BULGARIAN ACADEMY OF SCIENCES

CYBERNETICS AND INFORMATION TECHNOLOGIES • Volume **12**, No 1

Sofia • 2012

Algorithm for Multiple Model Adaptive Control Based on Input-Output Plant Model

Tsonyo Slavov

Department of Automatics, Technical University of Sofia, 1756 Sofia, Bulgaria
Email: ts_slavov@tu-sofia.bg

Abstract: *An algorithm for multiple model adaptive control of a time-variant plant in the presence of measurement noise is proposed. This algorithm controls the plant using a bank of PID controllers designed on the base of time invariant input/output models. The control signal is formed as weighting sum of the control signals of local PID controllers. The main contribution of the paper is the objective function minimized to determine the weighting coefficients. The proposed algorithm minimizes the sum of the square general error between the model bank output and the plant output. An equation for on-line determination of the weighting coefficients is obtained. They are determined by the current value of the general error covariance matrix. The main advantage of the algorithm is that the derived general error covariance matrix equation is the same as this in the recursive least square algorithm. Thus, most of the well known RLS modifications for the tracking time-variant parameters can be directly implemented. The algorithm performance is tested by simulation. Results with both SISO and MIMO time variant plants are obtained.*

Keywords: *Multiple model adaptive control, input-output model, PID controllers, time variant plants.*

1. Introduction

The control system design has to be often realized under apriori uncertainty of the process model parameters. On the other hand, many processes are significantly changing their parameters during their normal functioning. Consequently, a general control system design problem is to provide efficient control of the processes with significant parameters changes. The *Multiple Model Adaptive Control* (*MMAC*) is one of the major approaches for control under significant parameters uncertainty [1-5]. The main idea of MMAC is that the complex plant dynamics can be represented by a discrete finite set of simple local models with constant parameters. Each of them describes the dynamics for one or more regimes. Then a limited discrete set of simple local controllers tuned according to the corresponding simple model is designed. The control is formed as weighting sum of local controllers control signals. The weighting coefficients are determined on-line.

Historically, the MMAC arises with the necessity to use the set of linear controllers in a state space system under the conditions of apriori uncertainty in the plant dynamics. In order to estimate the corresponding state vector in the plant linearized in a certain operation mode, linear Kalman filters are used [4]. The first idea for utilizing a set of linear Kalman filters is formed into the so called static multiple-model state space estimator [6]. Later a scheme of interactive multiple-model estimator is proposed in [7], where a general state vector is calculated with different weight of each local filter operation according to apriori defined transition matrix. There are many different MMAC algorithms based on the state space controllers and Kalman filters [8] and there is a lack of such based on the input-output model. Very close to MMAC based on the input-output model is the *Multiple-Model Adaptive Switching Control* [9-12]. The main idea is that each controller of the bank takes an independent action in the control system tuned according to the corresponding plant model at the corresponding regime. The on-line controller switching is based on the performance index evaluation of the bank of models and/or controllers [3, 13]. This approach is suitable when the plant operating regimes are apriori known and/or well defined. The useful MMAC algorithm based on the linear input-output model and deadbeat controller is proposed in [12, 14, 15]. This algorithm does not use Kalman filters. It is especially suitable for control in case of low variance measurement noise. The weighting coefficients are determined based on the current value of the inverse output model error or the current value of the inverse exponential smoothing output model error.

In this paper MMAC algorithm based on input-output models and PID controllers is proposed. Each model has the same structure but different values of the parameters. The MMAC algorithm is similar to the multiple model adaptive control and state estimation algorithm presented in [16]. The principal difference is that the algorithm suggested in [16] is based on the state space models. It uses the bank of linear Kalman filters and the corresponding bank of LQR controllers. The other significant difference is in the objective function minimized to determine the weighting coefficients. The proposed algorithm minimizes the sum of the square general error between the model bank output and the plant output whereas the algorithm described in [16] minimizes the trace of the general residual variance or the trace of the general innovation term variance. The general residual and the innovation term are a multiple model Kalman filter residual and an innovation term. The advantage of the algorithm proposed here is that the current values of the weighting coefficients are determined by the current value of the general error covariance matrix. As it can be later seen the derived general error covariance matrix equation is the same as this in the recursive least square algorithm (RLS). This means that most of the well known RLS modifications for the tracking time-variant parameters can be directly implemented in the suggested algorithm.

The content of the paper is as follows. In Section 2 the proposed MMAC algorithm is derived. In Section 3 the pseudo code of the MMAC algorithm is given. The results from the simulation of MMAC of both SISO and MIMO systems are presented in Section 4 and some conclusions are made in Section 5.

2. Multiple model adaptive control algorithm

The block-diagram of the control system based on the proposed MMAC algorithm is shown in Fig. 1.

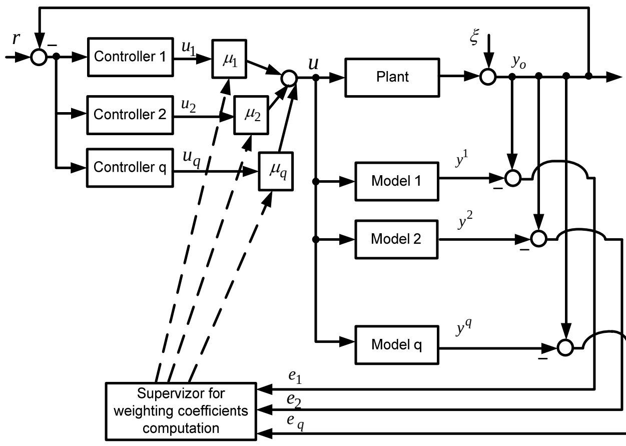

Fig. 1. Block-diagram of the control system based on MMAC algorithm

Let the controlled plant is time variant and be described with the equation

(1) $$y_0(s) = W(s,t)u(s) + \xi(s),$$

where $y_0(s) \in R^r$ is a vector containing plant outputs, $u(s) \in R^m$ is a vector containing plant inputs, $\xi(s) \in R^r$ is a vector containing the measurement noises and $W(s,t)$ is the transfer matrix. It has the form

$$W(s,t) = \begin{bmatrix} W_{11}(s,t) & W_{12}(s,t) & \dots & W_{1m}(s,t) \\ W_{21}(s,t) & W_{22}(s,t) & \dots & W_{2m}(s,t) \\ \vdots & \vdots & \vdots & \vdots \\ W_{r1}(s,t) & W_{r2}(s,t) & \dots & W_{rm}(s,t) \end{bmatrix}.$$

The elements of $W(s,t)$ are given by

$$W_{ij}(s,t)=\frac{b_{0_{ij}}(t)s^{n_{ij}}+b_{1_{ij}}(t)s^{n_{ij}-1}+...+b_{n_{ij}}(t)}{s^{n_{ij}}+a_{1_{ij}}(t)s^{n_{ij}-1}+...+a_{n_{ij}}(t)},\ i=1,2,...,r,\ \ j=1,2,...,m\,,$$

where $b_{1_{ij}}(t),\ b_{2_{ij}}(t),...,b_{n_{ij}}(t)$ and $a_{1_{ij}}(t),\ a_{2_{ij}}(t),...,a_{n_{ij}}(t)$ are the transfer function parameters. It is supposed that the parameters $b_{l_{ij}}$ and $a_{l_{ij}}$, $l=1,2,...,n$ are changing according to known value intervals $b_{l_{ij}}\in[b_{l_{ij}\min}\ \ b_{l_{ij}\max}]$ and $a_{l_{ij}}\in[a_{l_{ij}\min}\ \ a_{l_{ij}\max}]$. Then the complex plant dynamics can be approximated with a limited set of time invariant models referred as local ones. Each local model contains a combination of $b_{l_{ij}}$ and $a_{l_{ij}}$ values of parameters into the known variation intervals. These continuous-time transfer functions are put into a discrete-time form in order to design a set of discrete PID controllers.

