FUZZY MODEL PREDICTIVE CONTROL OF A MIMO SYSTEM

M. Petrov, S. Ahmed, A. Taneva, Y. Todorov

Technical University Sofia, branch Plovdiv, Plovdiv 4000, 25 Tsanko Dustabanov str., Bulgaria
e-mail: mpetrov@tu-plovdiv.bg,sevil.ahmed@gmail.com,altaneva@tu-plovdiv.bg, yancho.todorov@ieee.org

Abstract: In this paper Nonlinear Model Predictive Control (NMPC) is studied as a more applicable approach for optimal control of multivariable processes. A state-space representation of a Takagi-Sugeno type fuzzy-neural model is proposed as a predictive model. This type of model ensures easier description and direct computation of the gradient control vector during the predictive optimization task. The identification procedure relies on a two-step training algorithm, which is known in field of artificial neural networks. The proposed Fuzzy NMPC approach is studied by experimental simulations in Matlab/Simulink® environment in order to control the liquid levels in a multi tank system. The simulation results demonstrate that the main process variables have a good performance and the process control quality is satisfied.

Key words: Model Predictive Control, Optimal Control, Fuzzy Control, MIMO system

INTRODUCTION

Model Predictive Control is a model-based control strategy that obtains the optimal control action sequence, solving an optimization problem at each sampling time. Achieving the desired plant behaviour strongly depends on applied modelling techniques. The model accuracy is important in order to provide an efficient and adequate control action. In this paper an approach of Fuzzy-Neural Model Predictive Control with significantly simplified optimization procedure is proposed.

A state-space representation of a Takagi-Sugeno type fuzzy-neural model [1,4] is applied to provide an accurate predictive model of the studied nonlinear system. The fuzzy-neural identification procedure relies on an efficient training algorithm, which is known in field of artificial neural networks.

A simplified calculation method is proposed to solve the optimization problem of MPC. The proposed approach solves an unconstrained MPC problem avoiding the necessity to inverse predictive dynamic matrices at each sampling time [5]. Therefore, the computational burden is decreased, which makes the strategy, situated in this paper, attractive for real-time implementation for control of nonlinear industrial processes [7, 8].

The proposed approach is studied by experimental simulations in Matlab/Simulink® environment in order to control the levels in a multi tank system [3]. The case study is suitable to show how the proposed NMPC algorithm handle with multivariable processes control problem.

FUZZY MODEL PREDICTIVE MODEL

The Takagi-Sugeno fuzzy-neural models are powerful modelling tools for a wide class of nonlinear systems. Fuzzy reasoning is capable of handling uncertain and imprecise information while neural networks can learn from samples. Fuzzy-neural networks combine the advantages of both artificial intelligent techniques and incorporate them in adaptive features as well. The main contribution of using fuzzy-neural models in MPC strategy is their adaptive modelling capabilities based on a real-time learning algorithm. The importance of the used in MPC strategy models and their adaptive characteristics is obvious. The accuracy of the model determines the accuracy of the control action. The proposed fuzzy-neural model is implemented in a classical NMPC scheme (Fig. 1) as a predictor [2].

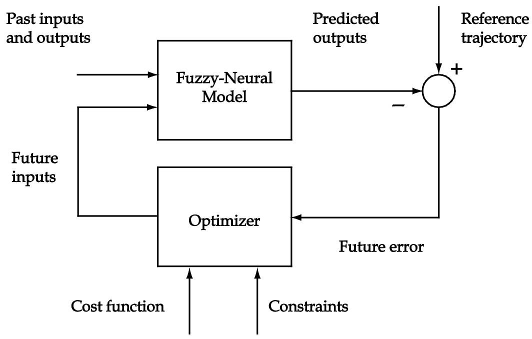

Fig. 1. Basic scheme of model predictive control

In this paper a nonlinear discrete time state-space implementation is considered to represent the system dynamic:

$$\begin{aligned} x(k+1) &= f_X(x(k),u(k)) \\ y(k) &= f_Y(x(k),u(k)) \end{aligned} \quad (1)$$

where $x(k) \in \Re^n$, $u(k) \in \Re^m$ and $y(k) \in \Re^q$ are state, control and output variables of the system, respectively. The unknown nonlinear functions f_x and f_y can be approximated by the Takagi-Sugeno type fuzzy rules in the following form:

$$R_l : \; if \quad z_1(k) \; is \; M_{l1} \quad and \quad \dots \quad and \quad z_i(k) \; is \; M_{li}$$
$$and \dots z_p(k) \; is \; M_{lp} \quad then \quad \begin{cases} x_l(k+1) = A_l x(k) + B_l u(k) \\ y_l(k) = C_l x(k) + D_l u(k) \end{cases} \quad (2)$$

where R_l is the l^{-th} rule of the rule base. Each rule is represented by an *if-then* conception, where $z_i(k)$ is an i^{-th} linguistic input variable and M_{li} is a membership function defined by a fuzzy set of the universe of discourse of the input z_i. Note that the input regression vector $\mathbf{z}(k) \in \Re^p$ in this paper contains the system states and inputs. A state-space implementation is used in the consequent part of R_l (2), where $A_l \in \Re^{n\times n}$, $B_l \in \Re^{n\times m}$, $C_l \in \Re^{q\times n}$ and $D_l \in \Re^{q\times m}$ are the state-space matrices of the model [1]. The state in the next sampling time $\hat{x}(k+1)$ and the system output $\hat{y}(k)$ can be obtained by taking the weighted sum of the activated fuzzy rules, using

$$\begin{aligned}\hat{x}(k+1) &= \sum_{l=1}^{L} \bar{\mu}_{yl}(k)(A_l x(k) + B_l u(k)) \\ \hat{y}(k) &= \sum_{l=1}^{L} \bar{\mu}_{yl}(k)(C_l \hat{x}(k) + D_l u(k))\end{aligned} \tag{3}$$

On the other hand, the state-space matrices A, B, C, and D for the global state-space plant model could be calculated as a weighted sum of the local matrices A_l, B_l, C_l, and D_l from the activated fuzzy rules:

$$\begin{aligned} A(k) &= \sum_{l=1}^{L} A_l \bar{\mu}_{yl}(k) \qquad \mathrm{B}(k) = \sum_{l=1}^{L} B_l \bar{\mu}_{yl}(k) \\ C(k) &= \sum_{l=1}^{L} C_l \bar{\mu}_{yl}(k) \qquad \mathrm{D}(k) = \sum_{l=1}^{L} D_l \bar{\mu}_{yl}(k)\end{aligned} \tag{4}$$

where $\bar{\mu}_{yl} = \mu_{yl} \Big/ \sum_{l=1}^{L} \mu_{yl}$ is the normalized value of the membership function degree μ_{yl} upon the l^{-th} activated fuzzy rule and L is the number of the activated rules at the moment k. Fuzzy implication in the l^{-th} rule (2) can be realized by means of a product composition

$$\mu_{yl} = \prod_{i=1}^{p} \mu_{ij} \tag{5}$$

where μ_{ij} specifies the membership degree upon the activated j^{-th} fuzzy set of the corresponded i^{-th} input signal. The quantity of μ_{ij} is calculated according to the type of chosen membership functions. In this paper a Gaussian type membership function with following statement is used:

$$\mu_{ij}(z_i) = \exp - \frac{(z_i - c_{Gij})^2}{2\sigma_{ij}^{\ 2}} \tag{6}$$

where z_i is the current input value of the i^{-th} model input, c_{Gij} is the centre and σ_{ij} is the standard deviation of the j^{-th} membership function ($j=1, 2, .., s$).
The identification procedure of the fuzzy-neural predictive model is made according to the algorithm described in [1].

MODEL PREDICTIVE OPTIMIZATION ALGORITHM

Model predictive control is based on real-time optimization of a cost function. The typical for Generalized Predictive Control (GPC) cost function $J(k)$ [2] is

$$J(k) = \sum_{i=H_w}^{H_p+H_w-1} \left\| \hat{y}(k+i) - r(k+i) \right\|_{\tilde{Q}}^2 + \sum_{i=0}^{H_u-1} \left\| \Delta\hat{u}(k+i) \right\|_{\tilde{R}}^2 \tag{7}$$

where $\hat{y}(k)$, $r(k)$ and $\Delta\hat{u}(k)$ are the predicted outputs, the reference trajectories, and the predicted control increments at time k, respectively. The length of the prediction horizon is H_p, and the first sample to be included in the horizon is H_w. The control horizon is given by H_u. $\tilde{Q} \geq 0$ and $\tilde{R} > 0$ are weighting matrices assumed to be constant over the prediction horizon.
In the general formulation of the model predictive control, the discrete-time state-space equations of the system are used to estimate the future states of the system [4,5]

$$\begin{aligned}\hat{x}(k+1) &= Ax(k) + Bu(k-1) + B\Delta\hat{u}(k) \\ \hat{y}(k) &= C\hat{x}(k) + Du(k-1) + D\Delta\hat{u}(k)\end{aligned} \tag{8}$$

