Methods of Prediction Improvement in Efficient MPC Algorithms Based on Fuzzy Hammerstein Models

Piotr M. Marusak

Institute of Control and Computation Engineering, Warsaw University of Technology, ul. Nowowiejska 15/19, 00–665 Warszawa, Poland,
P.Marusak@ia.pw.edu.pl

Abstract. Methods of prediction improvement in Model Predictive Control (MPC) algorithms utilizing fuzzy Hammerstein models are proposed in the paper. Though they can significantly improve control system operation, they modify the prediction in such a way that it is described by relatively simple analytical formulas. Thus, the prediction has such a form that the MPC algorithms using it are formulated as numerically efficient quadratic optimization problems. First prediction improvement method uses values of the future control changes which were derived by the MPC algorithm in the last iteration. Thanks to such an approach the prediction can be iteratively adjusted and the MPC algorithm using it can offer very good control performance. It is also shown how the prediction can be extended with information about influence of a measurable disturbance. It can significantly improve control performance offered by the algorithm. Efficiency of the MPC algorithms based on the prediction utilizing the proposed methods of improvement is demonstrated in the example control system of a nonlinear control plant with significant time delay.

Key words: fuzzy control, fuzzy systems, predictive control, nonlinear control, constrained control

1 Introduction

Model Predictive Control (MPC) algorithms are widely used in practical applications due to numerous advantages they offer. They use prediction of the control plant behavior during calculation of control signal [2, 6, 13, 15]. Thanks to such an approach they can be successfully applied to processes with difficult dynamics (large time delays, inverse response), also constraints existing in the control system (put e.g. on manipulated and output variables) can be relatively easy taken into consideration in these algorithms. Moreover, all information available to a control system designer can be relatively easy used to improve prediction the MPC algorithm is based on and, as a result, to improve control performance offered by the MPC algorithm. This feature will be exploited in the paper.

In order to obtain prediction a model of the control plant is utilized. As different models can be used to obtain the prediction, different MPC algorithms were designed. In the standard MPC algorithms linear process models are used. In such a case however, if the control plant is nonlinear the MPC algorithm may generate inefficient results. Therefore, it is good to use nonlinear models during prediction generation. An interesting class of nonlinear models are Hammerstein models [5]. In these models the nonlinear static block precedes the linear dynamic block. Such structure of the model makes possible to do synthesis of the MPC algorithm relatively easy if the model is appropriately used (see e.g. [9, 10]). In current research it is assumed that the dynamic part of the Hammerstein model has the form of the step response. It is also assumed that in the static block of the Hammerstein model the fuzzy Takagi–Sugeno (TS) model is used. Thanks to such an approach advantages offered by the fuzzy TS models [12], like e.g. relative easiness of model identification, relatively small number of rules needed to describe highly nonlinear functions or property that the fuzzy reasoning makes possible to obtain a linear approximation of the model relatively easy, can be utilized.

Direct usage of a nonlinear process model in the MPC algorithm leads to its formulation as a nonlinear, nonquadratic, often nonconvex optimization problem, which must be solved in each iteration of the algorithm. Despite improved versions of procedures solving the nonlinear optimization problems are designed (see e.g. [3] for modifications of particle swarm optimization and of genetic algorithms, taking into account properties of modern CPUs), nonlinear, nonconvex optimization has serious drawbacks. Time needed to find the solution of such an optimization problem is hard to predict. There is also problem of local minima. Moreover, in some cases numerical problems may occur. The drawbacks of the MPC algorithms formulated as nonlinear optimization problems caused that usually MPC algorithms formulated as the standard quadratic programming problems, and utilizing a linear approximation of the control plant model, obtained at each iteration, are used [7–11, 15].

The methods of prediction improvement proposed in the paper can be divided into two groups. The fist one consists in using sophisticated methods of prediction generation using the fuzzy Hammerstein model of the process. Let remind that in [9] prediction is obtained using the original fuzzy Hammerstein model and its linear approximation. In [10] improvement of this method of prediction was proposed. It uses not only the original fuzzy Hammerstein model and its linear approximation but also values of the future control changes derived by the MPC algorithm in the previous iteration. In the paper it is described how to improve this method of prediction even more. It is possible because the prediction can be iteratively adjusted. As a result it is closer to the prediction obtained using the nonlinear process model only. However, still the prediction can be obtained relatively easy and the MPC algorithm using it can be formulated as the quadratic optimization problem. The second group of methods makes possible to take into consideration disturbance measurement during prediction generation.

It will be demonstrated in the example control system that utilization of such mechanisms may radically improve control performance.

In the next section the general idea of the MPC algorithms is described. In Sect. 3 MPC algorithms based on linear models are described. In Sect. 4 the MPC algorithms based on the fuzzy Hammerstein model are reminded and then the proposed methods of prediction improvement are described. The efficacy of the MPC algorithms utilizing the proposed methods of prediction improvement is illustrated by the example results presented in Sect. 5. The last section contains summary of the paper.

2 General Idea of Model Predictive Control Algorithms

The MPC algorithms during its operation use prediction of the future process behavior many sampling instants ahead. The prediction is obtained utilizing a model of the control plant. Future values of the control signal are calculated in such a way that the prediction fulfills assumed criteria. These criteria are used to formulate an optimization problem which is solved at each iteration of the algorithm. The optimization problem has usually the following form [2, 6, 13, 15]:

$$\arg\min_{\Delta \boldsymbol{u}} \left\{ J_{\mathrm{MPC}} = (\overline{\boldsymbol{y}} - \boldsymbol{y})^T \cdot (\overline{\boldsymbol{y}} - \boldsymbol{y}) + \Delta \boldsymbol{u}^T \cdot \boldsymbol{\Lambda} \cdot \Delta \boldsymbol{u} \right\} \tag{1}$$

subject to:

$$\Delta \boldsymbol{u}_{\min} \le \Delta \boldsymbol{u} \le \Delta \boldsymbol{u}_{\max} \ , \tag{2}$$

$$\boldsymbol{u}_{\min} \le \boldsymbol{u} \le \boldsymbol{u}_{\max} \ , \tag{3}$$

$$\boldsymbol{y}_{\min} \le \boldsymbol{y} \le \boldsymbol{y}_{\max} \ , \tag{4}$$

where $\overline{\boldsymbol{y}} = [\overline{y}_k, \ldots, \overline{y}_k]^T$ is the vector of length p, $\overline{y}_k$ is a set–point value, $\boldsymbol{y} = \left[y_{k+1|k}, \ldots, y_{k+p|k}\right]^T$, $y_{k+i|k}$ is a value of the output for the $(k+i)^{\mathrm{th}}$ sampling instant, predicted at the k^{th} sampling instant, $\Delta \boldsymbol{u} = \left[\Delta u_{k+1|k}, \ldots, \Delta u_{k+s-1|k}\right]^T$, $\Delta u_{k+i|k}$ are future changes in manipulated variable, $\boldsymbol{u} = \left[u_{k+1|k}, \ldots, u_{k+s-1|k}\right]^T$, $\boldsymbol{\Lambda} = \lambda \cdot \boldsymbol{I}$ is the $s \times s$ matrix, $\lambda \ge 0$ is a tuning parameter, p and s denote prediction and control horizons, respectively; $\Delta \boldsymbol{u}_{\min}$, $\Delta \boldsymbol{u}_{\max}$, $\boldsymbol{u}_{\min}$, $\boldsymbol{u}_{\max}$, $\boldsymbol{y}_{\min}$, $\boldsymbol{y}_{\max}$ are vectors of lower and upper limits of changes and values of the control signal and of the values of the output variable, respectively. The vector of optimal changes of the control signal is the solution of the optimization problem (1–4). From this vector, the first element, i.e. $\Delta u_{k|k}$ is applied in the control system and the algorithm goes to the next iteration.