The set of local models forms the model bank in the structure scheme presented in Fig. 1. The description of *i*-th discrete-time local model is given by

$$y^i(k)=W^i(q)u(k),$$

(2) $$W_{ij}(q)=\frac{b_{d1_{ij}}q^{-1}+b_{d2_{ij}}q^{-2}+...+b_{dn_{ij}}q^{-n_{ij}}}{1+a_{d1_{ij}}q^{-1}+...+a_{dn_{ij}}q^{-n_{ij}}},\ i=1,2,...,r,\ \ j=1,2,...,m,$$

where $b_{d1_{ij}}, b_{d2_{ij}},...,b_{dn_{ij}}$, and $a_{d1_{ij}}, a_{d2_{ij}},...,a_{dn_{ij}}$ are the local model parameters. For each sample the combination of the local models is used to model the global plant behavior. A PID controller is designed for each local model.

The set of PID controllers forms the controller bank in the scheme presented in Fig. 1. The control signal is obtained as weighting sum of the local controllers control signals

(3) $$u(k)=\mu_1u_1(k)+\mu_2u_2(k)+...+\mu_qu_q(k)\,,$$

where $u_i(k),\ i=1,2,...,q,$ are the local controllers control signals and $\mu_i,\ i=1,2,...,q,$ are the normalized weighting coefficients

(4) $$\sum_{i=1}^{q}\mu_i=1\,.$$

The description of the *j*-th digital PID controller is given by:

(5) $$u_j(k)=u_{p_j}(k)+u_{\text{int}_j}(k)+u_{dj}(k)\,,$$

(6) $$u_{p_j}(k)=K_{p_j}\left[d_jr(k)-y(k)\right],$$

(7) $$u_{\text{int}_j}(k)=u_{\text{int}_j}(k-1)+b_{\text{int}1j}\left[r(k)-y(k)\right]+b_{\text{int}2_j}\left[r(k-1)-y(k-1)\right],$$

(8) $$u_{d_j}(k)=a_{d_j}u_{d_j}(k-1)+b_{d_j}[c_jr(k)-c_jr(k-1)-y(k)+y(k-1)],$$

where $b_{\text{int}1_j}=K_{p_j}\dfrac{T_0}{T_{\text{int}_j}}$, $b_{\text{int}2_j}=0$, $a_{d_j}=\dfrac{T_{d_j}}{T_{d_j}+N_jT_0}$, $b_{d_j}=K_{p_j}\dfrac{T_{d_j}N_j}{T_{d_j}+N_jT_0}$, $r(k)$

is the reference signal, K_{p_j} – the proportional gain of the j-th PID controller, T_{int_j} – integral time of the j-th PID controller, T_{d_j} – derivative time of the j-th PID controller, $\frac{T_{d_j}}{N_j}$ – a time constant of the j-th first-order low pass filter, T_0 – the sample time, d_j, c_j – weighting coefficients of the j-th PID controller.

The block named "Supervisor" determines on-line the current value of the weighting coefficients according to the proposed MMAC algorithm. In the algorithm a general model output $\tilde{y}$ is used. It is formed as weighting sum of the local models outputs

(9) $$\tilde{y}^{\mathrm{T}} = \mu^{\mathrm{T}} y^{\mathrm{T}},$$

where

(10) $$\mu = \begin{bmatrix}\mu_1 & \mu_2 & \dots & \mu_q\end{bmatrix}^{\mathrm{T}}$$

is a vector containing the normalized weighting coefficients, and

(11) $$y = \begin{bmatrix}y^1 & y^2 & \dots & y^q\end{bmatrix}$$

is a $r \times q$ matrix containing the local model outputs y^i, $i = 1, 2, \dots, q$. The equation (4) can be described in the form

(12) $$\mu^{\mathrm{T}} \underline{1} = 1,$$

where

$$\underline{1} = [\underbrace{1 \quad 1 \quad \dots \quad 1}_{q}]^{\mathrm{T}}.$$

The general output error $\tilde{e}$ is given by

(13) $$\tilde{e}^{\mathrm{T}} = y_{\mathrm{o}}{}^{\mathrm{T}} - \tilde{y}^{\mathrm{T}}.$$

Taking into account equations (9) and (12) the general output error can be expressed as

(14) $$\tilde{e}^{\mathrm{T}} = \mu^{\mathrm{T}} \underline{1} y_{\mathrm{o}}^{\mathrm{T}} - \mu^{\mathrm{T}} y^{\mathrm{T}} = \mu^{\mathrm{T}} [\underline{1} y_{\mathrm{o}}^{\mathrm{T}} - y^{\mathrm{T}}] = \mu^{\mathrm{T}} e,$$

where

$$e = [e_1 \quad e_2 \quad \dots \quad e_q]^{\mathrm{T}}$$

is a $q \times r$ matrix containing the errors

(15) $$e_i = y_{\mathrm{o}} - y^i, \; i = 1, 2, \dots, q,$$

between the plant output and the local model output.

The main contribution of the paper is the objective function used for determination of the weighting coefficients. This function is defined as

(16) $$J(\mu) = \frac{1}{2} \sum_{i=1}^{k} \tilde{e}^{\mathrm{T}}(i) \tilde{e}(i).$$

From expressions (16) and (14) the following equation is obtained

(17) $$J(\mu) = \frac{1}{2}\mu^{\mathrm{T}} P^{-1}(k)\mu,$$

where

(18) $$P^{-1}(k) = \sum_{i=1}^{k} e(i)e^{\mathrm{T}}(i)$$

is a $q \times q$ matrix.

After taking into account the normalization (4), the weighting coefficients are determined from

(19) $$\min_{\mu} L(\mu, \lambda)\,,$$

where

(20) $$L(\mu,\lambda) = J(\mu) + \lambda[\mu^{\mathrm{T}}\underline{1} - 1]\,,$$

and λ is the Lagrange gain. The necessary conditions for the extremum of (20) are

(21) $$\nabla L_{\mu} = 0\,, \nabla L_{\lambda} = 0\,,$$

where ∇L_{μ} is the gradient with respect to c and ∇L_{λ} is the gradient with respect to λ. After differentiation of (20), according to (21) one obtains

(22) $$P^{-1}(k)\mu + \lambda\underline{1} = 0, \quad \mu^{\mathrm{T}}\underline{1} = 1\,.$$

Thus, the vector μ can be obtained from

(23) $$\mu = -P(k)\lambda\underline{1}\,.$$

After multiplying (23) to the left by $\underline{1}^{\mathrm{T}}$ one finds

(24) $$\underline{1}^{\mathrm{T}}\mu = -\underline{1}^{\mathrm{T}}P(k)\underline{1}\lambda\,.$$

Thus, taking into account the normalization (4), the Lagrange gain can be expressed as