The recurrent equation for the output predictions $\hat{y}(k+j\,|\,k)$, where $j_p = 1, 2, ..., H_p - 1$, is in the next form:

$$\begin{aligned}\hat{y}(k+j_p|\,k) &= CA^{j_p} x(k) + \left(C \sum_{i=0}^{j_p-1} A^i B + D \right) u(k-1) + \\ &+ \sum_{i=0}^{H_u-1} \left\{ \left(C \sum_{j=0, i<j_p}^{j_p-i-1} A^j B + D \right) \Delta u(k+i\,|\,k) \right\}\end{aligned} \tag{9}$$

Taking in account equation (9), the cost function (7) could be rewritten in a matrix form

$$J(k) = \left\| Y(k) - T(k) \right\|_Q^2 + \left\| \Delta U(k) \right\|_R^2 \tag{10}$$

where

$$Y(k) = \begin{bmatrix} \hat{y}(k) \\ \vdots \\ \hat{y}(k+H_p-1) \end{bmatrix} T(k) = \begin{bmatrix} r(k) \\ \vdots \\ r(k+H_p-1) \end{bmatrix} \Delta U(k) = \begin{bmatrix} \Delta\hat{u}(k) \\ \vdots \\ \Delta\hat{u}(k+H_u-1) \end{bmatrix}$$

$$Q = \begin{bmatrix} \tilde{Q}(1) & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \tilde{Q}(H_p) \end{bmatrix} \qquad R = \begin{bmatrix} \tilde{R}(1) & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \tilde{R}(H_u) \end{bmatrix}$$

By deriving the prediction expression (9), it can be written

$$Y(k) \ = \ \Psi x(k) \ + \ \Gamma u(k-1) \ + \ \Theta\Delta U(k) \tag{11}$$

where

$$\Psi = \begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{Hp-1} \end{bmatrix} \quad \Gamma = \begin{bmatrix} D \\ CB+D \\ CAB+CB+D \\ \vdots \\ C\sum_{i=0}^{Hp-2} A^i B + D \end{bmatrix}$$

$$\Theta = \begin{bmatrix} D & & 0 \\ CB+D & D & 0 \\ \vdots & \ddots & \vdots \\ C\sum_{i=0}^{Hu-2} A^i B + D & \cdots & D \\ \vdots & \ddots & \vdots \\ C\sum_{i=0}^{Hp-2} A^i B + D & \cdots & C\sum_{i=0}^{Hp-Hu-1} A^i B + D \end{bmatrix}.$$

Also, let

$$E(k) \ = \ T(k) - \Psi x(k) - \Gamma u(k-1) \tag{12}$$

This vector can be thought as a *tracking error*, in the sense that it is the difference between the future target trajectory and the *free response* of the system, namely the response that would occur over the prediction horizon if no input changes were made, i.e. $\Delta U(k)=0$. Hence, the quantity of the so called *free response* $F(k)$ is defined as follows

$$F(k) \ = \ \Psi x(k) \ + \Gamma u(k-1) \tag{13}$$

In this paper, the study is focused on the optimization problem of the unconstrained nonlinear predictive control with the quadratic cost function (10). The section presents an analytical solution of the problem where the information given by the obtained fuzzy-neural model is used to solve the problem. The unconstrained optimization problem can be formulated in a matrix form. First, the predictor can be constructed as follows

$$Y(k) = \Theta\Delta U(k) + F(k) \tag{14}$$

The second element of the matrix equation (14) does not include the predicted values of the control action and represents the so called *free response*. Second, the cost function (7) can be rewritten as

$$J(\Delta U) = (T - Y)^T Q(T - Y) + \Delta U^T R \Delta U \qquad (15)$$

Hence, substituting the predictive model (14) into the expression (15) the cost function of the model predictive optimization problem can be specified as follows:

$$J(\Delta U) = \Delta U^T(\Theta^T Q\Theta + R)\Delta U^T + 2(T - F)^T Q\Theta\Delta U + (T - F)^T Q(T - F) \qquad (16)$$