The predicted values of the output variable $y_{k+i|k}$ are derived using the dynamic control plant model. If this model is nonlinear then the optimization problem (1–4) is, in general, nonconvex, nonquadratic, nonlinear and hard to solve. Examples of such algorithms are described e.g. in [1, 4].

3 DMC Algorithm as an Example of MPC Algorithms Based on Linear Models

The algorithms based on fuzzy Hammerstein models are related to the Dynamic Matrix Control (DMC) algorithm. Therefore it will be now described. Moreover, in the DMC algorithm incorporation of the mechanism of disturbance measurement utilization is relatively simple and can be adapted in the algorithms based on fuzzy Hammerstein models.

3.1 Model of the Process

In the DMC algorithm the process model in the form of the step response is used:

$$\widehat{y}_k = \sum_{i=1}^{p_d-1} a_i \cdot \Delta u_{k-i} + a_{p_d} \cdot u_{k-p_d} \tag{5}$$

where $\widehat{y}_k$ is the output of the control plant model at the k^{th} sampling instant, Δu_k is a change of the manipulated variable at the k^{th} sampling instant, a_i ($i = 1, \ldots, p_d$) are step response coefficients of the control plant, p_d is equal to the number of sampling instants after which the coefficients of the step response can be assumed as settled, u_{k-p_d} is a value of the manipulated variable at the $(k-p_d)^{\text{th}}$ sampling instant.

3.2 Generation of the Prediction

The output prediction is calculated using the following formula:

$$y_{k+i|k} = \sum_{n=1}^{i} a_n \cdot \Delta u_{k-n+i} + \sum_{n=i+1}^{p_d-1} a_n \cdot \Delta u_{k-n+i} + a_{p_d} \cdot u_{k-p_d+i} + d_k \tag{6}$$

where $d_k = y_k - \widehat{y}_{k-1}$ is assumed the same at each sampling instant in the prediction horizon (a DMC–type disturbance model), thus (6) can be transformed into the following form

$$\begin{aligned} y_{k+i|k} = y_k &+ \sum_{n=i+1}^{p_d-1} a_n \cdot \Delta u_{k-n+i} + a_{p_d} \cdot \sum_{n=p_d}^{p_d+i-1} \Delta u_{k-n+i} - \sum_{n=1}^{p_d-1} a_n \cdot \Delta u_{k-n} \\ &+ \sum_{n=p}^{i} a_n \cdot \Delta u_{k-n+i|k} \end{aligned} \tag{7}$$

Thus, the vector of predicted output values $\boldsymbol{y}$ can be decomposed into the following components:

$$\boldsymbol{y} = \widetilde{\boldsymbol{y}} + \boldsymbol{A} \cdot \Delta \boldsymbol{u} \ , \tag{8}$$

where $\widetilde{\boldsymbol{y}} = \left[\widetilde{y}_{k+1|k}, \ldots, \widetilde{y}_{k+p|k}\right]^T$ is a free response of the plant. It contains future values of the output variable calculated assuming that the control signal does not change in the prediction horizon, i.e. describes influence of the manipulated variable values from the past; $\boldsymbol{A}\cdot\boldsymbol{\Delta u}$ is the forced response which depends only on future changes of the control signal (decision variables); $\boldsymbol{A}$ is a matrix composed of the control plant step response coefficients and is called the dynamic matrix

$$\boldsymbol{A} = \begin{bmatrix} a_1 & 0 & \ldots & 0 & 0 \\ a_2 & a_1 & \ldots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_p & a_{p-1} & \ldots & a_{p-s+2} & a_{p-s+1} \end{bmatrix} \tag{9}$$

In the DMC algorithm the free response is given by the following formula:

$$\widetilde{\boldsymbol{y}} = \boldsymbol{y}_k + \widetilde{\boldsymbol{A}} \cdot \boldsymbol{\Delta u}^p \ , \tag{10}$$

where

$$\widetilde{\boldsymbol{A}} = \begin{bmatrix} a_2 - a_1 & a_3 - a_2 & \ldots & a_{p_d-1} - a_{p_d-2} & a_{p_d} - a_{p_d-1} \\ a_3 - a_1 & a_4 - a_2 & \ldots & a_{p_d} - a_{p_d-2} & a_{p_d} - a_{p_d-1} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{p+1} - a_1 & a_{p+2} - a_2 & \ldots & a_{p_d} - a_{p_d-2} & a_{p_d} - a_{p_d-1} \end{bmatrix},$$

$\boldsymbol{y}_k = [y_k, \ldots, y_k]^T$ is a vector of length p, $\boldsymbol{\Delta u}^p = [\Delta u_{k-1}, \ldots, \Delta u_{k-p_d}]^T$ is a vector of past changes of the manipulated variable.

Remark 1. In any MPC algorithm based on a linear model the superposition principle can be applied and the prediction $\boldsymbol{y}$ is described by the formula (8) [2, 6, 13, 15]. It can be also shown that the dynamic matrix $\boldsymbol{A}$ is present in all such algorithms [15]. However, formulas describing the free response will be different depending on the form of a linear model used to obtain the prediction.

3.3 Formulation of the Optimization Problem

After application of the prediction (8) to the performance function from the optimization problem (1) and to the output constraints (4) one obtains the following optimization problem:

$$\arg\min_{\boldsymbol{\Delta u}} \left\{ J_{\mathrm{LMPC}} = (\overline{\boldsymbol{y}} - \widetilde{\boldsymbol{y}} - \boldsymbol{A}\cdot\boldsymbol{\Delta u})^T \cdot (\overline{\boldsymbol{y}} - \widetilde{\boldsymbol{y}} - \boldsymbol{A}\cdot\boldsymbol{\Delta u}) + \boldsymbol{\Delta u}^T \cdot \boldsymbol{\Lambda} \cdot \boldsymbol{\Delta u} \right\} \tag{11}$$

subject to:

$$\boldsymbol{\Delta u}_{\min} \le \boldsymbol{\Delta u} \le \boldsymbol{\Delta u}_{\max} \ , \tag{12}$$

$$\boldsymbol{u}_{\min} \le \boldsymbol{u}_{k-1} + \boldsymbol{J}\cdot\boldsymbol{\Delta u} \le \boldsymbol{u}_{\max} \ , \tag{13}$$

$$\boldsymbol{y}_{\min} \le \widetilde{\boldsymbol{y}} + \boldsymbol{A}\cdot\boldsymbol{\Delta u} \le \boldsymbol{y}_{\max} \ , \tag{14}$$

where

$$\boldsymbol{J} = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 1 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \dots & 1 \end{bmatrix} .$$

Remark 2. Note that the performance function in (11) depends quadratically on decision variables $\boldsymbol{\Delta u}$. Moreover, as the prediction (8) was applied in the constraints (4), all the constraints depend linearly on decision variables. As a result, the optimization problem (11–14) is a standard linear–quadratic optimization problem.

3.4 Mechanism of Disturbance Measurement Utilization

The utilization of the disturbance measurement in the DMC algorithm is relatively simple. It is because it is sufficient to expand the model of the control plant with components describing influence of the disturbance on the control plant and then generate the prediction using the expanded model; see e.g. [15]. In short the procedure is as follows. The expanded model in the form of the step response has now the following form:

$$\hat{y}_k = \sum_{i=1}^{p_d-1} a_i \cdot \Delta u_{k-i} + a_{p_d} \cdot u_{k-p_z} + \sum_{i=1}^{p_z-1} e_i \cdot \Delta v_{k-i} + e_{p_z} \cdot v_{k-p_z} \tag{15}$$

where Δv_k is a change of the disturbance variable at the k^{th} sampling instant, e_i $(i = 1, \dots, p_z)$ are coefficients of disturbance step response of the control plant, p_z is equal to the number of sampling instants after which the coefficients of the disturbance step response can be assumed as settled, v_{k-p_z} is a value of the disturbance variable at the $(k-p_z)^{\text{th}}$ sampling instant.