(25) $$\lambda = -\frac{1}{\underline{1}^{\mathrm{T}}P(k)\underline{1}}\,.$$

After substituting (25) into (23), the vector μ is determined from

(26) $$\mu = \frac{P(k)\underline{1}}{\underline{1}^{\mathrm{T}}P(k)\underline{1}}\,.$$

The matrix $P(k)$ is inverse of the one defined by (18). The equation (18) can be expressed as

(27) $$P^{-1}(k) = \sum_{i=1}^{k-1} e(i)e^{\mathrm{T}}(i) + e(k)e^{\mathrm{T}}(k) = P^{-1}(k-1) + e(k)e^{\mathrm{T}}(k)\,.$$

It is important to note that the equation (27) is the same as the one for the inverse covariance matrix in the RLS algorithm. The determination of the weighting coefficients current values requires real time computation of the matrix $P(k)$ rather than of the matrix $P^{-1}(k)$. Thus it is necessary to derive a recursive equation for the computation of $P(k)$. Then the matrix inverse lemma is useful [17]. After applying the matrix inverse lemma to equation (27) one obtains

(28) $$P(k) = P(k-1) - P(k-1)e(k)[I_r + e^{\mathrm{T}}(k)P(k-1)e(k)]^{-1}e^{\mathrm{T}}(k)P(k-1),$$

where I_r is the unit matrix. The current value of μ is determined from equation (26) where the current value of $P(k)$ is determined from equation (28). As can be seen for the single output plant the equation (28) is the same as one for the covariance matrix in the RLS algorithm if the regressors are substituted by the error $e(k)$. It is well known that the determination of $P(k)$ according to equation (28) makes the recursive algorithm insensitive to the plant parameters changes. There are many useful modifications of RLS that modify the covariance matrix equation in order to keep algorithm's sensitivity. The main advantage of the proposed MMAC algorithm is that most of these RLS modifications are applicable for equation (28).

In this paper four well known modifications for covariance matrix determination are used. Their covariance matrix equations are presented in Table 1 [18].

Table 1

Name	Covariance matrix equation
RLS with regularization of $P(k)$	$P(k) = \left(1 - \frac{c_{\min}}{c_{\max}}\right)P(k-1) -$ $- P(k-1)e(k)[I_r + e^{\mathrm{T}}(k)P(k-1)e(k)]^{-1}e^{\mathrm{T}}(k)P(k-1) + c_{\min}I,$ where $0 \le c_{\min} < \lambda_{\min}(P(k)) < \lambda_{\max}(P(k)) \le c_{\max})$
RLS with dependent update of $P(k)$	$\overline{P}(k) = P(k-1) - P(k-1)e(k)[I_r + e^{\mathrm{T}}(k)P(k-1)e(k)]^{-1}e^{\mathrm{T}}(k)P(k-1)$, $P(0) = P_0$, $P(k) = \overline{P}(k) + cI$, if $\mathrm{trace}(\overline{P}(k)) \le c_{\min}$, $P(k) = \overline{P}(k) - cI$, if $\mathrm{trace}(\overline{P}(k)) \ge c_{\max}$, $P(k) = \overline{P}(k)$, if $c_{\min} < \mathrm{trace}(\overline{P}(k)) < c_{\max}$
RLS with directional forgetting(only for SISO system)	$P(k) = P(k-1) - P(k-1)e(k)[\varepsilon^{-1} + e^{\mathrm{T}}(k)P(k-1)e(k)]^{-1}e^{\mathrm{T}}(k)P(k-1)$, $\varepsilon(k-1) = \lambda' - \frac{1-\lambda'}{e^{T}(k)P(k-1)e(k)}$, where λ' can be chosen as in RLS with exponential forgetting algorithm
RLS with exponential forgetting	$P(k) = \frac{1}{\lambda}[P(k-1) - P(k-1)e(k)[I_r + e^{\mathrm{T}}(k)P(k-1)e(k)]^{-1}e^{\mathrm{T}}(k)P(k-1)]$

3. Pseudo code of MMAC algorithm

The pseudo code of the MMAC algorithm involves the following steps.

Step 1. Input initial conditions for the design process and choice of a limited model set.

This step is performed off-line and it includes:

- *Choice of the model number q.* This is a very important task. MMAC offers an approach to model the complex plant dynamics by combination of simple local models. MMAC algorithm will ensure control system performance if the model set represents adequately the plant dynamics. On the other hand, utilization of unnecessary large number of models and controllers cannot guarantee the control performance. There are no general rules for the local model choice. The solutions for the particular tasks based on the state space models can be found in [19, 20]. The amount of the selected models is usually related to the operating condition at which the control system is expected to work. If the value intervals of the plant parameters changes are unknown, then the parameters of the local models can be estimated by an identification procedure for the time variant plant.

- *Choice of the sample time* T_0,

- *Choice of the weighting coefficients initial values*. The initial values of the weighting coefficients are usually chosen as $\mu_j(0) = \frac{1}{q},\ j = 1, 2, ..., q.$

- *Tuning of a limited set of local PID controllers.*

Each local PID controller is tuned off-line according to the corresponding local model in the model bank.

- *Choice of the initial value of the covariance matrix.*

The initial value of the covariance matrix has to be chosen in a similar manner as the one in the corresponding RLS algorithm.

- *Set the zero initial conditions for the local models and local controllers.*

Step 2. The output $y^i(k)$ of each local model is determined from equations (2).

Step 3. The error e_i of each local model is determined from equation (15)

Step 4. The covariance matrix $P(k)$ is determined from one of the equations presented in Table 1.

Step 5. The weighting coefficients c_i, $i = 1, 2, ..., q,$ are evaluated according to equation (26).

Step 6. The control signal of each local PID controller is determined from the equations (5).

Step 7. The general control signal $u(k)$ is calculated from equation (3).

4. Simulation results

The performance of the proposed MMAC algorithm in presence of measurement noise is tested by several simulated experiments. For this purpose software working in *MATLAB* and *Simulink* environment is developed. The MMAC algorithm performance is investigated in comparison with the control system based on the conventional PID controller tuned for an average plant model.

Example 1

The time variant SISO plant is described by equation (1). The transfer function is given by

$$W(s,t)=\frac{K(t)(-T(t)s+1)}{(s+1)(0.7s+1)(0.8s+1)}, \tag{29}$$

where $K(t)\in[2 \quad 5]$ and $T(t)\in[0.5 \quad 1]$. The measurement noise $\xi(t)$ is zero mean white Gaussian noise with covariance $D_{\xi}=0.03^2$. It is supposed that the plant dynamics can be approximated by three local models. Their transfer functions are chosen as:

