Input-output formulation of multidimensional adaptive predictive control

Květoslav Belda, Josef Böhm

Abstract
The paper deals with the design of predictive control intended for systems, which can be described by linear autoregressive models (ARX models), inclusive multidimensional cases. The model parameters are assumed to be constant or slightly time-varying and to be obtained on-line by identification of least squares. The design of predictive control arises from a state-space formulation. In the paper, the initial ARX model representing Input/Output form is transformed to specific ‘pseudo state-space’ form with non-minimal state. This form requires the same computational demands at control design as usual state-space approach.

Keywords: ARX models, on-line identification, state-space predictive control

Introduction

Systems in engineering practice usually change their properties. From mathematical point of view and also from control theory point of view, the key properties are being usually expressed by parameters included in a mathematical model, which describes considered system. Some of the parameters may be more or less constant without any connection to system property changes or may be variable. Changes may be caused either by a system wear or by its nonlinear character.

In control design, it is advantageous, if information on these changes can be involved in the computation of control actions. Model-based control strategies offer such possibility. In these strategies, the different types of parametric models are used. There are generally two ways to obtain values of the parameters. One way is to determine parameters by mathematical-physical analysis. The second way is using some experimental identification, at consideration of changeable parameters, running on-line.

The model-based control strategy, which uses models with appropriately changed parameters on-line, is usually called adaptive control strategy. In this paper, the one of such strategies – adaptive predictive control – will be investigated.

The predictive control [1] is nowadays very popular strategy mainly in chemical processes. However, it is also efficient in applications of mechanical engineering. It offers to simply manage multidimensional systems with different number of inputs and outputs [6] simultaneously with different types of constraints. Due to its multi-step character, it can optimize future control actions, which fit real demands. The tuning of the predictive control is not difficult and it follows directly from order of controlled system and from requirements on system behavior.

As a model with possible time-varying parameters, the autoregressive model with external input (ARX model) is considered in this paper.

The paper is organized as follows:

At first, the multidimensional ARX model will be defined. Then, the identification based on least square method will be outlined in square-root form. Next section will deal with reformulation of ARX model leading to state-space like model. In subsequent section, the derivation of predictive control will be shown in brief. Closing sections will contain several practical examples with laboratory models:

- ball on rod representing single-input and single-output system;
- simplified model of the helicopter representing system with two inputs and two outputs.

Model definition

Predictive control can be designed with different model forms. Standard forms are the ARX model (input-output formulation) and state-space model. The both have their pros and cons.

The ARX model represents unique description and is more suitable form Single-Input Single-Output (SISO) systems. It operates only with delayed inputs and outputs, therefore it does not need any observers.

On the other hand, the state-space model is not unique, but it is more transparent for multidimensional cases (i.e. for Multi-Input Multi-Output (MIMO) systems). In general, it needs state-space observer. In this paper, the advantages of both will be taken together.

Due to digital character of automating devices, the discrete control techniques are preferred. Therefore, the models used for control design are considered to be discrete in spite of the facts that controlled systems are continuous. Discrete realization is useful, because naturally respects finite predefined time for computation of control actions.

As was mentioned in introduction, the design of predictive control will be based on specific state-space formulation. However, as initial model, the ARX model is used.

Let us arise from definition of ARX model:

$$y(k) = \sum_{i=1}^{n} b_i\, u(k-i) - \sum_{i=1}^{n} a_i\, y(k-i) + e(k) \tag{1}$$

where n is a system order, $y(\cdot)$ and $u(\cdot)$ are values of system output and input, and $e(k)$ is model error, respectively noise of measurement of output $y(k)$. The coefficients b_i and a_i are model parameters. The model (1) can be also written in the following condensed forms; either in row parameter orientation

$$y(k) = \vartheta_k\, \mathbf{f}_k + e(k) \tag{2}$$

or in column parameter orientation $\theta_k = \vartheta_k^T$

$$y(k) = \mathbf{f}_k^T\, \theta_k + e(k) \tag{3}$$

i.e. vector of the parameters $\vartheta_k (= \theta_k^T)$ is defined as

$$\vartheta_k = [b_1 \cdots b_n\; -a_1 \cdots -a_n] \tag{4}$$

and data vector is composed from delayed values of control actions (inputs) and measured outputs

$$\mathbf{f}_k = [u(k-1) \cdots u(k-n)\; y(k-1) \cdots y(k-n)]^T \tag{5}$$

The model (1) and its condensed forms (2) and (3) are intended for SISO systems. They can serve also for MIMO systems. Thus, the model (1) for MIMO systems is:

$$\mathbf{y}(k) = \sum_{i=1}^{n} \mathbf{B}_i\, \mathbf{u}(k-i) - \sum_{i=1}^{n} \mathbf{A}_i\, \mathbf{y}(k-i) + \mathbf{e}(k) \tag{6}$$

where n is still system order, $\mathbf{u}(\cdot)$ and $\mathbf{y}(\cdot)$ are vectors of values of nu inputs and ny outputs, i.e.