The output prediction is thus now calculated using the following formula:

$$\begin{aligned} y_{k+i|k} = & \sum_{n=1}^{i} a_n \cdot \Delta u_{k-n+i} + \sum_{n=i+1}^{p_d-1} a_n \cdot \Delta u_{k-n+i} + a_{p_d} \cdot u_{k-p_d+i} \\ & + \sum_{n=1}^{i} e_n \cdot \Delta v_{k-n+i} + \sum_{n=i+1}^{p_z-1} e_n \cdot \Delta v_{k-n+i} + e_{p_z} \cdot v_{k-p_z+i} + d_k \ , \end{aligned} \tag{16}$$

where $d_k = y_k - \widehat{y}_{k-1}$ is assumed the same at each sampling instant in the prediction horizon and describes the influence of disturbances which cannot be measured or estimated. Note that if an estimate of the disturbance v is available, then the components dependent on it in (16) are known. Therefore, after transformation, they will be present in the free response. As a result the free response will be given by:

$$\widetilde{\boldsymbol{y}} = \boldsymbol{y}_k + \widetilde{\boldsymbol{A}} \cdot \boldsymbol{\Delta u}^p + \widetilde{\boldsymbol{E}} \cdot \boldsymbol{\Delta v}^p \ , \tag{17}$$

where

$$\widetilde{\boldsymbol{E}} = \begin{bmatrix} e_1 & e_2 - e_1 & e_3 - e_2 & \ldots & e_{p_z-1} - e_{p_z-2} & e_{p_z} - e_{p_z-1} \\ e_2 & e_3 - e_1 & e_4 - e_2 & \ldots & e_{p_z} - e_{p_z-2} & e_{p_z} - e_{p_z-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ e_p & e_{p+1} - e_1 & e_{p+2} - e_2 & \ldots & e_{p_z} - e_{p_z-2} & e_{p_z} - e_{p_z-1} \end{bmatrix} ,$$

$\Delta \boldsymbol{v}^p = [\Delta v_k, \Delta v_{k-1}, \ldots, \Delta v_{k-p_z}]^T$ is a vector of known changes of the disturbance variable v. Thus, in the prediction (8) only the free response component changes and the prediction linearly depends on decision variables. Thus, the optimization problem which must be solved at each iteration is the efficient linear–quadratic one.

4 MPC Algorithms Based on Fuzzy Hammerstein Models

4.1 Model of the Process

It is assumed that in the Hammerstein process model (Fig. 1) the static part has the form of a fuzzy Takagi–Sugeno model:

$$z_k = f(u_k) = \sum_{j=1}^{l} w_j(u_k) \cdot z_k^j = \sum_{j=1}^{l} w_j(u_k) \cdot (b_j \cdot u_k + c_j) \ , \tag{18}$$

where z_k is the output of the static block, $w_j(u_k)$ are weights obtained using fuzzy reasoning, z_k^j are outputs of local models in the fuzzy static model, l is the number of fuzzy rules in the model, b_j and c_j are parameters of the local models. Moreover, it is assumed that the dynamic part of the model has the form of the step response:

$$\widehat{y}_k = \sum_{n=1}^{p_d-1} a_n \cdot \Delta z_{k-n} + a_{p_d} \cdot z_{k-p_d} \ , \tag{19}$$

where $\widehat{y}_k$ is the output of the fuzzy Hammerstein model, a_i are coefficients of the step response, p_d is the horizon of the process dynamics.

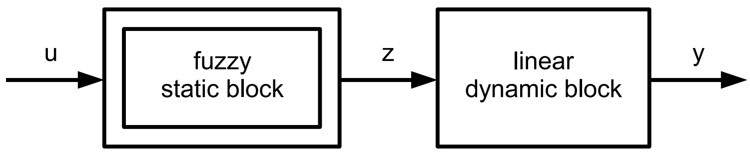

Fig. 1. Structure of the fuzzy Hammerstein model

Note, that form of the dynamic model is the same as of the model used for prediction in the DMC algorithm (Sect. 3). Thus, prediction can be obtained in a similar way as in the DMC algorithm.

Remark 3. If in a Hammerstein model the linear dynamic model is of different form than the step response, it can be always easily transformed to the form assumed here. Then the proposed approach can be applied.

4.2 Generation of the Prediction

In order to obtain the MPC algorithms in which the control signal is obtained using the linear–quadratic optimization problem, the prediction is obtained using a linear approximation of the fuzzy Hammerstein model. However, it is used only to obtain the dynamic matrix whereas the free response is obtained using the original nonlinear fuzzy model. Thanks to such an approach the algorithms offer very good control performance.

Generation of the Free Response

Method I

The free response can be obtained using its definition, i.e. assuming that the control signal will not change on the whole control horizon; see [9]. Note that the output of the model (19) in the i^{th} sampling instant is described by the following formula:

$$\widehat{y}_{k+i} = \sum_{n=1}^{i} a_n \cdot \Delta z_{k-n+i} + \sum_{n=i+1}^{p_d-1} a_n \cdot \Delta z_{k-n+i} + a_{p_d} \cdot z_{k-p_d+i} \ . \tag{20}$$

In (20) the first component depends on future action and the next ones on past control actions. The free response can be then calculated using the following formula:

$$\widetilde{y}_{k+i|k} = \sum_{n=i+1}^{p_d-1} a_n \cdot \Delta z_{k-n+i} + a_{p_d} \cdot z_{k-p_d+i} + d_k \ , \tag{21}$$

where $d_k = y_k - \widehat{y}_k$ is the DMC–type disturbance model of unmeasured disturbances, containing also influence of modeling errors.

Method II

The prediction can be improved thanks to utilization of future control increments derived by the MPC algorithm in the last sampling instant during calculation of the free response. It was described in [10]. Now, the method will be improved even more, because it is possible to adjust the prediction iteratively.

In the first internal iteration assume that future control values can be decomposed into two parts:

$$u^{(1)}_{k+i|k} = \breve{u}^{(1)}_{k+i|k} + u_{k+i|k-1} \ , \tag{22}$$

where $\breve{u}^{(1)}_{k+i|k}$ can be interpreted as the correction of the control signal $u_{k+i|k-1}$ obtained in the last $(k-1)^{\text{st}}$ iteration of the MPC algorithm; the upper index denotes the number of internal iteration of the prediction generation. Analogously, the future increments of the control signal will have the following form:

$$\Delta u^{(1)}_{k+i|k} = \Delta \breve{u}^{(1)}_{k+i|k} + \Delta u_{k+i|k-1} \ . \tag{23}$$

Note that the output of the model (19) in the i^{th} sampling instant is described by (20) where the first component depends on future action whereas the next ones depend on past control actions. Taking into account the decomposition of the input signal (22), (20) can be rewritten as:

$$\widehat{y}_{k+i} = \sum_{n=1}^{i} a_n \cdot \Delta \breve{z}^{(1)}_{k-n+i|k} + \sum_{n=1}^{i} a_n \cdot \Delta z_{k-n+i|k-1} + \sum_{n=i+1}^{p_d-1} a_n \cdot \Delta z_{k-n+i} + a_{p_d} \cdot z_{k-p_d+i} \ , \tag{24}$$

where $\Delta z_{k+i|k-1} = z_{k+i|k-1} - z_{k+i-1|k-1}$; $z_{k+i|k-1} = f(u_{k+i|k-1})$ and $\breve{z}^{(1)}_{k+i|k-1} = z^{(1)}_{k+i|k} - z_{k+i|k-1}$. In (24) the second component is known and can be included in the free response of the control plant. Therefore, the final formula describing the elements of the free response will have the following form:

$$\widetilde{y}_{k+i|k} = \sum_{n=1}^{i} a_n \cdot \Delta z_{k-n+i|k-1} + \sum_{n=i+1}^{p_d-1} a_n \cdot \Delta z_{k-n+i} + a_{p_d} \cdot z_{k-p_d+i} + d_k \ . \tag{25}$$

In the next step, the free response together with the Dynamic Matrix obtained as described in the next part of Sect. 4.2 are used to formulated the optimization problem (Sect. 4.3) after solution of which a new control signal trajectory is obtained. Now, either the control signal is applied to the control plant or the prediction is improved in the next internal iteration of the algorithm. In the latter case the next internal iterations are as follows.