Model 1: $$W(s)=\frac{2(-0.5s+1)}{(s+1)(0.7s+1)(0.8s+1)},$$

Model 2: $$W(s)=\frac{3.5(-0.75s+1)}{(s+1)(0.7s+1)(0.8s+1)},$$

Model 3: $$W(s)=\frac{5(-s+1)}{(s+1)(0.7s+1)(0.8s+1)}.$$

The sample time is chosen as 0.25 s. During the simulation the reference signal and the parameters of transfer function (29) are varied as follows

$$r(t)=\begin{cases}1.5 \text{ for } 0\le t<100\text{ s},\ 200\le t<300\text{ s},\ 400\le t<500\text{ s},\ 600\le t<700\text{ s},\\ \quad 800\le t<900\text{ s}, 1000\le t<1100\text{ s},\\ 0.5 \text{ for } 100\le t<200\text{ s},\ 300\le t<400\text{ s}, 500\le t<600\text{ s}, 700\le t<800\text{ s},\\ \quad 900\le t<1000\text{ s},\end{cases}$$

$$K(t)=\begin{cases}2,\ 0\le t<280\text{ s}\\ 4, 280\le t<530\text{ s}\\ 4.5, 530\le t<730\text{ s}\\ 5, 730\le t<1100\text{ s}\end{cases},\quad T(t)=\begin{cases}0.5,\ 0\le t<280\text{ s}\\ 0.75,\ 280\le t<530\text{ s}\\ 0.85, 530\le t<730\text{ s}\\ 1, 730\le t<930\text{ s}\\ 0.6,\ 930\le t<1100\text{ s}\end{cases}.$$

It is important to note that only in the time range of 0-280 s the transfer function of the plant matches with some of the transfer functions in the model bank. The local PID controllers are tuned by optimization technique. The objective function used in the optimization procedure is given by

$$I_{\text{ise}}=\int_0^{20}(r(t)-y(t))^2\,dt.$$

The following PID controller parameters are obtained:

PID controller 1: $Kp_1=0.5157,\ T_{\text{int}_1}=1.9597,\ T_{d_1}=0.9002, c_1=1, b_1=1,\ N_1=10,$

PID controller 2: $Kp_2=0.2397,\ T_{\text{int}_2}=1.8144, T_{d_2}=0.9954, c_2=1, b_2=1, N_2=10,$

PID controller 3: $Kp_3=0.1398,\ T_{\text{int}_3}=1.8192, T_{d_3}=0.9912, c_3=1, b_3=1, N_3=10.$

The conventional PID controller is tuned for **Model 2**. In Fig. 2 the output signals of the system based on MMAC algorithm with regularization of the covariance matrix (denoted by "MMACR") and the system based on the conventional PID controller (denoted by "PID") are shown. The simulation of the MMAC algorithms are done for the following initial conditions

MMACR algorithm: $c_{\min} = 0.3,\ c_{\max} = 100,\ P(0) = 10I_3$,

MMACCI algorithm: $c_{\min} = 1,\ c_{\max} = 4,\ c = 0.2,\ P(0) = 0.1I_3$,

MMACDF algorithm: $\lambda' = 0.9,\ P(0) = 0.1I_3$.

For better visualization the output signals of the same systems within the range of 700-1000 s are depicted in Fig. 3. In Figs. 4-5 the output signals of the system based on MMACR algorithm, the system based on MMAC algorithm with a directional forgetting factor (denoted by "MMACDF") and the system based on MMAC algorithm with dependent updating of the covariance matrix (denoted by "MMACCI") are indicated. In Figs. 6-9 the control signals of the same systems as the ones shown in Figs. 2-5 are presented.

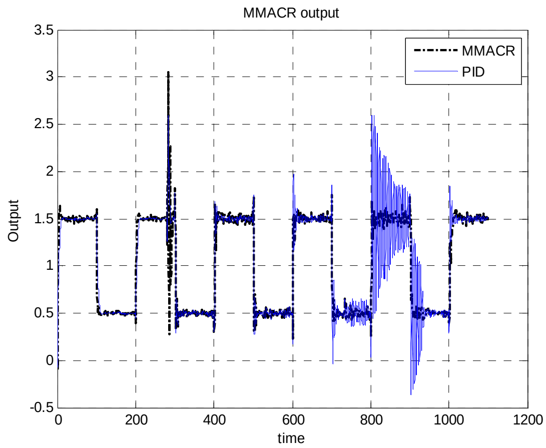

Fig. 2. Output signals of the control systems based on MMACR and PID controllers

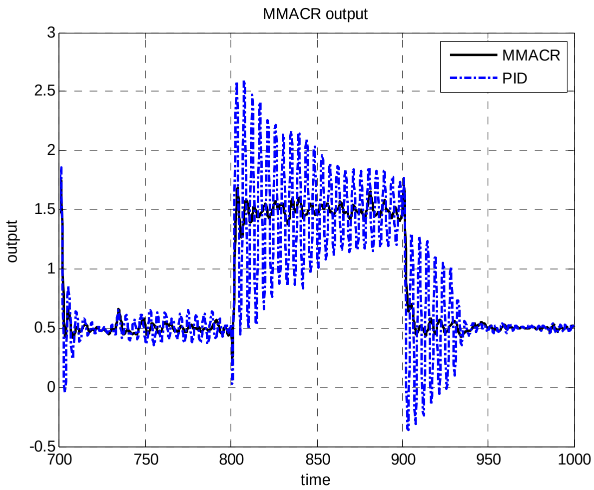

Fig. 3. Output signals of systems based on MMACR and PID controllers in the range 700-1000 s

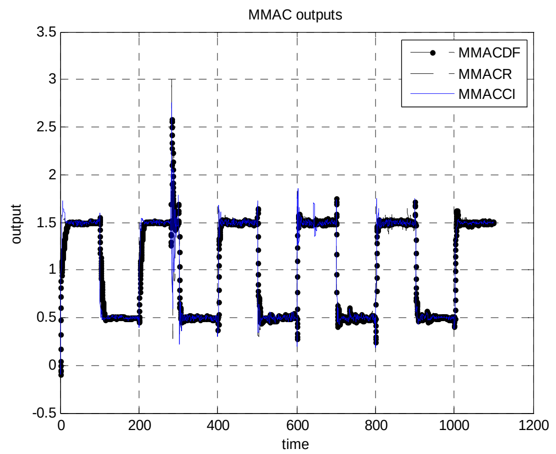

Fig. 4. Output signals of systems based on MMACR, MMACDF and MMACCI controllers

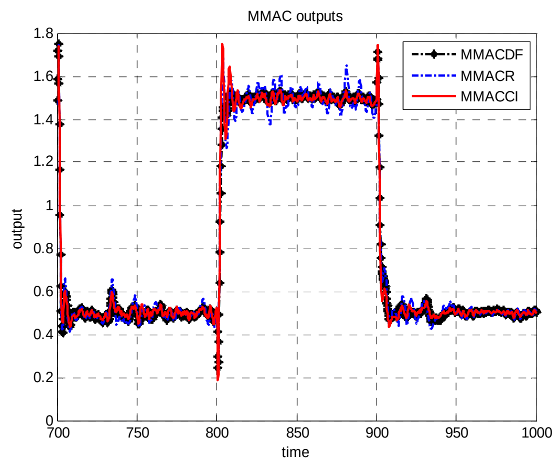

Fig. 5. Outputs of systems based on MMACR, MMACDF and MMACCI controllers in the range 700-1000 s