$$\mathbf{u}(k-i) = [u_1(k-i), \cdots, u_{nu}(k-i)]^T$$

$$\mathbf{y}(k-i) = [y_1(k-i), \cdots, y_{ny}(k-i)]^T$$

and $\mathbf{e}(k)$ is an ny dimensional vector of noise of measurement of system outputs $\mathbf{y}(k)$. The model parameters are included in matrices $\mathbf{B}_i$ and $\mathbf{A}_i$

$$\mathbf{B}_i = \begin{bmatrix} b_i^{11} & \cdots & b_i^{1\,nu} \\ \vdots & \ddots & \vdots \\ b_i^{ny\,1} & \cdots & b_i^{ny\,nu} \end{bmatrix}, \quad \mathbf{A}_i = \begin{bmatrix} a_i^{11} & \cdots & a_i^{1\,nu} \\ \vdots & \ddots & \vdots \\ a_i^{ny\,1} & \cdots & a_i^{ny\,nu} \end{bmatrix}$$

The model (6) can be rewritten again in two possible forms.

One of them is a multivariate linear regression form

$$\begin{aligned} \mathbf{y}(k) &= [\mathbf{B}_1 \cdots \mathbf{B}_n\; \mathbf{A}_1 \cdots \mathbf{A}_n]\, \mathbf{f}_k + \mathbf{e}(k) \\ \mathbf{y}(k) &= \qquad \boldsymbol{\vartheta}_k \qquad\quad \mathbf{f}_k + \mathbf{e}(k) \end{aligned} \tag{7}$$

where individual coefficients (parameters) are in horizontal rectangular matrix $\boldsymbol{\vartheta}_k$; and $\mathbf{f}_k$ is

$$\mathbf{f}_k = [\mathbf{u}^T(k) \cdots \mathbf{u}^T(k-n)\;\; \mathbf{y}^T(k-1) \cdots \mathbf{y}^T(k-n)]^T$$

The second form is a fully polynomial form

$$\mathbf{y}(k) = \mathfrak{F}_k^T\, \theta_k + \mathbf{e}(k) \tag{8}$$

where individual parameters are situated in column vector θ_k and the data are in horizontal rectangular matrix $\mathfrak{F}_k^T$.

Their internal structures are the following:

$$\mathfrak{F}_k^T = \begin{bmatrix} \mathbf{f}_k^T & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \mathbf{f}_k^T \end{bmatrix}, \quad \mathfrak{F}_k^T \in \Re^{ny \times n(nu \cdot ny + ny \cdot ny)}$$

$$\theta_k = \begin{bmatrix} \theta_k^1 \\ \vdots \\ \theta_k^{ny} \end{bmatrix}, \quad \left(\begin{bmatrix} (\theta_k^1)^T \\ \vdots \\ (\theta_k^{ny})^T \end{bmatrix} = [\mathbf{B}_1 \cdots \mathbf{B}_n\; \mathbf{A}_1 \cdots \mathbf{A}_n] \right)$$

$$\begin{aligned} \theta_k^1 &= [b_1^{11} \cdot\cdot\, b_1^{1\,nu} \cdot\cdot\, b_n^{11} \cdot\cdot\, b_n^{1\,nu}\;\; a_1^{11} \cdot\cdot\, a_1^{1\,ny} \cdot\cdot\, a_n^{11} \cdot\cdot\, a_n^{1\,ny}]^T \\ &\vdots \\ \theta_k^{ny} &= [b_1^{ny1} \cdot\cdot\, b_1^{ny\,nu} \cdot\cdot\, b_n^{ny\,1} \cdot\cdot\, b_n^{ny\,nu}\;\; a_1^{ny1} \cdot\cdot\, a_1^{ny\,ny} \cdot\cdot\, a_n^{ny\,1} \cdot\cdot\, a_n^{ny\,ny}]^T \end{aligned}$$

The model (8) can be written separately for each system output

$$\begin{aligned} y_1(k) &= \mathbf{f}_k^T\, \theta_k^1 + e_1(k) \\ &\vdots \\ y_{ny}(k) &= \mathbf{f}_k^T\, \theta_k^{ny} + e_{ny}(k) \end{aligned} \tag{9}$$

This form means decomposition to a set of equations expressing the relation of individual current outputs to all appropriate inputs (current and delayed) and appropriate delayed outputs. Then, the identification can be realized formally as well as in SISO case.