Now, as the next control signal trajectory is available, future control values can be decomposed as:

$$u^{(j)}_{k+i|k} = \breve{u}^{(j)}_{k+i|k} + u^{(j-1)}_{k+i|k} \ , \tag{26}$$

where the upper index denotes the number of internal iteration of the algorithm and $j = 2, 3, ...$ The future increments of the control signal will now have the following form:

$$\Delta u^{(j)}_{k+i|k} = \Delta \breve{u}^{(j)}_{k+i|k} + \Delta u^{(j-1)}_{k+i|k} \ . \tag{27}$$

Then now (20) can be expressed as:

$$\widehat{y}_{k+i} = \sum_{n=1}^{i} a_n \cdot \Delta \breve{z}^{(j)}_{k-n+i|k} + \sum_{n=1}^{i} a_n \cdot \Delta z^{(j-1)}_{k-n+i|k} + \sum_{n=i+1}^{p_d-1} a_n \cdot \Delta z_{k-n+i} + a_{p_d} \cdot z_{k-p_d+i} \ . \tag{28}$$

As in the first internal iteration, in (28) the second component is known and can be included in the free response of the control plant. Therefore, the final formula describing the elements of the free response will now have the following form:

$$\widetilde{y}_{k+i|k} = \sum_{n=1}^{i} a_n \cdot \Delta z^{(j-1)}_{k-n+i|k} + \sum_{n=i+1}^{p_d-1} a_n \cdot \Delta z_{k-n+i} + a_{p_d} \cdot z_{k-p_d+i} + d_k \ . \tag{29}$$

Remark 4. The smaller values of obtained corrections of the control trajectory are the better prediction is obtained. It is because then it will be closer to the prediction obtained using only the nonlinear fuzzy model.

Remark 5. Number of internal iterations of the algorithm can be chosen depending on what is the size of changes of the future control increments. If the corrections of the control signal trajectory become rather small, there is no use to do more internal iterations.

Remark 6. Note that the proposed method is a generalization of the method proposed in [10] as the latter one is obtained by assumption that only one internal iteration is made.

Generation of the Dynamic Matrix

Next, at each iteration of the algorithm the dynamic matrix can be easily derived using a linear approximation of the fuzzy Hammerstein model (19) [9]:

$$\widehat{y}_k = dz_k \cdot \left(\sum_{n=1}^{p_d - 1} a_n \cdot \Delta u_{k-n} + a_{p_d} \cdot u_{k-p_d} \right) \ , \tag{30}$$

where dz_k is a slope of the static characteristic near the z_k. It can be calculated either analytically (if possible) or numerically using the formula

$$dz_k = \frac{\sum_{j=1}^{l} \left(w_j(u_k + du) \cdot (b_j \cdot (u_k + du) + c_j) - w_j(u_k) \cdot (b_j \cdot u_k + c_j) \right)}{du} \ , \tag{31}$$

where du is a small number. The dynamic matrix will be thus described by the following formula:

$$\boldsymbol{A}_k = dz_k \cdot \begin{bmatrix} a_1 & 0 & \dots & 0 & 0 \\ a_2 & a_1 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_p & a_{p-1} & \dots & a_{p-s+2} & a_{p-s+1} \end{bmatrix} \ . \tag{32}$$

Finally, the prediction can be obtained using the free response (29) and the dynamic matrix (32):

$$\boldsymbol{y} = \widetilde{\boldsymbol{y}} + \boldsymbol{A}_k \cdot \Delta\breve{\boldsymbol{u}} \ , \tag{33}$$

where $\Delta\breve{\boldsymbol{u}} = \left[\Delta\breve{u}_{k+1|k}^{(j)}, \dots, \Delta\breve{u}_{k+s-1|k}^{(j)} \right]^T$.

4.3 Formulation of the Optimization Problem

After application of prediction (33) to the performance function from (1) and in constraints (4) one obtains:

$$\arg\min_{\Delta\breve{\boldsymbol{u}}} \left\{ J_{\text{FMPC}} = (\overline{\boldsymbol{y}} - \widetilde{\boldsymbol{y}} - \boldsymbol{A}_k \cdot \Delta\breve{\boldsymbol{u}})^T \cdot (\overline{\boldsymbol{y}} - \widetilde{\boldsymbol{y}} - \boldsymbol{A}_k \cdot \Delta\breve{\boldsymbol{u}}) + \Delta\boldsymbol{u}^T \cdot \boldsymbol{\Lambda} \cdot \Delta\boldsymbol{u} \right\} \tag{34}$$

subject to:

$$\Delta \boldsymbol{u}_{\min} \leq \Delta \boldsymbol{u} \leq \Delta \boldsymbol{u}_{\max} \ , \tag{35}$$

$$\boldsymbol{u}_{\min} \leq \boldsymbol{u}_{k-1} + \boldsymbol{J} \cdot \Delta \boldsymbol{u} \leq \boldsymbol{u}_{\max} \ , \tag{36}$$

$$\boldsymbol{y}_{\min} \leq \widetilde{\boldsymbol{y}} + \boldsymbol{A}_k \cdot \Delta \breve{\boldsymbol{u}} \leq \boldsymbol{y}_{\max} \ , \tag{37}$$

where $\Delta \boldsymbol{u} = \Delta \breve{\boldsymbol{u}} + \Delta \boldsymbol{u}^{\boldsymbol{p}}$, $\Delta \boldsymbol{u}^{\boldsymbol{p}} = \left[\Delta u_{k|k}^{(j)}, \ldots, \Delta u_{k+s-2|k}^{(j)}, \Delta u_{k+s-1|k}^{(j)}\right]^T$; compare with (27). Moreover, formulas (26) and (27) are used to modify the constraints (35) and (36) respectively. Then, the linear–quadratic optimization problem with the performance function (34), constraints (35)–(37) and the decision variables $\Delta \breve{\boldsymbol{u}}$ is solved at each iteration in order to derive the control signal.

Remark 7. It is also possible to use slightly modified performance function in which in the second component, only corrections of the control changes $\Delta \breve{\boldsymbol{u}}$ are panelized, i.e.

$$J_{\mathrm{FMPC2}} = (\overline{\boldsymbol{y}} - \widetilde{\boldsymbol{y}} - \boldsymbol{A}_k \cdot \Delta \breve{\boldsymbol{u}})^T \cdot (\overline{\boldsymbol{y}} - \widetilde{\boldsymbol{y}} - \boldsymbol{A}_k \cdot \Delta \breve{\boldsymbol{u}}) + \Delta \breve{\boldsymbol{u}}^T \cdot \boldsymbol{\Lambda} \cdot \Delta \breve{\boldsymbol{u}} \ . \tag{38}$$

Such a modification, however, causes that the meaning of the tuning parameter λ is different than in the classical performance function. As a consequence, the algorithm with the modified performance function generates faster responses. It will be demonstrated in the example control system.

4.4 Mechanisms of Disturbance Measurement Utilization

First method of disturbance measurement utilization consists in using the same approach as in the standard DMC algorithm. This method can be used e.g. when one wants to extend the existing algorithm with the disturbance measurement utilization mechanism. Then the process model should be extended like it is done in the standard DMC algorithm. Thus, the structure of the model will be as depicted in Fig. 2.