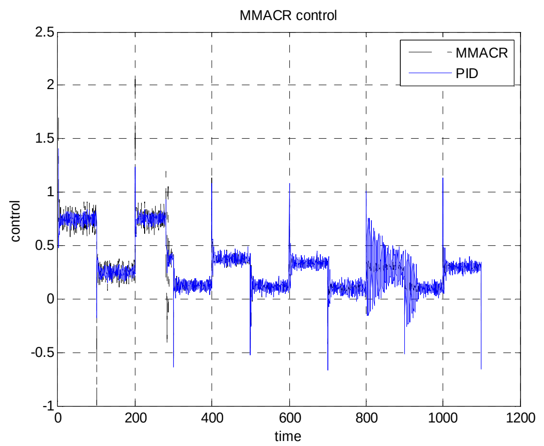

Fig. 6. Control signals of systems based on MMACR and PID controllers

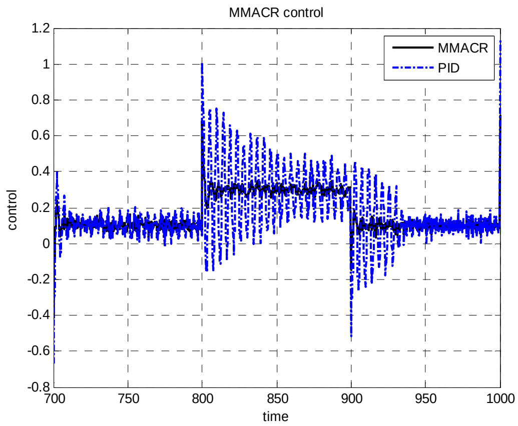

Fig. 7. Control signals of systems based on MMACR and PID controllers in the range 700-1000 s

It is seen from the figures that the performance of the systems based on all MMAC algorithms is better than this of the system based on a PID controller. The "PID" system response has large oscillations in the time range of 700-1000 s where the plant gain is higher than this used for the PID controller tuning. In the same range the systems based on MMAC algorithms kept their performance. The settling time of the systems based on all MMAC is considerably smaller than this of the system based on a PID controller. Furthermore, the "PID" system cannot work the reference in the time of 800-900 s. The "PID" system performance is good in the range 280-530 where the plant model is close to the one used for PID controller tuning. The results in Figs. 4-5 show that the output signals of the systems based on MMACR and MMACDF have smaller oscillations than those of the system based on MMACCI. Furthermore, the output response of MMACDF system is without overshoot in more cases. The MAACCI maximum deviation is greater than this of MMACR and MMACDF when the plant gain changes from 2 up to 4.

In order to characterize more precisely the dynamic behaviour of the control systems their maximal overshoot $\sigma_{\max}$ in the range 0-1100 s and the square mean error are computed. The square mean error e_{ise} is determined as

$$e_{\text{ise}} = \frac{1}{T}\int_{0}^{T}(r(t) - y(t))^2\,dt\,.$$

The computed performance indices are shown in Table 2.

Table 2. Mean square error and maximal overshoot of the control systems

Indices	MMACR	MMACCI	MMACDF	PID
e_{ise}	0.0323	0.0294	0.0319	0.0545
$\sigma_{\max}$, %	8	7.5	9	172

The indices presented in Table 2 point out the advantages of the proposed MMAC algorithms. The maximal overshoot of MMACR, MMACCI and MMACDF is approximately 9 times smaller than the corresponding value for PID.

The mean square error of the proposed algorithms is approximately 50% smaller than the corresponding value of the PID.

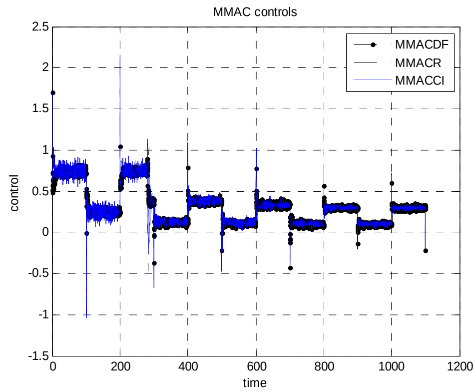

Fig. 8. Control signals of systems based on MMACR, MMACDF and MMACCI controllers

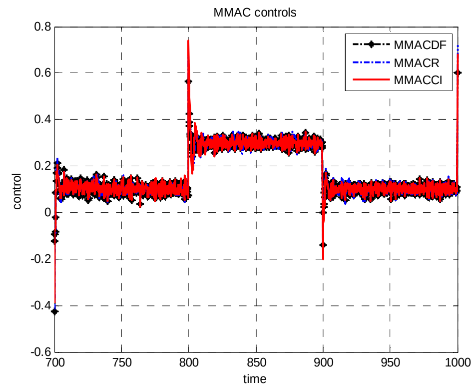

Fig. 9. Control signals of systems based on MMACR, MMACDF and MMACCI algorithms in the range of 700-1000 s

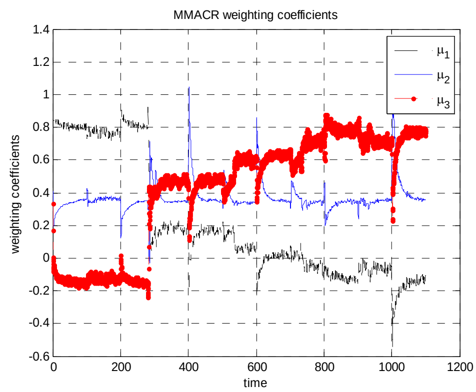

Fig. 10. Weighting coefficients of systems based on MMACR

In Figs. 10-12 the weighting coefficients of the system based on MMAC algorithms are shown. As can be seen from the figures, the value of coefficient μ_1 for MMACR and MMACCI is close to 1 in the time range 0-280 s where the plant has the same transfer function as the one of **Model 1**. The weighting coefficient of

MMACR and MMACCI are reevaluated faster after each change of the parameters $\tilde{k}$ and/or $\tilde{T}$ than the corresponding values of MMACDF. The value of coefficient μ_3 is close to 1 in the time range of 750-1100 s where the plant parameters are close to the ones of **Model 3.**

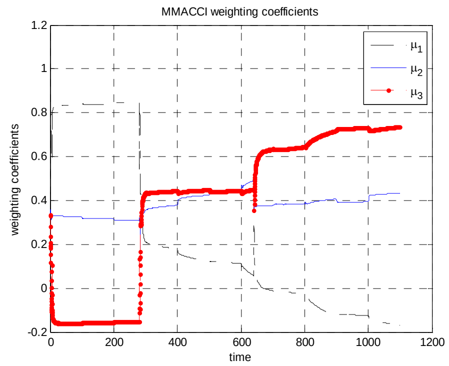

Fig. 11. Weighting coefficients of systems based on MMACCI

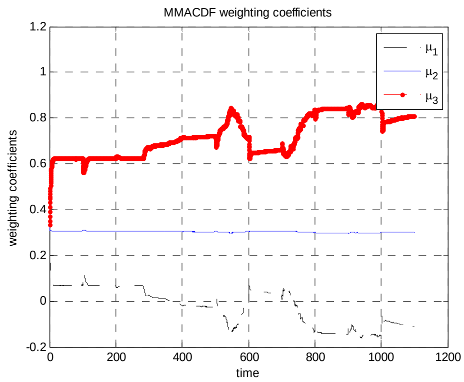

Fig. 12. Weighting coefficients of systems based on MMACDF

Example 2

The time variant two input two output plant is described by equation (1). Its transfer matrix has the form

$$W(s,t) = \begin{bmatrix} W_{11}(s,t) & W_{12}(s,t) \\ W_{21}(s,t) & W_{22}(s,t) \end{bmatrix}.$$