Minimization of the function $J(\Delta U)$ can be obtained by calculating the input sequence ΔU so that the derivatives $\partial J/\partial \Delta U = 0$. Then the optimal sequence ΔU^* is

$$\Delta U^* = (\Theta^T Q\Theta + R)^{-1}\Theta^T Q(T - F) \qquad (17)$$

The input applied to the controlled plant at time k is taken from the first element $\Delta\hat{u}^*(k)$ of the vector ΔU^*

$$u(k) = u(k-1) + \Delta\hat{u}^*(k) \qquad (18)$$

On the other hand, the predictive task can be solved consequently, using a system of equations, avoiding the necessity to inverse the gain matrix in (17) at each sampling time k. Applying this method, minimization of the GPC-criterion (7) is based on a calculation of the gradient vector of the criterion cost function J at the moment k subject to the predicted control actions:

$$\nabla J(k) = \left[\frac{\partial J(k)}{\partial \Delta\hat{u}(k)}, \ldots, \frac{\partial J(k)}{\partial \Delta\hat{u}(k + H_u - 1)}\right]^T \qquad (19)$$

Each element of this gradient vector (19) can be calculated using the following derivative matrix equation:

$$\frac{\partial J(k)}{\partial \Delta U(k)} = \left[-2\left[T(k) - Y(k)\right]^T Q \frac{\partial Y(k)}{\partial \Delta U(k)} + 2\Delta U(k)^T R \frac{\partial \Delta U(k)}{\partial \Delta U(k)}\right] \qquad (20)$$

From the above expression (20) it can be seen that it is necessary to obtain two groups of partial derivatives. The first group of derivatives in (20) have the following matrix form:

$$\frac{\partial Y(k)}{\partial \Delta U(k)} = \begin{bmatrix} \dfrac{\partial \hat{y}(k + H_w)}{\partial \Delta\hat{u}(k)} & \cdots & \dfrac{\partial \hat{y}(k + H_w)}{\partial \Delta\hat{u}(k + H_u - 1)} \\ \vdots & & \vdots \\ \dfrac{\partial \hat{y}(k + H_p + H_w - 1)}{\partial \Delta\hat{u}(k)} & \cdots & \dfrac{\partial \hat{y}(k + H_p + H_w - 1)}{\partial \Delta\hat{u}(k + H_u - 1)} \end{bmatrix} \qquad (21)$$

For simplicity of calculation let assume that H_w=0 (7). Then each element of the matrix (21) is calculated by the expressed equations according to the Takagi-Sugeno rules consequents in (2).

The second group partial derivatives in (20) has the following matrix form:

$$\frac{\partial \Delta U(k)}{\partial \Delta U(k)} = \begin{bmatrix} \dfrac{\partial \Delta\hat{u}(k)}{\partial \Delta\hat{u}(k)} & \cdots & \dfrac{\partial \Delta\hat{u}(k)}{\partial \Delta u(k + H_u - 1)} \\ \vdots & \ddots & \vdots \\ \dfrac{\partial \Delta\hat{u}(k + H_u - 1)}{\partial \Delta\hat{u}(k)} & \cdots & \dfrac{\partial \Delta\hat{u}(k + H_u - 1)}{\partial \Delta\hat{u}(k + H_u - 1)} \end{bmatrix} \qquad (22)$$

Since $\Delta\hat{u}(k) = u(k) - u(k-1)$, the matrix (22) is

$$\frac{\partial \Delta U(k)}{\partial \Delta U(k)} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ -1 & 1 & & \\ \vdots & -1 & \ddots & \vdots \\ 1 & \vdots & \ddots & 1 \\ -1 & 1 & \cdots & -1 \end{bmatrix} \qquad (23)$$

Each element of the gradient-vector (19) could be obtained by the following recurrent form [6]:

$$\frac{\partial J(k)}{\partial \Delta\hat{u}(k + j - 1)} = 2\sum_{i=1}^{H_p} \hat{e}(k+i)\tilde{Q}(i)\frac{\partial \hat{y}(k+i)}{\partial \Delta\hat{u}(k + j - 1)} + 2\sum_{i=j}^{H_u} \tilde{R}(i)(-1)^{(i-j)}\Delta\hat{u}(k + i - 1) = 0 \qquad (24)$$

where $j=1,2,\ldots,H_u$ and $\hat{e}(k+i) = r(k+i) - \hat{y}(k+i)$ is the predicted system error.