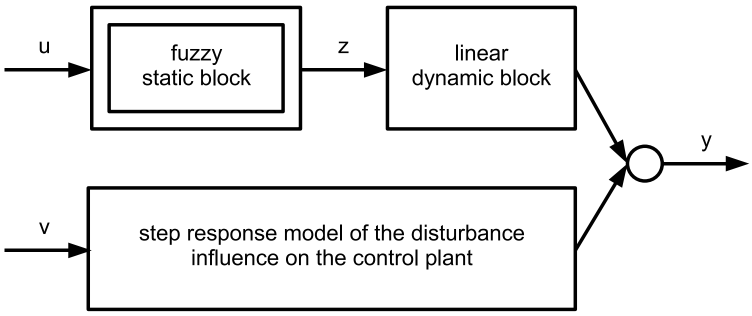

Fig. 2. Hammerstein model supplemented with step response model of influence of the disturbance on the control plant

As a result the prediction changes. Precisely, to the free response obtained in Sect. 4.2 an appropriate term from (17) is simply added. As a result the following formula describing the prediction is obtained (compare with (33)):

$$\boldsymbol{y} = \widetilde{\boldsymbol{y}} + \widetilde{\boldsymbol{E}} \cdot \Delta \boldsymbol{v}^p + \boldsymbol{A}_k \cdot \Delta \breve{\boldsymbol{u}} \ , \tag{39}$$

Second method of disturbance measurement utilization is in fact easier to apply. It is because it consists in proper usage of knowledge about disturbance estimates and original Hammerstein model of the process during the free response generation. Assume that the model has structure shown in Fig. 3.

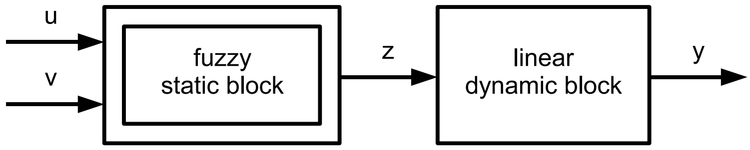

Fig. 3. Hammerstein model with static part dependent on the disturbance

Application of the method is straightforward it is because the fuzzy static model is given by the following formula:

$$z_k = f(u_k, v_k) = \sum_{j=1}^{l} w_j(u_k, v_k) \cdot z_k^j = \sum_{j=1}^{l} w_j(u_k) \cdot (b_j \cdot u_k + e_j \cdot v_k + c_j) \ , \tag{40}$$

where v_k is the current disturbance estimate, $w_j(u_k, v_k)$ are weights obtained using fuzzy reasoning, e_j are parameters of the local models. Thus, the formulas describing the prediction in the FMPC algorithm given in Sect. 4.2 can be applied without changes.

5 Simulation Experiments

5.1 Control Plant

The control plant under consideration is an ethylene distillation column DA–303 from petrochemical plant in Plock. It is a highly nonlinear plant with a large time delay. The steady–state characteristic of the plant is shown in Fig. 4. The Hammerstein model of the control plant has the structure shown in Fig. 5. The output of the plant y is the impurity of the product. The manipulated variable u is the reflux. The higher the reflux is the purer product is obtained. During experiments it was assumed that the reflux is constrained $4.05 \leq u \leq 4.4$. The measurable disturbance variable v is the composition of the raw substance.

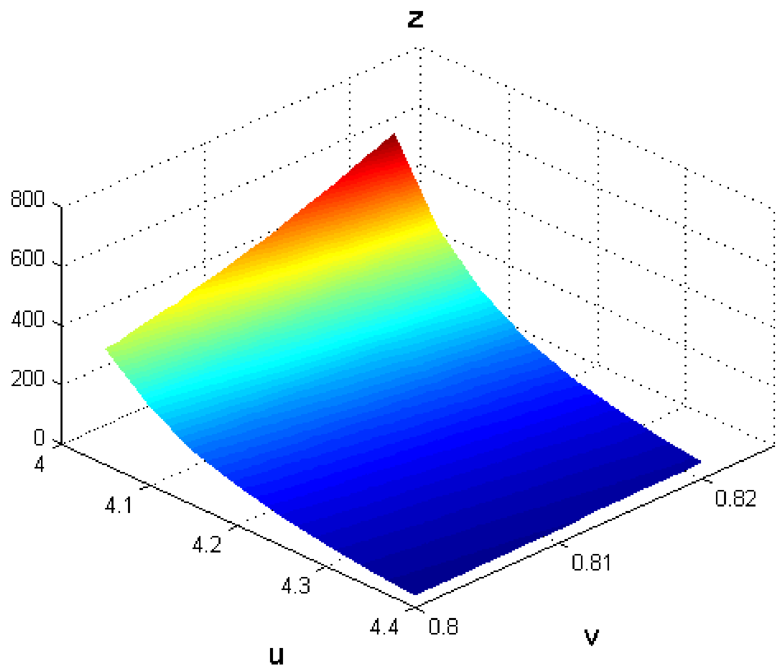

Fig. 4. Steady–state characteristic of the plant

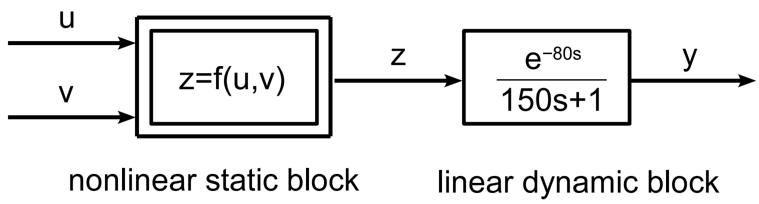

Fig. 5. Hammerstein model of the control plant

In the Hammerstein model of the plant (Fig. 5) the static part was modeled using ANFIS from Matlab. In order to train the system two sets of data were used: one for training and the second one for validation. The number of membership functions (assumed the same for both inputs) and number of training epochs were set experimentally. During the experiments increase of both parameters brought decrease of the MSE for data used for training. However, increase of the number of membership functions brought increase of the MSE obtained for validation data set. Therefore, two membership functions for both input variables were finally set. The number of epochs was assumed equal to 200. The system composed of 4 rules was finally obtained with the membership functions as shown in Fig. 6. The steady–state characteristic modeled by the ANFIS is almost the same as the one depicted in Fig. 4 (differences are negligible).

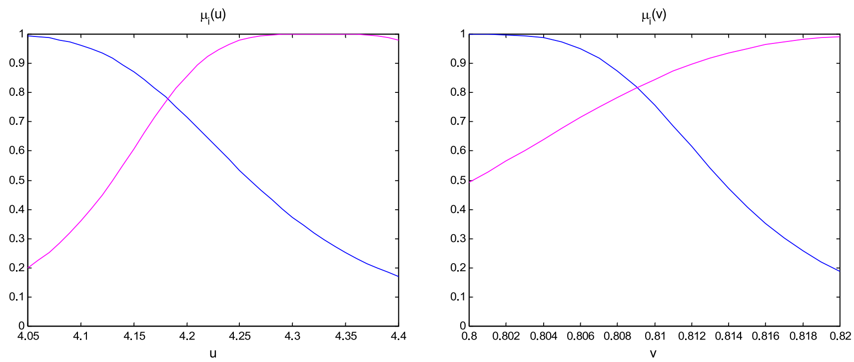

Fig. 6. Membership functions in the static part of the Hammerstein model

5.2 Results

A few MPC algorithms were designed:

- the NMPC one (with nonlinear optimization),
- the LMPC one (based on a linear model) and
- the FMPC ones (using the proposed method of prediction generation, based on the fuzzy Hammerstein model). Both versions of the algorithm were tested: the first one (FMPC1) with the classical performance function (34) and the second one (FMPC2) with the modified performance function (38). In both algorithms only one internal iteration was used.

The sampling period was assumed equal to $T_s = 20$ min, the prediction horizon $p = 44$ and the control horizon $s = 20$.