The difference of both forms for MIMO systems becomes evident in initialization of identification and influence of evolution of identified parameters by a priori information.

Model identification

The sufficient and well known identification method is least square method [4]. In this paper will be briefly summed up in square-root form.

For definiteness, let us consider ARX model (7), where $\mathbf{e}(k)$ represents, in view of least squares, model error:

$$\mathbf{e}(k) = \mathbf{y}(k) - \boldsymbol{\vartheta}_k\, \mathbf{f}_k \tag{10}$$

The expression (10) do not represent enough equations for identification, since it is only n_y equations for determination of $n_y \times (n \cdot n_y + n \cdot n_u)$ parameters ϑ_k.

On the assumption, that the parameters are close to constants or they are varied only slightly during a real control process, then it is possible to write needful number of equations of errors with changeless vector of parameters ϑ_k

$$\mathbf{e}_k = \mathbf{y}_k - \mathbf{F}_k\, \vartheta_k^T = [\mathbf{F}_k \;\; \mathbf{y}_k] \begin{bmatrix} -\vartheta_k^T \\ \mathbf{I} \end{bmatrix} \tag{11}$$

where $\mathbf{F}_k$ is a square matrix of order $n \cdot n_y + n \cdot n_u$ composed of appropriate data vectors $\mathbf{f}_i^T$.

Then, the criterion for identification is

$$J_k = \mathbf{e}_k^T\, \mathbf{e}_k = [-\vartheta_k \;\; \mathbf{I}] \begin{bmatrix} \mathbf{F}_k^T \\ \mathbf{y}_k^T \end{bmatrix} [\mathbf{F}_k \;\; \mathbf{y}_k] \begin{bmatrix} -\vartheta_k^T \\ \mathbf{I} \end{bmatrix} \tag{12}$$

To minimize the criterion, it is sufficient to minimize only its square-root $\mathbf{J}$ as it follows from (12)

$$\min J_k = \|\mathbf{J}\|^2 = \left\| [\mathbf{F}_k \quad \mathbf{y}_k] \begin{bmatrix} -\vartheta_k^T \\ \mathbf{I} \end{bmatrix} \right\|^2 \tag{13}$$

The computationally effective minimization is provided by orthogonal-triangular decomposition (14) (e.g. house-holder algorithm [5]).

$$\mathbf{Q}[\mathbf{F}_k \quad \mathbf{y}_k] \begin{bmatrix} -\vartheta_k^T \\ \mathbf{I} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{c}_l \end{bmatrix} \tag{14}$$

which converts matrix $[\mathbf{F}_k \quad \mathbf{y}_k]$ to upper triangle (15):

$$\mathbf{Q}[\mathbf{F}_k \quad \mathbf{y}_k] = \mathbf{R} = \begin{bmatrix} \mathbf{R}_{PP} & \mathbf{R}_{PR} \\ & \mathbf{c}_l \end{bmatrix} \tag{15}$$

This matrix consists of sub-matrices partly corresponding to unknown parameters and partly to square-root $\mathbf{c}_l$ of loss of the criterion.

By considering sub-matrices related to unknown parameters, the following equation is obtained

$$-\mathbf{R}_{PP}\vartheta_k^T + \mathbf{R}_{PR} = \mathbf{0} \tag{16}$$

from which, the unknown parameters can be determined by backward substitution (due to triangular form of matrix $\mathbf{R}_{PP}$). This process is provided on-line in each time step with connecting refreshed data $\mathbf{f}_k$ and $\mathbf{y}(k)$ to current triangular matrix $\mathbf{R}$.

Thus, appropriate part of matrix $\mathbf{R}$ is restored to new upper triangular matrix $\mathbf{R}_{new}$.

$$\mathbf{Q}\begin{bmatrix} \mathbf{R}_{PP}^{prev} & \mathbf{R}_{PR}^{prev} \\ \mathbf{f}_k^T & \mathbf{y}^T(k) \end{bmatrix} = \mathbf{R}_{new} \tag{17}$$

Since the identification of parameters should not start from zeros, the initial filling of matrix $\mathbf{R}$ (a priori parameter setting) can be done as follows

$$\mathbf{R} = K\mathbf{I}_{(m+n_y)},\ approx.\ K = 10^{-6},\ m = n \cdot n_y + n \cdot n_u$$
$$\mathbf{R}_{(1:m,\ m+1:m+n_y)} = \mathbf{R}_{(1:m,\ 1:m)}\vartheta_0^T \ (= \mathbf{R}_{PR} = \mathbf{R}_{PP}\,\vartheta_0^T) \tag{18}$$

This selection (diagonal elements of $\mathbf{R}$) influences evolution of identified parameters. In case of parameters of SISO systems and in case parameterization (9) for MIMO systems these diagonal elements correspond to individual parameters. By setting of some element to zero means that appropriate parameter is fixed, i.e. keeps its initial value.