The elements of $W(s,t)$ are given by

$$W_{11}(s,t) = \frac{(k_1 \tilde{k}_2 k_3 (T_1 s+1)(\tilde{T}_2 s+1)(T_3 s+1)+1)\tilde{k}_4}{(T_1 s+1)(\tilde{T}_2 s+1)(T_3 s+1)(T_4 s+1)},$$

$$W_{12}(s,t) = \frac{\tilde{k}_2 k_3 \tilde{k}_4}{(\tilde{T}_2 s+1)(T_3 s+1)(T_4 s+1)},\quad W_{21}(s,t) = \frac{k_1 \tilde{k}_2}{(T_1 s+1)(\tilde{T}_2 s+1)},$$

$$W_{22}(s,t) = \frac{\tilde{k}_2}{(\tilde{T}_2 s+1)},$$

where $k_1 = 1$, $\tilde{k}_2 \in [2 \quad 5]$, $k_3 = 0.5$, $\tilde{k}_4 \in [2 \quad 3]$, $T_1 = 1$, $\tilde{T}_2 \in [0.5 \quad 1]$, $T_3 = 0.7$ and $T_4 = 0.7$. The measurement noise $\xi(t)$ is zero mean white Gaussian noise with covariance $D_\xi = 0.01^2 I_2$. It is supposed that the plant dynamics can be approximated with the help of three local models. Their parameters are chosen as:

Model 1: $k_1 = 1,\ k_2 = 2,\ k_3 = 0.5,\ k_4 = 3,\ T_1 = 1,\ T_2 = 0.5,\ T_3 = 0.7,\ T_4 = 0.7,$

Model 2: $k_1 = 1,\ k_2 = 3.5,\ k_3 = 0.5,\ k_4 = 2,\ T_1 = 1,\ T_2 = 0.75,\ T_3 = 0.7,\ T_4 = 0.7,$

Model 3: $k_1 = 1,\ k_2 = 5,\ k_3 = 0.5,\ k_4 = 3,\ T_1 = 1,\ T_2 = 1,\ T_3 = 0.7,\ T_4 = 0.7.$

Two PID controllers are tuned for each local model. The first one of them is based on a feedback from the first output and the second one is based on a feedback from the second output. The conventional PID controllers are tuned for a model with parameters as follows:

$$k_1 = 1,\ k_2 = 3,\ k_3 = 0.5,\ k_4 = 2.5,\ T_1 = 1,\ T_2 = 0.5,\ T_3 = 0.7,\ T_4 = 0.7.$$

The sample time is chosen as 0.25 s. During simulation the reference signal and the parameters of the transfer matrix vary as follows:

$$r_1(t) = r_2(t) = \begin{cases} 1.5 \text{ for } 0 \le t < 100 \text{ s},\ 200 \le t < 300 \text{ s}, 400 \le t < 500 \text{ s}, 600 \le t < 700 \text{ s}, \\ \qquad 800 \le t < 900 \text{ s}, 1000 \le t < 1100 \text{ s}, \\ 0.5 \text{ for } 100 \le t < 200 \text{ s},\ 300 \le t < 400 \text{ s}, 500 \le t < 600 \text{ s}, \\ \qquad 700 \le t < 800 \text{ s}, 900 \le t < 1000 \text{ s}, \end{cases}$$

$$\tilde{k}_2 = \begin{cases} 2,\ 0 \le t < 280 \text{ s} \\ 4,\ 280 \le t < 530 \text{ s} \\ 4.5,\ 530 \le t < 730 \text{ s} \\ 5,\ 730 \le t < 1100 \text{ s} \end{cases},\quad \tilde{k}_4 = \begin{cases} 2,\ 0 \le t < 280 \text{ s} \\ 2.5,\ 280 \le t < 530 \text{ s} \\ 3,\ 530 \le t < 730 \text{ s} \\ 2.7,\ 730 \le t < 1100 \text{ s} \end{cases},\quad \tilde{T}_2 = \begin{cases} 0.5,\ 0 \le t < 280 \text{ s} \\ 0.8,\ 280 \le t < 530 \text{ s} \\ 1,\ 530 \le t < 1100 \text{ s} \end{cases}.$$

The local PID controllers are tuned by optimization technique. The objective function is given by

$$I_{\text{ise}} = \int_0^{20} (r_1(t) - y_1(t))^2\, dt + \int_0^{20} (r_2(t) - y_2(t))^2\, dt .$$

The following PID controller parameters are obtained:

Model 1: PID1 $K_{p1} = 1.1610, b_{\text{int}1} = 0.1829,\ a_{d1} = -0.0126,\ b_{d1} = -0.1467,$

PID2 $K_{p1} = 0.6922, b_{\text{int}1} = 0.3805,\ a_{d1} = 0.0130,\ b_{d1} = 0.0899$,

Model 2: PID1 $K_{p2} = 1.7463, b_{\text{int}1} = 0.0168,\ a_{d1} = -0.0117,\ b_{d1} = -0.2042$,

PID2 $K_{p2} = 0.6891, b_{\text{int}1} = 0.1642,\ a_{d1} = 0.0126,\ b_{d1} = 0.0871,$

Model 3: PID1 $K_{p3} = 1.2062, b_{\text{int}1} = 0.1294,\ a_{d1} = -0.0128,\ b_{d1} = -0.1549,$

PID2 $K_{p3} = 0.6480, b_{\text{int}1} = 0.1783,\ a_{d1} = 0.0124,\ b_{d1} = 0.0801.$

Conventional PID controllers:

PID1 $K_p = 0.5780, b_{\text{int}1} = 0.0315,\ a_{d1} = 0.0039,\ b_{d1} = 0.0223,$

PID2 $K_p = 0.6822, b_{\text{int}1} = 0.0118,\ a_{d1} = -0.0153,\ b_{d1} = -0.1042.$

The simulation of MMAC algorithm operation is done according to the initial conditions

MMACR algorithm: $c_{\min} = 0.3$, $c_{\max} = 100$, $P(0) = 10I_3$,

MMACCI algorithm: $c_{\min} = 0.1$, $c_{\max} = 10$, $c = 1$, $P(0) = I_3$,

MMACEF algorithm: $\lambda = 0.98$, $P(0) = I_3$.

In Figs. 13-14 the output signals of the system based on MMAC algorithm with regularization of the covariance matrix (denoted by "MMACR") and the system based on the conventional PID controller (denoted by "PID") are shown. In Figs. 15-16 the output signals of the system based on MMACR algorithm, the system based on MMAC algorithm with dependent updating of the covariance matrix (denoted by "MMACCI") and the system based on MMAC algorithm with exponential forgetting (denoted by "MMACEF") are depicted. In Figs. 17-18 the control signals of the same systems as the ones shown in Figs 13-16 are presented. In Fig. 19 the square error in the range 0-1100 s is presented. The control systems maximal overshoots, the square errors and settling times are shown in Tables 3-5.