First, the experiments as in [10] were done because the control plant model used to design the FMPC and NMPC algorithms was changed (it was obtained using ANFIS instead of a heuristic approach). Example responses to the set–point changes, obtained for $\lambda = 10^7$ are shown in Fig. 7 and those obtained for $\lambda = 10^6$ — in Fig. 8.

The responses obtained with the LMPC algorithm to the set–point change to $\overline{y}_3 = 400$ ppm are unacceptable. The control system is very close to the boundary of stability in the case of $\lambda = 10^7$ and for $\lambda = 10^6$ it is unstable. Moreover, the overshoot in responses to the set–point change to $\overline{y}_2 = 300$ ppm are very big. On the contrary, both FMPC algorithms work well for all the set–point values and the responses have similar shape regardless the set–point value.

The responses obtained with the FMPC1 algorithm are very close to those obtained with the NPMC algorithm. However, it should be emphasized that the control signal is generated by the FMPC1 algorithm, after solving the numerically robust linear–quadratic optimization problem, much faster than in the case of the NMPC algorithm.

In the case of the NMPC algorithm numerical problems during calculation of the control signal may occur. It happened for $\lambda = 10^6$ in the case described in [10], when it utilized the fuzzy Hammerstein model in which the fuzzy static model was obtained heuristically. Fortunately, in the case considered now, the NMPC algorithm found the control signal at each iteration but it took much more time than in the case of the FMPC algorithms.

The fastest responses were obtained with the FMPC2 algorithm (blue lines in Figs. 7 and 8). They are better even than those obtained with the NMPC algorithm. It is however a result of assuming, in the optimization problem solved by the FMPC2 algorithm, a slightly different performance function than in other tested algorithms.

After decrease of the tuning parameter value to $\lambda = 10^6$, FMPC and NMPC algorithms work faster (Fig. 8) than in the case when $\lambda = 10^7$ (Fig. 7). The FMPC2 algorithm in response to the set–point change to $\overline{y}_1 = 200$ ppm works slightly faster even than the NMPC algorithm. In the case of set–point changes to $\overline{y}_2 = 300$ ppm and to $\overline{y}_3 = 400$ ppm the differences in operation between FMPC1, FMPC2 and NMPC algorithms are small. It is due to the fact that the control signal hits the constraint.

In the next experiments the mechanisms of disturbance measurement utilization were tested. In the case of both FMPC algorithms similar responses were obtained for different values of λ. The first method of disturbance measurement utilization (magenta lines in Figs. 9, 10 and 11) consisting in using the same approach as in the standard DMC algorithm, works surprisingly well especially near 200 ppm (there it is practically as good as the second method). It is because the model of disturbance influence on the control plant was obtained near this output value. However, for 300 ppm and 400 ppm the second method which uses appropriately prepared fuzzy Hammerstein model is the best (red lines). It can be also noticed that for $\lambda = 10^7$ the FMPC2 algorithm compensates the disturbance influence better than the FMPC1 algorithm. However, for $\lambda = 10^6$ the differences between algorithms were so small that only responses obtained in the control system with FMPC1 algorithm were presented (in Fig. 11). It

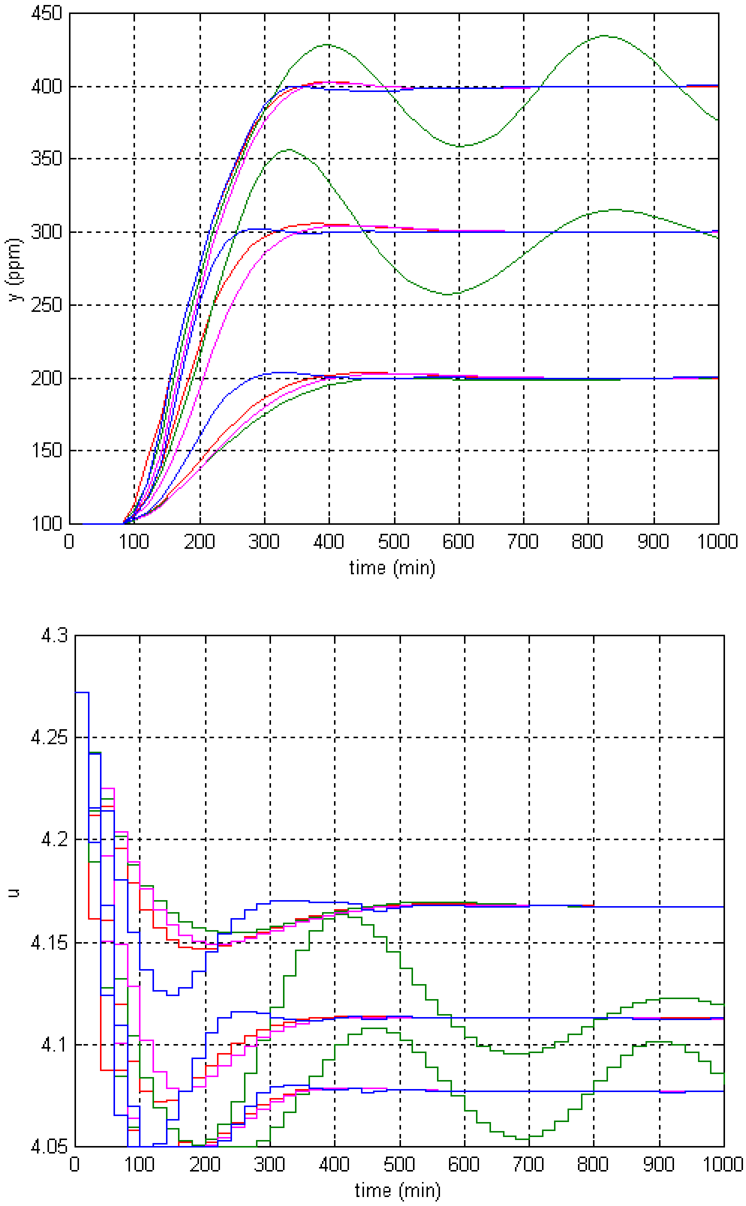

Fig. 7. Responses of the control systems to the changes of the set–point values to $\overline{y}_1 = 200$ ppm, $\overline{y}_2 = 300$ ppm and $\overline{y}_3 = 400$ ppm, $\lambda = 10^7$; NMPC – red lines, FMPC1 – magenta lines, FMPC2 – blue lines, LMPC – green lines; left – output signal, right – control signal