This property is useful e.g. for a priori setting of the dependency of individual outputs on inputs and other outputs; i.e. presence of the appropriate parameter or not. The parameterization in multivariate linear regression form has limited this property. However, it is less computationally demanding.

To increase the weight of the newest data, the exponential forgetting factor $fi_{(=0.9-1)}$ $\mathbf{R} = fi\,\mathbf{R}$ is useful [4]. It is realized after obtaining current parameters.

Design of predictive control

At design of predictive control, the composition of predictive equations is the most important. In this section, the emphasis is laid on specific reorganization of ARX model (7) [3]. Their result is state-space like model, which preserves Input-Output character simultaneously with transparency of state-space formulation applied to MIMO systems.

The reorganized model is used in the composition of equations of predictions. At the end of this section, the criterion minimization is briefly recapitulated.

Model reorganization

To compose equations of predictions from available ARX model (7), there are several possibilities how to do it. One of the possibilities is to express the equations directly from ARX model [2]. However, such way requires solution of Diofantic equation and storing previous values of inputs and outputs as in second possibility – utilization of 'pseudo state-space' model.

The second possibility obtaining suitable model will be explained. Let us arise from ARX model in one-ahead predictive style.

$$\mathbf{y}(k+1) = \sum_{i=1}^{n}\mathbf{B}_i\mathbf{u}(k-i+1) - \sum_{i=1}^{n}\mathbf{A}_i\mathbf{y}(k-i+1) \tag{19}$$

Then, the suitable form can be structured as follows

$$\begin{bmatrix} \mathbf{y}(k-n+2) \\ \vdots \\ \mathbf{y}(k) \\ \mathbf{y}(k+1) \end{bmatrix} = \begin{bmatrix} 0 & \mathbf{I} & & \\ \vdots & & \ddots & \\ 0 & & & \mathbf{I} \\ -\mathbf{A}_n & & \cdots & -\mathbf{A}_1 \end{bmatrix} \begin{bmatrix} \mathbf{y}(k-n+1) \\ \vdots \\ \mathbf{y}(k-1) \\ \mathbf{y}(k) \end{bmatrix}$$
$$+ \begin{bmatrix} \mathbf{0} & \cdots & \mathbf{0} \\ \vdots & & \vdots \\ \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{B}_n & \cdots & \mathbf{B}_1 \end{bmatrix} \begin{bmatrix} \mathbf{u}(k-n+1) \\ \vdots \\ \mathbf{u}(k-1) \\ \mathbf{u}(k) \end{bmatrix} \Rightarrow$$

$$\begin{bmatrix} y_{(k-n+2)} \\ \vdots \\ y(k+1) \end{bmatrix} = \mathbf{A} \begin{bmatrix} y_{(k-n+1)} \\ \vdots \\ y(k) \end{bmatrix} + \mathbf{B_0} \begin{bmatrix} u_{(k-n+1)} \\ \vdots \\ u(k) \end{bmatrix} \tag{20}$$
$$\mathbf{x}(k+1) = \mathbf{A}\quad \mathbf{x}(k) + \mathbf{B_0}\quad \mathbf{u}(k)$$

$$\mathbf{y}(k) = [0 \quad \cdots \quad 0 \quad 1]\,\mathbf{x}(k)$$
$$\mathbf{y}(k) = \quad \mathbf{C} \quad \mathbf{x}(k) \tag{21}$$

State-space model (equations (20) and (21)) has two pseudo state-space matrices $\mathbf{A}$ and $\mathbf{B_0}$. Their dimensions are similar as in the use of usual (pure) state-space models. Subscript of matrix $\mathbf{B_0}$ will be significant in next subsection at the composition of equations of predictions.

Equations of predictions

Usual composition of equations of predictions follows from ordinary state-space model

$$\begin{aligned} \mathbf{x}(k+1) &= \mathbf{A}_k\,\mathbf{x}(k) + \mathbf{B}_k\,\mathbf{u}(k) \\ \mathbf{y}(k) &= \mathbf{C}_k\,\mathbf{x}(k) \end{aligned} \tag{22}$$

which is a model with minimal state. It maps interval of one sampling period.

