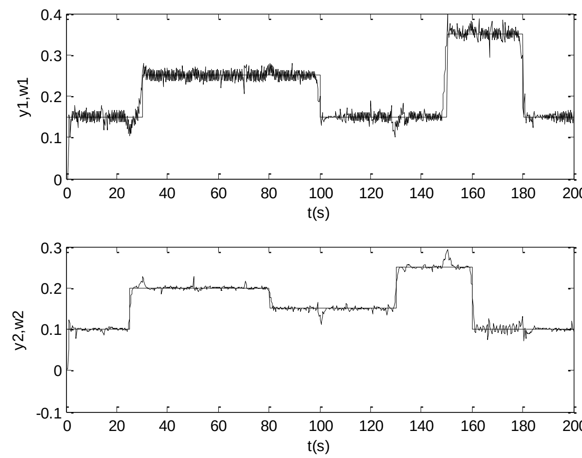


Fig. 6 Control of the coupled drives apparatus-experiment with steady parameters

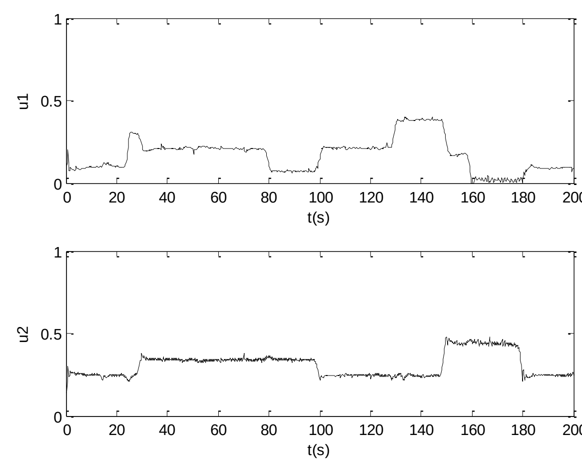


Fig. 7 Control of the coupled drives apparatus-experiment with steady parameters-manipulated variable

VII. Conclusion

Model predictive self-tuning controller was proposed and verified by control of nonlinear time varying system. The adaptive control strategy was applied especially due to nonlinear behaviour of the controlled system.

It is necessary to recognize that self-tuning controllers do not work satisfactorily in the initial adaptation phase if the initial parameter estimates are chosen without a priori information. However, the most important property for practical use of self-tuning controllers is their performance after the adaptation phase. The performance of the controller in the adaptation phase was significantly improved by choosing the initial parameter estimates with a priori information.

General principles were elaborated on a specific system with two inputs and two outputs that is often applicable in industrial practice. Control law based on specific model was derived in the form of self-contained expressions that is especially useful for practical applications of control on common industrial devices. An advantage of the proposed strategy lies in its simplicity and applicability. The control tests executed on the laboratory model provided very satisfactory results, even though its nonlinear dynamics were described by a linear model.

References

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

[2] W. L. Luyben, “Simple Method for Tuning SISO Controllers in Multivariable Systems”. *Ing. Eng. Chem. Process Des. Dev.*, 25, 1986, 654 – 660.

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

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

[5] R. R. Bitmead, M. Gevers, V. Hertz, *Adaptive Optimal Control. The Thinking Man’s GPC* (Prentice Hall, Englewood Cliffs, New Jersey, 1990).

[6] A. M. Yousef, “Model Predictive Control Approach Based LoadFrequency Controller”, WSEAS Transactions on Systems and Control, Vol. 6, Issue 7, July 2011, pp. 265-275, ISSN:1991-8763.

[7] Z. Ju, W. Wanliang, “Synthesis of Explicit Model Predictive Control System with Feasible Region Shrinking,” in Proc. 8th WSEAS Int. Conf.on Robotics, Control and Manufacturing Technology (ROCOM ’08), Hangzhon, China, 2008, pp. 80-85.

[8] A. H. Mazinan, N. Sadati, “Fuzzy Multiple Modeling and Fuzzy Predictive Control of a Tubular Heat Exchanger System,” in Proc. 7th WSEAS Int. Conf. on Application of Electrical Engineering (AEE ’08), Trondheim, Norway, 2008, pp. 77-82.

[9] P. Thitiyasook, P. Kittisupakorn, “Model Predictive Control of a Batch Reactor with Membrane – Based Separation,” in Proc. 7th WSEAS Int.Conf. on Signal Processing, Robotics and Automation (ISPRA ’08),University of Cambridge, UK, 2008, pp. 88-92.

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

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

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

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

[14] *CE108 Coupled Drives Apparatus*. Manual. (TecQuipment Ltd., Nottingham, England, 1997).

[15] W. H. Kwon, H. Choj, D. G. Byun, S. Noh, “Recursive solution of generalized predictive control and its equivalence to receding horizon tracking control”, *Automatica*, 28(6), 1992, 1235–1238.

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

[17] J.M. Maciejowski, *Predictive Control with Constraints* (Prentice Hall, London, 2002).

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

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