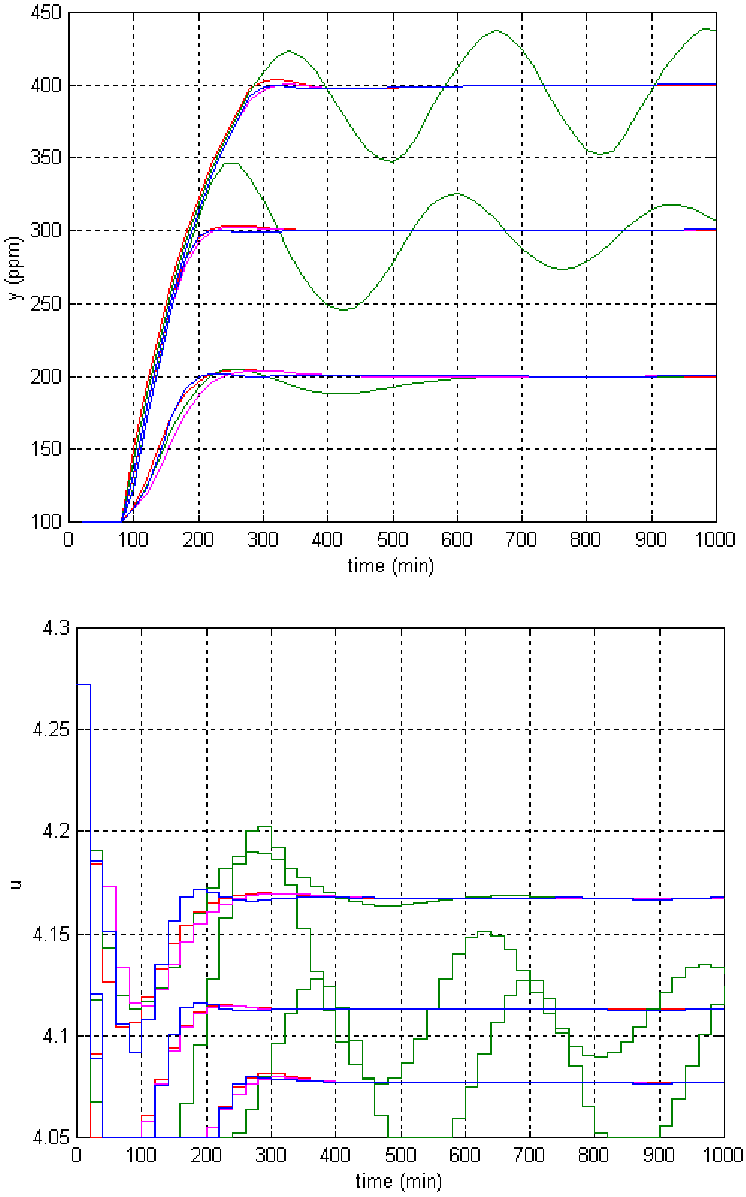

Fig. 8. Responses of the control systems to the changes of the set–point values to $\overline{y}_1 = 200$ ppm, $\overline{y}_2 = 300$ ppm and $\overline{y}_3 = 400$ ppm, $\lambda = 10^6$; NMPC – red lines, FMPC1 – magenta lines, FMPC2 – blue lines, LMPC – green lines; left – output signal, right – control signal

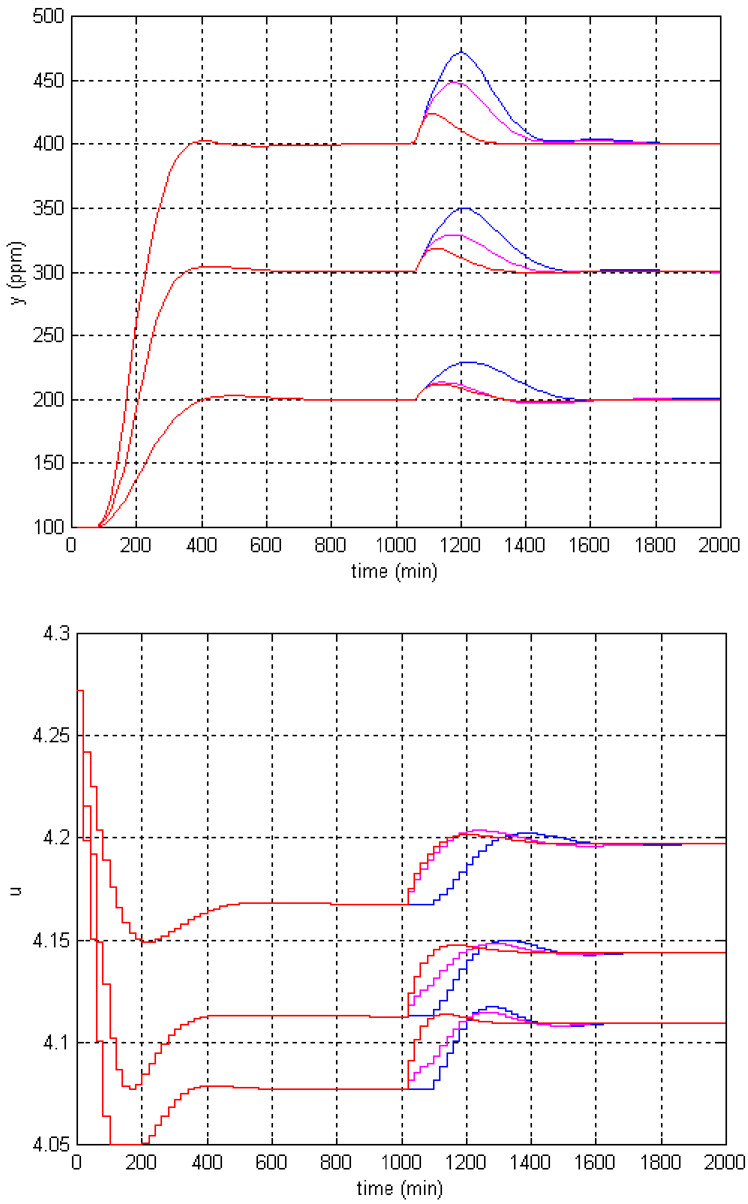

Fig. 9. Responses of the control system with FMPC1 algorithm to the changes of the set–point values to $\overline{y}_1 = 200$ ppm, $\overline{y}_2 = 300$ ppm and $\overline{y}_3 = 400$ ppm at the beginning of experiment and changes of the disturbance v in the 1000^{th} minute, $\lambda = 10^7$; disturbance measurement: not utilized – blue lines, utilized using the first method – magenta lines, utilized using the second method – red lines; left – output signal, right – control signal

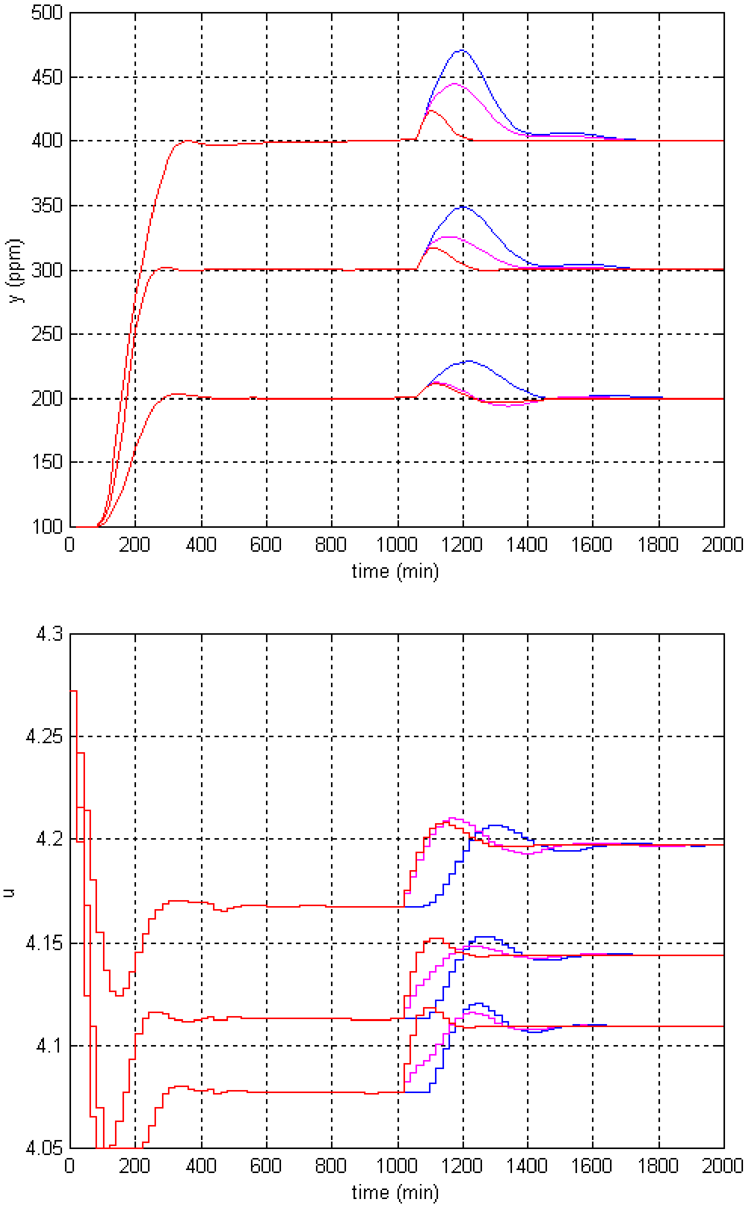

Fig. 10. Responses of the control system with FMPC2 algorithm to the changes of the set–point values to $\overline{y}_1 = 200$ ppm, $\overline{y}_2 = 300$ ppm and $\overline{y}_3 = 400$ ppm at the beginning of experiment and changes of the disturbance v in the 1000^{th} minute, $\lambda = 10^7$; disturbance measurement: not utilized – blue lines, utilized using the first method – magenta lines, utilized using the second method – red lines; left – output signal, right – control signal