Principle of the equations is expression (prediction) of future values of outputs $\mathbf{y}$ from current measured state $\mathbf{x}(k)$ as follows:

$$\begin{aligned}
\hat{\mathbf{x}}(k+1) &= \mathbf{A}\,\mathbf{x}(k+1) + \mathbf{B}\mathbf{u}(k)\\
\hat{\mathbf{y}}(k+1) &= \mathbf{C}\mathbf{A}\,\mathbf{x}(k+1) + \mathbf{C}\mathbf{B}\mathbf{u}(k)\\
&\vdots\\
\hat{\mathbf{x}}(k+N) &= \mathbf{A}^{N}\mathbf{x}(k+1) + \mathbf{A}^{N-1}\mathbf{B}\mathbf{u}(k)\cdots + \mathbf{B}\mathbf{u}(k+N-1)\\
\hat{\mathbf{y}}(k+N) &= \mathbf{C}\mathbf{A}^{N}\mathbf{x}(k+1) + \mathbf{C}\mathbf{A}^{N-1}\mathbf{B}\mathbf{u}(k)\cdots + \mathbf{C}\mathbf{B}\mathbf{u}(k+N-1)
\end{aligned} \tag{23}$$

Equation (23) can be condensed in matrix notation

$$\hat{\mathbf{y}} = \mathbf{f} + \mathbf{G}\,\mathbf{u},\quad \hat{\mathbf{y}} = [\hat{\mathbf{y}}(k+1)\ \hat{\mathbf{y}}(k+2)\cdots\hat{\mathbf{y}}(k+N)]^{T}$$
$$\mathbf{u} = [\mathbf{u}(k)\ \mathbf{u}(k+1)\cdots\mathbf{u}(k+N-1)]^{T} \tag{24}$$

$$\mathbf{f} = \begin{bmatrix}\mathbf{CA}\\ \vdots\\ \mathbf{CA}^{N}\end{bmatrix}\mathbf{x}(k),\quad \mathbf{G} = \begin{bmatrix}\mathbf{C} & \mathbf{B} & \\ \vdots & & \ddots\\ \mathbf{CA}^{N-1}\mathbf{B} & \cdots & \mathbf{CB}\end{bmatrix} \tag{25}$$

Considering the state-space model (20) without (21), the equations can be composed again recursively as it was indicated in (23).

$$\begin{aligned}
»\hat{\mathbf{x}}(k+1) &= \mathbf{A}\mathbf{x}(k) + \mathbf{B}_0\mathbf{u}(k)\\
\hat{\mathbf{x}}(k+2) &= \mathbf{A}^2\mathbf{x}(k) + \mathbf{A}\mathbf{B}_0\mathbf{u}(k) + \mathbf{B}_0\overline{\mathbf{u}}(k+1)\\
»\hat{\mathbf{x}}(k+2) &= \mathbf{A}^2\mathbf{x}(k) + \underbrace{([\mathbf{A}\mathbf{B}_0\,\mathbf{0}] + [\mathbf{0}\,\mathbf{B}_0])}_{\mathbf{B}_1}\mathbf{u}(k+1)\\
\hat{\mathbf{x}}(k+3) &= \mathbf{A}^3\mathbf{x}(k) + \mathbf{A}\mathbf{B}_1\mathbf{u}(k+1) + \mathbf{B}_0\overline{\mathbf{u}}(k+2)\\
»\hat{\mathbf{x}}(k+3) &= \mathbf{A}^3\mathbf{x}(k) + \underbrace{([\mathbf{A}\mathbf{B}_1\,\mathbf{0}] + [\mathbf{0}\,\mathbf{0}\,\mathbf{B}_0])}_{\mathbf{B}_2}\mathbf{u}(k+2)\\
&\ddots\\
\hat{\mathbf{x}}(k+N) &= \mathbf{A}^{N}\mathbf{x}(k) + \mathbf{A}\mathbf{B}_{N-2}\,\mathbf{u}(k+N-2) + \mathbf{B}_0\overline{\mathbf{u}}(k+N-1)\\
»\hat{\mathbf{x}}(k+N) &= \mathbf{A}^{N}\mathbf{x}(k) + \underbrace{([\mathbf{A}\mathbf{B}_{N-2}\,\mathbf{0}] + [\mathbf{0}_{n,N-1}\,\mathbf{B}_0])}_{\mathbf{B}_{N-1}}\mathbf{u}(k+N-1)
\end{aligned} \tag{26}$$

The support notation has the following meaning

$$\begin{aligned}
\overline{\mathbf{u}}(k+1) &= [u(k-n+2)\cdots u(k+1)]^{T}\\
\mathbf{u}(k+1) &= [u(k-n+1)\ u(k-n+2)\cdots u(k)\ u(k+1)]^{T}\\
&= [\mathbf{u}^{T}(k),\ u(k+1)]^{T}\\
&\vdots\\
\overline{\mathbf{u}}(k+N-1) &= [u(k-n+N)\cdots u(k+N-1)]^{T}\\
\mathbf{u}(k+N-1) &= [u(k-n+1)\cdots u(k+N-2)\ u(k+N-1)]^{T}\\
&= [\mathbf{u}^{T}(k+N-2),\ u(k+N-1)]^{T}
\end{aligned} \tag{27}$$