According to (24) the last control action $u(k+H_u-1)$ has to be calculated first. Then the procedure continues with calculation of the previous control action $u(k+H_u-2)$ and this way consequently to calculate the whole number of the control actions over the horizon H_u. The order of the calculations is important since the calculations should consist of known quantities. After that, only the first control action $u(k)$ will be used at the moment k to the input of the controlled process. The software implementation of these computations is very easy and it is in accordance to the next recurrent expression ($j=1,2,\ldots,H_u$):

$$\Delta\hat{u}(k + j - 1) = \Delta\hat{u}(k + j) + \tilde{R}(j)^{-1}\sum_{i=1}^{H_p} \hat{e}(k+i)\tilde{Q}(i)\frac{\partial \hat{y}(k+i)}{\partial \Delta\hat{u}(k + j - 1)} \qquad (25)$$

The proposed unconstrained predictive control algorithm could be summarized in the following steps.

Step 0: Initial identification of the Takagi-Sugeno fuzzy-neural predictive model;
Step 1: Start the algorithm at the sample k with the initial parameters;
Step 2: Calculate the predicted model output $\hat{y}(k+j)$ using the tuned fuzzy-neural model (2);
Step 3: Calculate the derivatives for the matrix (21);
Step 4: Calculate predicted control actions according to (25) and update the sequence;
Step 5: Apply the first optimal control action $u(k)$;
Step 6: Modify the model parameters into the rule (3) and update them for the next step 2.

DESCRIPTION OF THE MULTI TANK SYSTEM

The case study is implemented in MATLAB&Simulink® environment with Inteco® Multi tank system. The Inteco® Multi tank System (Fig. 2) comprises from three separate tanks fitted with drain valves [3]. The top (first) tank has a constant cross section, while others are conical or spherical, so they are with variable cross sections. This causes the main nonlinearities in the system.

A variable speed pump is used to fill the upper tank. The liquid outflows the tanks due to gravity. The tank valves act as flow resistors C_1, C_2, C_3. The area ratio of the valves is controlled and can be used to vary the outflow characteristic. Each tank is equipped with a level sensor PS_1, PS_2, PS_3 based on hydraulic pressure measurement.

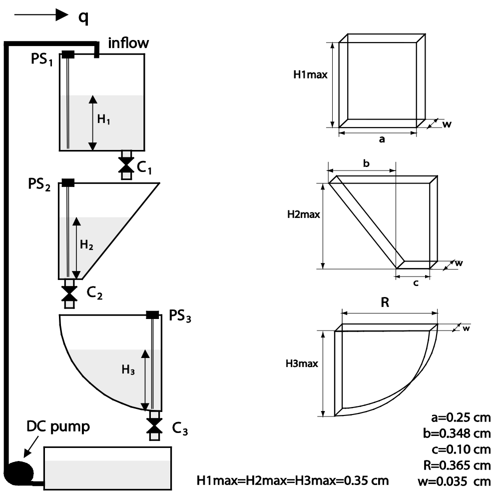

Fig. 2. Controlled laboratory multi tank system

Liquid levels H_1, H_2, H_3 in the tanks are the state variables of the system. The Inteco Multi Tank system has four controlled inputs: liquid inflow q and valves settings C_1, C_2, C_3.

SIMULATION RESULTS

The proposed unconstrained MPC algorithm with the Takagi-Sugeno fuzzy-neural model as a predictor has been applied to the level control problem. The experiments have been implemented with the following parameters: prediction horizon $H_p=10$, First included sample of the prediction horizon $H_w=1$, control horizon $H_u=3$, time of simulation *600 s*, sample time $T_s=1$. The weighting matrices are specified as follow: $\tilde{Q} = 0.01 * diag(1, 1, 1)$ and $\tilde{R} = 10e4*diag(1, 1, 1, 1)$. Note that the weighting matrix $\tilde{R}$ is constant over all prediction horizon, which allows to avoid matrix inversion at each sampling time with one calculation of $\tilde{R}^{-1}$ at time $k=0$.