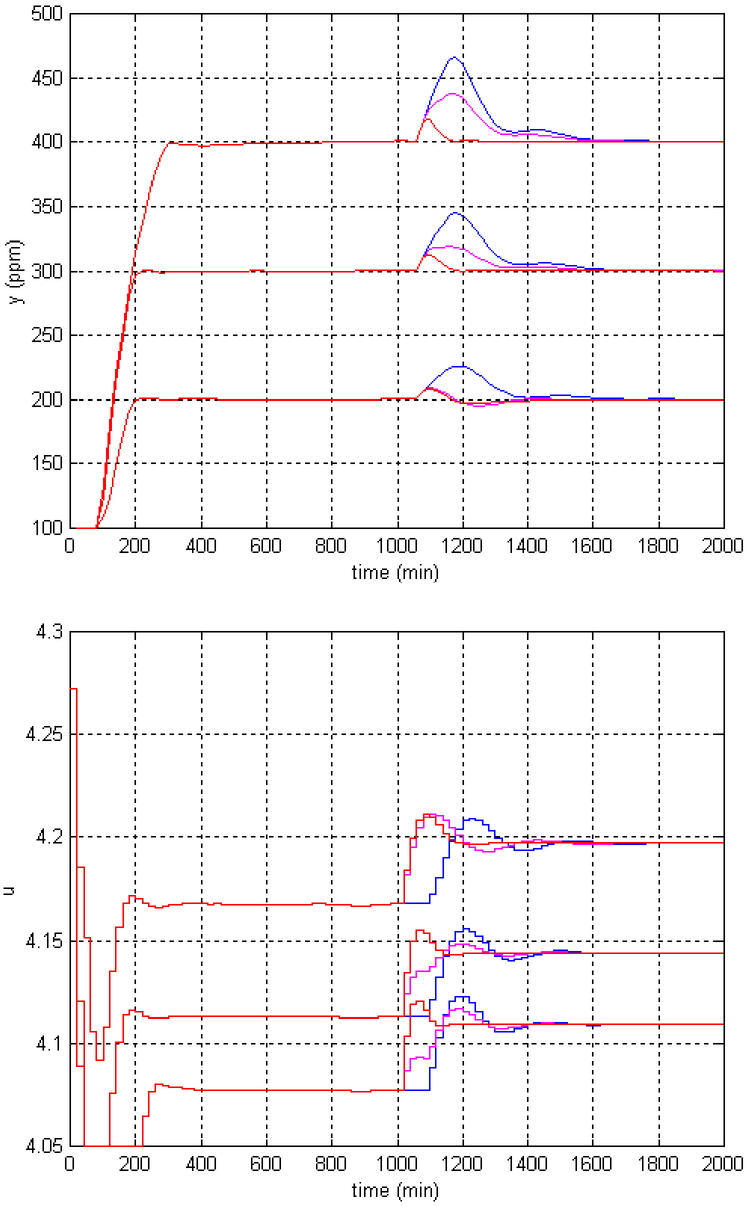

Fig. 11. Responses of the control system with FMPC1 algorithm to the changes of the set–point values to $\overline{y}_1 = 200$ ppm, $\overline{y}_2 = 300$ ppm and $\overline{y}_3 = 400$ ppm at the beginning of experiment and changes of the disturbance v in the 1000^{th} minute, $\lambda = 10^6$; disturbance measurement: not utilized – blue lines, utilized using the first method – magenta lines, utilized using the second method – red lines; left – output signal, right – control signal

can be also noticed that when λ was smaller (the algorithms were faster) the disturbance compensation was better.

6 Summary

The methods of prediction improvement in the MPC algorithms based on fuzzy Hammerstein models were described. The first method uses the future changes of the control signal, calculated in the last iteration by the MPC algorithm to derive the free response of the control plant. Such an approach causes that the prediction is closer to the one obtained using only the nonlinear process model. Moreover, it can be iteratively adjusted if needed. The second modification consists in utilization of the disturbance measurement to make its compensation better. Two mechanisms were proposed. Both do its job well but the one based on appropriately prepared fuzzy Hammerstein model gives very good results in the example control system of the highly nonlinear control plant with large time delay.

All the proposed modifications were performed in such a way that the prediction is described by relatively simple analytical formulas and the MPC algorithms using it are formulated as numerically efficient quadratic optimization problems. As a result, the FMPC algorithms using the proposed mechanisms offer practically the same control performance as the NMPC algorithm but need much less calculations (and time) to generate the control signal. At the same time they outperform its counterparts based on linear models.

References

1. Babuska R., te Braake H. A. B., van Can H. J. L., Krijgsman A. J., Verbruggen H. B.: Comparison of intelligent control schemes for real–time pressure control. Control Engineering Practice, **4** (1996) 1585–1592
2. Camacho, E.F., Bordons, C.: Model Predictive Control. Springer–Verlag (1999)
3. Chang, F.C., Huang, H.C.: A refactoring method for cache–efficient swarm intelligence algorithms. Information Sciences, in press, (2010), doi:10.1016/j.ins.2010.02.025
4. Fink A., Fischer M., Nelles O., Isermann R.: Supervision of nonlinear adaptive controllers based on fuzzy models. Control Engineering Practice, **8** (2000) 1093–1105
5. Janczak A.: Identification of nonlinear systems using neural networks and polynomial models: a block–oriented approach. Springer, Heidelberg (2005)
6. Maciejowski, J.M.: Predictive control with constraints. Prentice Hall, Harlow (2002)
7. Marusak P.: Advantages of an easy to design fuzzy predictive algorithm in control systems of nonlinear chemical reactors. Applied Soft Computing, **9** (2009) 1111–1125
8. Marusak P.: Efficient model predictive control algorithm with fuzzy approximations of nonlinear models. In: Kolehmainen, M., Toivanen, P., Beliczynski, B. (eds.) ICANNGA 2009. LNCS, vol. 5495, pp. 448–457. Springer, Heidelberg (2009)

9. Marusak P.: On prediction generation in efficient MPC algorithms based on fuzzy Hammerstein models. In: Rutkowski L., et al. (eds.) ICAISC 2010, Part I. LNAI, vol. 6113, pp. 136–143. Springer, Heidelberg (2010)
10. Marusak P.: Advanced prediction method in efficient MPC algorithm based on fuzzy Hammerstein models. Jedrzejowicz P., et al. (eds.): ICCCI 2011, Part I. LNCS, vol. 6922, pp. 193–202. Springer, Heidelberg (2011)
11. Morari, M., Lee J.H.: Model predictive control: past, present and future. Computers and Chemical Engineering, **23** (1999) 667–682
12. Piegat, A.: Fuzzy Modeling and Control. Physica–Verlag, Berlin (2001)
13. Rossiter, J.A.: Model-Based Predictive Control. CRC Press, Boca Raton (2003)
14. Takagi, T., Sugeno, M.: Fuzzy identification of systems and its application to modeling and control. IEEE Trans. Systems, Man and Cybernetics **15** (1985) 116–132
15. Tatjewski, P.: Advanced Control of Industrial Processes; Structures and Algorithms. Springer, London (2007)