The equations of predictions (26) considering (21) can be also rewritten to appropriate matrix notation:

$$\begin{bmatrix}\hat{\mathbf{y}}(k+1)\\ \vdots\\ \hat{\mathbf{y}}(k+N)\end{bmatrix} = \begin{bmatrix}\mathbf{CA}\\ \vdots\\ \mathbf{CA}^{N}\end{bmatrix}\begin{bmatrix}\mathbf{y}(k-n+1)\\ \vdots\\ \mathbf{y}(k)\end{bmatrix} + \begin{bmatrix}\mathbf{C}\mathbf{B}_0 & \cdots & \mathbf{0}\\ & \ddots & \vdots\\ & & \mathbf{C}\,\mathbf{B}_{N-1}\end{bmatrix}\begin{bmatrix}\mathbf{u}(k-n+1)\\ \vdots\\ \mathbf{u}(k+N-1)\end{bmatrix}$$

$$\begin{aligned}
\hat{\mathbf{y}} &= \mathbf{f} + \overline{\mathbf{G}}\,\mathbf{u}_{k+N+1}\\
\hat{\mathbf{y}} &= \mathbf{f} + \overline{\mathbf{G}}_{(:,1:n-1)}\begin{bmatrix}\mathbf{u}(k-n+1)\\ \vdots\\ \mathbf{u}(k-1)\end{bmatrix} + \overline{\mathbf{G}}_{(:,n:N-1)}\begin{bmatrix}\mathbf{u}(k)\\ \vdots\\ \mathbf{u}(k+N-1)\end{bmatrix}\\
\hat{\mathbf{y}} &= \overline{\mathbf{f}} + \mathbf{G}\,\mathbf{u}
\end{aligned} \tag{28}$$

representing: free response + forced response.

Such composed equations of predictions have the same dimension as the equations (24), which are based on state-space model with minimal state.

Control action computation

The control actions are obtained by minimization of quadratic criterion

$$J_k = \sum_{j=No+1}^{N}\left\|(\hat{\mathbf{y}}(k+j) - \mathbf{w}(k+j))\mathbf{Q}_y\right\|^2 + \sum_{j=1}^{Nu}\left\|\mathbf{u}(k+j-1)\mathbf{Q}_u\right\|^2 \tag{29}$$

where N, No and Nu are horizons; $\mathbf{Q}_y$ and $\mathbf{Q}_u$ are penalizations; and $\mathbf{w}(k+j)$ are desired values. That criterion can be again condensed in matrix notation

$$J_k = [(\hat{\mathbf{y}} - \mathbf{w})^{T}\ \mathbf{u}^{T}]\begin{bmatrix}\mathbf{Qy} & \mathbf{0}\\ \mathbf{0} & \mathbf{Qu}\end{bmatrix}^{T}\begin{bmatrix}\mathbf{Qy} & \mathbf{0}\\ \mathbf{0} & \mathbf{Qu}\end{bmatrix}\begin{bmatrix}\hat{\mathbf{y}} - \mathbf{w}\\ \mathbf{u}\end{bmatrix} \tag{30}$$

from which, only one part (square-root) is sufficient to minimize.

$$\mathbf{J} = \begin{bmatrix}\mathbf{Qy} & \mathbf{0}\\ \mathbf{0} & \mathbf{Qu}\end{bmatrix}\begin{bmatrix}\hat{\mathbf{y}} - \mathbf{w}\\ \mathbf{u}\end{bmatrix} = \begin{bmatrix}\mathbf{Qy\,G}\\ \mathbf{Qu}\end{bmatrix}\mathbf{u} - \begin{bmatrix}\mathbf{Qy}\,(\mathbf{w} - \overline{\mathbf{f}})\\ \mathbf{0}\end{bmatrix} \tag{31}$$

The minimization leads to the solution of algebraic equations for unknown control actions

$$\begin{bmatrix}\mathbf{Qy\,G}\\ \mathbf{Qu}\end{bmatrix}\mathbf{u} - \begin{bmatrix}\mathbf{Qy}\,(\mathbf{w} - \overline{\mathbf{f}})\\ \mathbf{0}\end{bmatrix} = \mathbf{0}$$

$$\mathbf{A}\quad\mathbf{u} - \quad\mathbf{b}\quad = \mathbf{0} \tag{32}$$

This system of algebraic equations can be effective evaluated again by orthogonal-triangular decomposition [5].

$$\begin{aligned}
\mathbf{A}\ \mathbf{u} &= \mathbf{b}\qquad /\times\mathbf{Q}^{T}\\
\mathbf{R}_1\mathbf{u} &= \mathbf{c}_1
\end{aligned} \tag{33}$$

Orthogonal matrix $\mathbf{Q}^{T}$ transforms the matrix $\mathbf{A}$ to upper triangle $\mathbf{R}_1$. Unknown control actions from the algebraic system (33) can be determined by backward substitution.