The next two figures - Fig. 3 and Fig. 4, show typical results about level control, where the references for H_1, H_2 and H_3 are changed consequently in different time. The change of every level reference behaves as a system disturbance for the other system outputs (levels). It is evident that the applied model predictive controller is capable to compensate these disturbances.

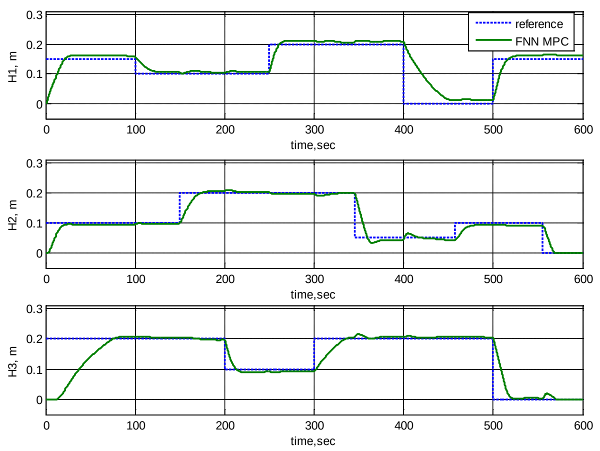

Fig. 3. Transient responses of multi tank system outputs with different level references

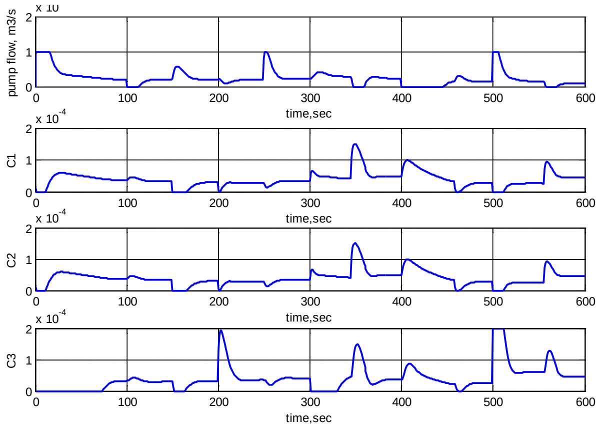

Fig. 4. Transient responses of multi tank system inputs with different level references

CONCLUSIONS

In this paper, the model predictive control scheme was employed to reduce structural response of the benchmark problem under a multi tank system. Model predictive control was applied successfully to the studied multi tank system, which is a very complex nonlinear and multivariable system. The inherent instability of the system makes it difficult to control. Adaptation of linear internal model is the most common way of dealing with plant nonlinearities in practice. The results show that the controlled levels have a good performance. The next efforts will be directed to the real-time implementation of model predictive control to the multi tank system.

ACKNOWLEDGMENTS

The authors would like to acknowledge the Ministry of Education and Science of Bulgaria, Research Fund project BY-TH-108/2005.

REFERENCES

1. Ahmed S., M. Petrov, A. Ichtev. Fuzzy Model-Based Predictive Control Applied to Multivariable Level Control of Multi Tank System. Proceedings of 2010 IEEE International Conference on Intelligent Systems (IS 2010), London, UK. July 2010. pp. 456 - 461.
2. Camacho E. F., C. Bordons (2004). Model Predictive Control (Advanced Textbooks in Control and Signal Processing). Springer-Verlag London, 2004.
3. Inteco Ltd. (2009). Multitank System - User's Manual. Inteco Ltd., http://www.inteco.com.pl.
4. Lee, J. H., Morari, M., & Garcia, C. E. State-space interpretation of model predictive control. Automatica, 30(4), 1994, 707–717.
5. Maciejowski J.M.. Predictive Control with Constraints. Prentice Hall, 2001.
6. Petrov M., I. Ganchev, A. Taneva. Fuzzy model predictive control of nonlinear processes. Preprints of the International Conference on "Automation and Informatics 2002", Sofia, Bulgaria, 2002. ISBN 954-9641-30-9, pp. 77-80.
7. Richalet, J. Industrial applications of model-based control. Automatica, 29, 1993, 1251–1274.
8. Rossitier J. A., Model based predictive control – A practical Approach, CRC Press, 2003.