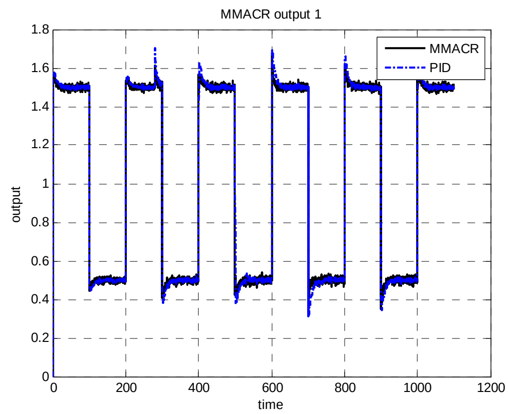

Fig. 13. First output of the control systems based on MMACR and PID controllers

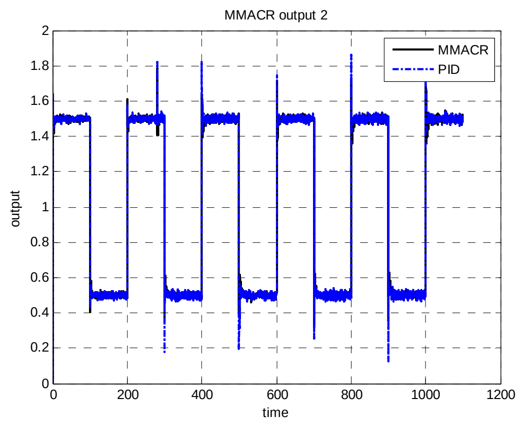

Fig. 14. Second output of the control systems based on MMACR and PID controllers

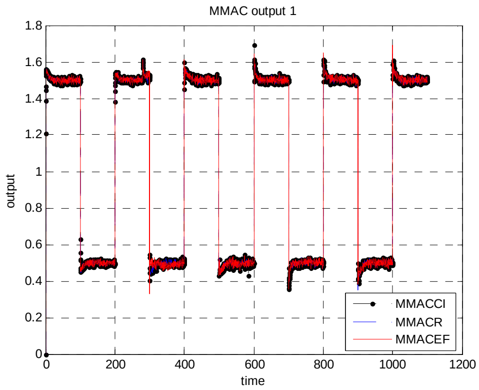

Fig. 15. First output of the control systems based on MMACR, MMACI and MMACEF

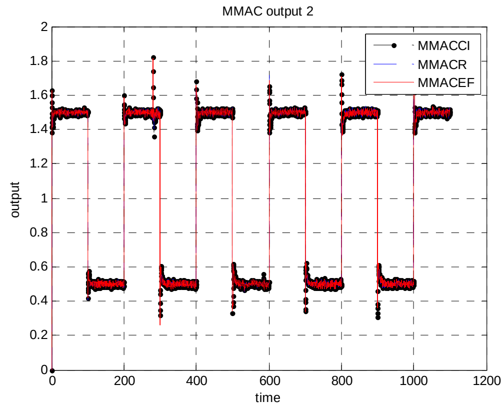

Fig. 16. Second output of the control systems based on MMACR, MMACI and MMACEF

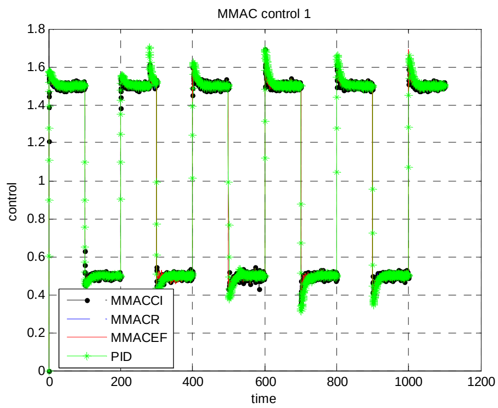

Fig. 17. First control signal of the systems based on MMACR, MMACI, MMACEF and PID

It is seen from the figures that the performance of the systems based on all MMAC algorithms is better than this of the system based on a PID controller. The systems based on all MMAC algorithms have a step response with sufficiently small overshoot (except MMACEF in the time range of 300-400 s) and a settling time in the range of 0-1100 s.

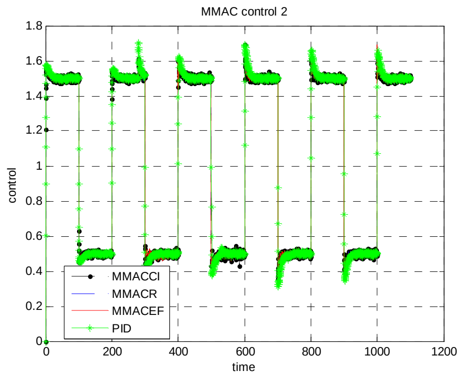

Fig. 18. Second control signal of the systems based on MMACR, MMACI, MMACEF and PID

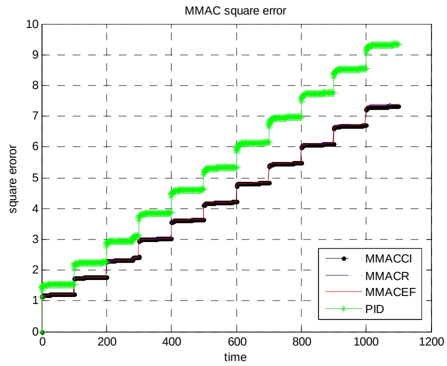

Fig. 19. Square error of the systems based on MMACR, MMACI, MMACEF and PID in the range 0-1100 s

Table 3. Overshoot of the control systems

Time range	MMACR	MMACCI	MMACEF	PID
300-400	18	20	34	24
400-500	4.67	6.67	7.33	8
500-600	15	15	15	25
600-700	10.67	11.27	9.67	12.67
800-900	7.33	7.35	10	10.67

Table 4. Settling time of the control systems

Time range	MMACR	MMACCI	MMACEF	PID
300-400	20	20	5	32
500-600	20	20	20	30
600-700	15	15	15	30
700-800	20	20	20	40
900-1000	15	15	20	30

Table 5. Square error of the control systems

Time range	MMACR	MMACCI	MMACEF	PID
0-300	2.43	2.42	2.42	3.12
0-500	3.635	3.634	3.635	4.62
0-700	4.84	4.84	4.84	6.15
0-800	5.49	5.48	5.48	7
0-1100	7.41	7.3	7.38	9.334

The indices presented in Tables 3-5 point out the advantages of the proposed MMAC algorithms. The overshoot of MMACR, MMACCI and MMACEF is smaller than the corresponding value for a PID. In almost all ranges the overshoot of MMACR is smaller than the corresponding value for MMACCI and MMACEF and considerably smaller than the corresponding value of PID. As can be seen from the results presented in Fig. 19 and Table 5 the square error of the proposed algorithms is approximately 25 % smaller than the corresponding value of PID in all ranges. The settling time of MMACR, MMACCI and MMACEF is 50-100 % smaller than the corresponding value of PID.

In Figs. 20-22 the weighting coefficients of the system based on MMAC algorithms are shown. It is seen that no value of the weighting coefficients is converging to 1 in the range 0-1100 s. This is due to the fact that the plant parameters do not coincide with the corresponding values of the models in the model bank. Nevertheless, the performance of the control system based on MMAC algorithms is kept. The weighting coefficient of MMACR and MMACCI are reevaluated faster after each change of the parameters than the corresponding values of MMACEF.