From obtained vector $\mathbf{u}$, which represents designed control actions for whole horizon N, only first appropriate actions are really applied to controlled system. This process is repeated in every time step.

Finally, let us discuss the parameter tuning of predictive control in case of MIMO systems. The selection of horizons corresponds to system order and rate of changes of desired values. It is more or less straightforward choice. The tuning of elements of penalization matrices depends on character (case) of individual inputs and outputs. When the matrices $\mathbf{Q}_y$ and $\mathbf{Q}_u$ are generally defined as follows:

$$\mathbf{Q}_y = diag([{}^{y}\lambda_1,\cdots,{}^{y}\lambda_{ny}]),\ \mathbf{Q}_u = diag([{}^{u}\lambda_1,\cdots,{}^{u}\lambda_{nu}]) \tag{34}$$

Then, the values of individual elements may have properties indicated in Tab. 1.

case \ penalizations	$\mathbf{Q}_u$	$\mathbf{Q}_y$
I similar + O sim.	${}^{u}\lambda_i = const.$	${}^{y}\lambda_i = 1$
I sim. + O different	${}^{u}\lambda_i = const.$	different ${}^{y}\lambda_i$
I diff. + O sim.	different ${}^{u}\lambda_i$	${}^{y}\lambda_i = 1$
I diff. + O diff.	different ${}^{u}\lambda_i$	different ${}^{y}\lambda_i$

Tab.1 Selection of penalization matrices (I = Inputs, O = Outputs)

Examples

Presented adaptive predictive control was tested on laboratory model ‘ball on rod’ and simplified model of helicopter. The control was implemented under MATLAB-Simulink environment using measuring card and Real-Time toolbox.

The model ‘ball on rod’ represents SISO system. The task is to stabilize the ball in defined position.

Fig.1 Laboratory model ‘ball on rod’

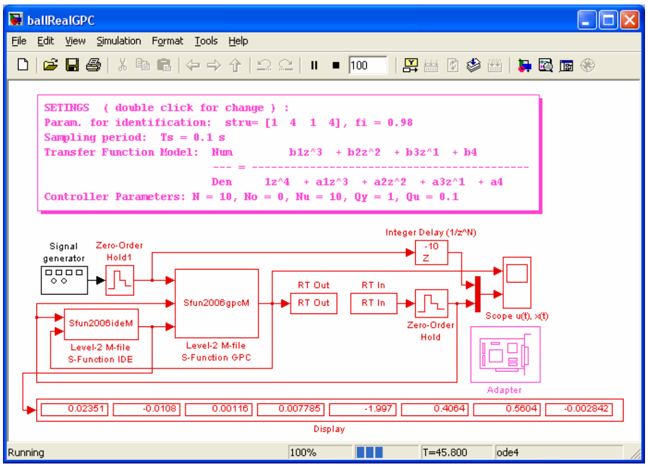

Fig.2 Appropriate Simulink scheme of adaptive predictive control

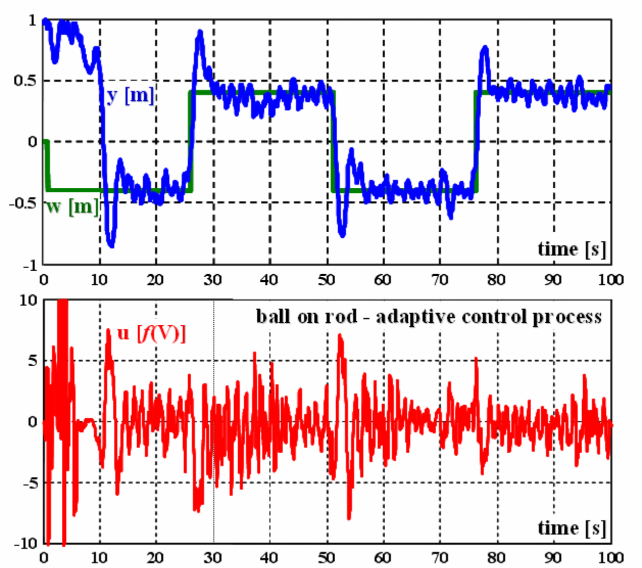

Fig.3 Time histories of ball position and corresponding control actions

The following model of helicopter (Fig. 4) represents MIMO system with two inputs and two outputs, which was controlled by circuit shown in Fig.5.

Fig.4 Model ‘helicopter’ (Humusoft Co.)

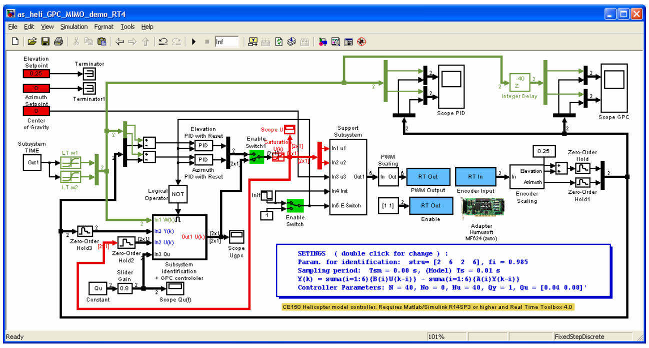

Fig.5 Simulink scheme with predictive controller

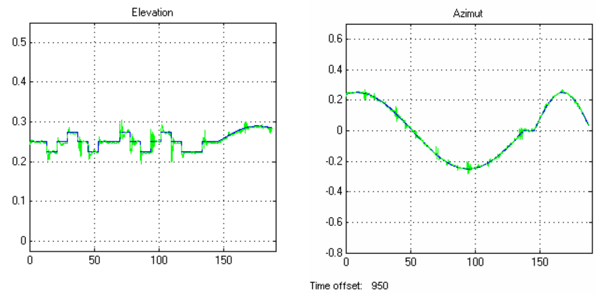

Fig.6 Time history of elevation and azimuth

The desired motion (blue curves) was combination of sin and rectangular signals. During the real-time control process, it is perceptible, that the identification was not adequately excited. Therefore the parameters were changed slower than reality. It caused degradation of the control, which lost adequate model. However, improper control actions caused exciting of identification and it improved the identification of model parameters and such way the control was corrected again.

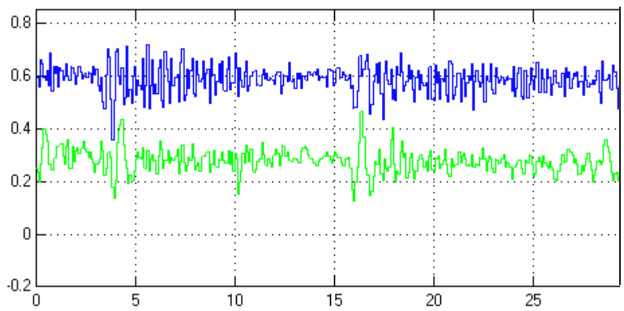

Fig.7 Time history of one selected segment of control actions referred to Fig. 6

Conclusion

In the paper, the adaptive predictive control was introduced. Multi-Input Multi-Output ARX model was used for composition of equations of predictions in specific state-space like formulation. This formulation requires comparable dimensions of matrices as usual state-space formulation; i.e. it has similar computational demands. Different condensed forms of ARX model were demonstrated. Finally, the tuning of the parameters of the predictive controller applied to MIMO systems was briefly outlined.

Acknowledgements

This research is supported by GAČR 102/06/P275 "Model-based Control of Mechatronic Systems for Robotics" and GAČR 102/05/0271 "Models of Predictive Control, Algorithms and Implementation".

References

[1] ORDYS, A., CLARKE, D.: A State - Space Description for GPC Controllers. *Int. J. Systems SCI.*, Vol. 24, No. 9, pp. 1727 – 1744, 1993.

[2] MACIEJOWSKI, J. M.: *Predictive Control with Constrains*. Prentice Hall, New York, 2001.

[3] BELDA, K., BÖHM, J.: Adaptive Predictive Control for Mechatronic Systems. *WSEAS Trans. on Systems*. Vol. 5, Issue 8, pp. 1830-1837, 2006.

[4] KULHAVÝ, R.: Restricted Exponential Forgetting in Real-time Identification. *Automatica,* Vol. 23, No. 5, pp. 589-600, 1987.

[5] GOLUB, H. G., Van, Ch. F. L.: *Matrix Computations*. The Johns Hopkins Univ. Press, 1989.

[6] BELDA, K. et al.: GPC pages http://as.utia.cz/asc/, 2005.

Ing. Květoslav Belda, Ph.D.
Ing. Josef Böhm, CSc.

Institute of Information Theory and Automation
Department of Adaptive Systems
Pod vodárenskou věží 4
182 08 Praha 8
Czech Republic
E-mail: belda@utia.cas.cz, bohm@utia.cas.cz