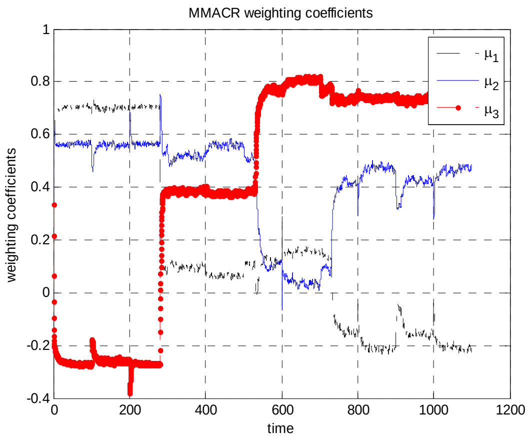

Fig. 20. Weighting coefficients of systems based on MMACR

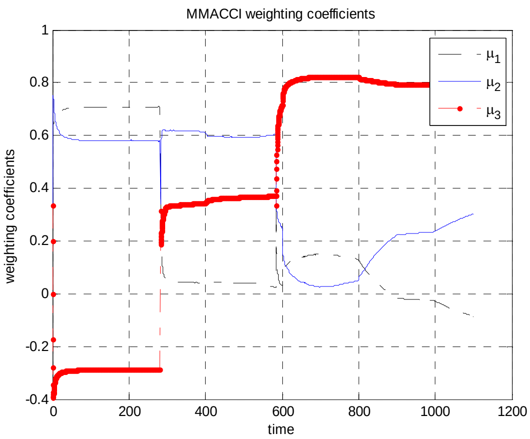

Fig. 21. Weighting coefficients of systems based on MMACCI

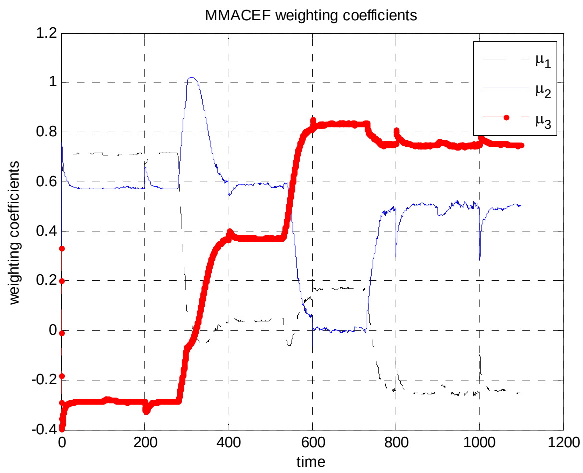

Fig. 22. Weighting coefficients of systems based on MMACR

5. Conclusion

In this paper a Multiple Model Adaptive Control (MMAC) algorithm for control of a time-variant plant in the presence of measurement noise is proposed. This algorithm controls the plant using a bank of PID controllers designed on the base of time invariant input-output models. The control signal is formed as weighting sum of the control signals of local PID controllers. The main contribution of the paper is the objective function minimized to determine the weighting coefficients. The proposed algorithm minimizes the sum of the square general error between the model bank output and the plant output. The equation for on-line determination of the weighting coefficients is obtained. They are determined by the current value of the general error covariance matrix. The main advantage of the algorithm is that the derived general error covariance matrix equation is the same as this in the recursive least square algorithm (RLS). Thus, most of the well known RLS modifications for the tracking time-variant parameters can be directly implemented in the suggested algorithms. Four well known RLS modifications (RLS with regularization, RLS with dependent updating, RLS with directional forgetting and RLS with exponential forgetting) are implemented. The algorithm performance is tested by simulation. For this aim software in *Matlab/Simulink* environment is developed. Simulation experiments with both SISO and MIMO time variant plants are carried out. Comparison between the control systems based on the developed MMAC algorithms and the control system based on a conventional PID controller tuned for average plant model, is performed. The results show the advantages of MMAC algorithms over the conventional PID. In more of the time ranges the evaluated performance indices are significantly smaller for the systems based on MMAC algorithms than the corresponding values for the system based on a single PID controller.

References

1. Narendra, K., S. Balakrishnan. Adaptive Control Using Multiple Models. – IEEE Trans. on AC, **37**, 1997, 191-194.
2. Li, X. R., Y. Bar-Shalom. Multiple-Model Estimation with Variable Structure. – IEEE Trans. AC, **41**, 1996, No 4, 478-493.
3. Murray-Smith, R. and T. A. Johansen. Multiple Model Approaches to Modelling and Control. London, Taylor & Francis, 1997.
4. Maybeck, P., G. Griffin, Jr.MMAE/MMAC Control for Bending with Multiple Uncertain Parameters. – IEEE Trans. AES, **33**,1997, No 3, 903-912.
5. Garipov, E., T. Puleva, E. Haralanova. Adaptive Control of Complex Plants by a Set of Input Output Models. Introduction to modern Robotics II. Chapter 10. Australia, Hong Kong, Iconcept Press, Ltd., 2011.
6. Magill, D. Optimal Adaptive Estimation of Sampled Stochastic Processes. – IEEE Trans. AC, **10**, 1965, 434-439.
7. Blom, H., Y. Bar-Shalom. The IMM Algorithms for Systems with Markovian Switching Coefficients. – IEEE Trans. AC, **33**, 1988, 780-783.
8. Semerdjiev, C., V. Jilkov, L. Mihailova. Multiple Model Estimation and Control of Stochastic Systems. – Automatics and Informatics, Vol. **5**, 2002, 3-15 (in Bulgarian).
9. Anderson, B. D. O., T. S. Brinsmead, D. Liberzon, A. S. Morse. C. Multiple Model Adaptive Control with Safe Switching. – International Journal of Adaptive Control and Signal Processing, August 2001, 446-470.
10. Hespanha, J. P., A. S. Morse. Switching Between Stabilizing Controllers. – Automatica, **38**, 2002, No 11, 1905-1917.
11. Hespanha, J. P., D. Liberzon, A. S. Morse. Hysteresis-Based Supervisory Control of Uncertain Linear Systems. – Automatica, **39**, 2003, No 2, 263-272.
12. Garipov, E., T. Stoilkov. Adaptive Control by a Set of Input Output Models. – In: Proc. of International Conference "Automatics and Informatics'06", 2006, 13-19.
13. Yoona, T. W., J. S. Kim, A. S. Morse. Supervisory Control using a New Control-Relevant Switching. – Automatica, **43**, 2007, 1791-1798.
14. Garipov, E., T. Stoilkov. Multiple Model Approach for Time-Variant Plant Control. – In: Proc. of 7th Int. Conf. on Challenges in Higher Education and Research in the 21st Sentury, 2009, Sozopol, Bulgaria, 118-124.
15. Garipov, E., T. Stoilkov, I. Kalaykov. Multiple-Model Dead-beat Controller in Case of Control Signal Constraints. – In: Fourth Int. Conf. on Informatics, Automation and Robotics, Angers, France.
16. Madjarov, N., L. Mihaylova. Adaptive Multiple Model Algorithms for Estimation and Control of Stochastic Systems. – Automatics and Informatics, **2**, 1998, 16-23 (in Bulgarian).
17. Ljung, L. System Identification: Theory for the User. 2nd Ed. Englewood Cliff, NJ, Prentice-Hall, 1999.
18. Garipov, E. System Identification. Technical University of Sofia, 1997 (in Bulgarian).
19. Li, X. R. Optimal Selection of Estimate for Multiple Model Estimation with Uncertain Parameters. – IEEE Trans. on AES, **34**, 1998, No 2, 653-657.
20. Sheldon, S., P. Maybeck. An Optimizing Design Strategy for Multiple Model Estimation and Control. – IEEE Trans. on AC, **38**, 1993, 651-654.