An Adaptive Internal Model-based Neural Controller with Embedded Integral Action

Milorad Bozic, Petar Maric and Igor Krcmar

Abstract- It is well known that the PID controllers are used in the most of industry applications (in more than 95% cases according to relevant reports and analyses). Further, it is a fact that these controllers are rarely without integral action. Also, Internal Model Control (IMC) based designs are well recognized and accepted inside control community thanks to their stability and performance robustness. Having in mind nonlinear and slowly varying dynamics of the industrial processes, in this paper we propose an Adaptive Internal Model-based Neural Controller with embedded Integral action (AIMNCI). The internal model of the controlled plant is implemented by the Fast Clustered Radial Basis Function Network (FCRBFN), which is equipped with the Stochastic Gradient Descent (SGD) learning algorithm. Some illustrations and performance comparison between the proposed AIMNCI controller and others are given by two examples.

***Index Terms*— adaptive control, neural networks, nonlinear internal model control, process control, zero steady-state error.**

I. INTRODUCTION

THE design of controllers based on the Internal Model Control (IMC) structure has been accepted as a well established approach not only in the cases of the linear system models [1], [2], [3], but also in nonlinear system control. Moreover, the design of the classical PID controllers is often based on this design approach thanks to its advantages concerning stability, robustness and accuracy in the steady-state [1], [4]. In terms of stability, one has to undertake some precautions, because it is assumed in the IMC structure that the plant model is stable. If the plant is not stable, then some kind of the PD controller in a local loop with the plant provides needed stability. In general, application of a PD controller in the local loop with the controlled plant in most of the cases improves overall dynamical behavior of that part of the feedback system. Unfortunately, this is not truth for the system accuracy. This means that system cannot eliminate the steady-state error in the case of constant reference and constant disturbances, especially when the controlled plant possesses larger nonlinearity, e.g. a plant with hysteresis. In the IMC control structure the pre-trained Neural Networks (NNs) are used to implement the nonlinear time invariant plant model [3],[5], and [6]. Further, the Fast Clustered Radial Basis Function Network (FCRBFN) [7] has been suggested for the implementation of the model of slowly varying plants [8]. Thus, design of the control system in the case of nonlinear dynamic plant requires, particular attention regarding the choice of the Q controller in the IMC system structure [1]. The Q controller should essentially provide the inverse of the plant model. If the Q controller static gain is equal to the inverse of its internal model static gain, and it provides stability of the overall system, then offset-free control is obtained for constant reference and output disturbances [3]. A simple way for an adjustment of the controller static gain was given in [8], but some problems have been noticed in the cases of strong model nonlinearity and very slow nonlinear process dynamics [9]. The Approximate Internal Model-based Neural Control (AIMNC) was proposed for unknown nonlinear discrete time processes [6], in order to achieve zero steady state error, in the presence of a constant disturbances and the possible mismatch between a real plant and its NN model. In the proposed AIMNC algorithm the NN model has been trained off line, i.e. at implementation it has a fixed architecture and parameters. Although, if complex enough the NN architecture provides good accuracy of the plant model. At the same time, it does not mean good approximation of its first derivative. The derivative is needed for calculation of the control increment in the AIMNC algorithm.

An elegant approach to obtain offset-free control of the linear system, in the case of constant reference and constant disturbance acting on the system output, is introduction of an integral action into control law [4]. Having this in mind, and the fact that the FCRBFN equipped with the SGD learning algorithm shows very good approximation capabilities in the estimation of slowly varying nonlinear processes, in this paper we incorporate integral term into the IMC control structure. The fact that dynamics of the controlled process can be slowly varying implies that NNs used in control algorithms should have at least some variable parameters, which can be tuned in real time. An essential part of adaptive control system is the parameter estimation algorithm that must have rapid convergence. It is well known that such characteristics have algorithms based on the Least Mean Squares (LMS) performance criterion, i.e. some variants of the Recursive LMS (RLMS) or the SGD algorithms. Thus, they are very suited for adaptive control system designs [10].

The proposed AIMNCI controller design is based on the FCRBFN and an estimation provided with the SGD learning algorithm of the Nonlinear Autoregressive Moving Average

M. Bozic, is with the University Banja Luka, Faculty of Electrical Engeneering, Patre 5, 78.000 Banja Luka, Bosnia and Herzegovina. (e-mail: milorad.bozic@etf.unibl.org

P. Maric is with the University Banja Luka, Faculty of Electrical Engeneering, Patre 5, 78.000 Banja Luka, Bosnia and Herzegovina. (e-mail: petar.maric@etf.unibl.org).

I. Krcmar, is with the University Banja Luka, Faculty of Electrical Engeneering, Patre 5, 78.000 Banja Luka, Bosnia and Herzegovina. (e-mail: igor.krcmar@etf.unibl.org).

with eXogenous input (NARMAX) model of the controlled process inside the IMC system structure with embedded integral action.

The rest of the paper is organized as follows. In the Section II, the plant modeling, i.e. the NN plant model and the SGD learning algorithm with a momentum term is outlined. In the Section III we explain how to incorporate an integral action into the IMC structure and propose the fully adaptive neural controller design based on that structure. Simulation results demonstrating performance of the proposed AIMNCI algorithm are presented in the Section IV. In the Section V we give conclusions of the work.

II. AN ADAPTIVE ESTIMATION ALGORITHM

A. The plant modeling

As a starting point we have assumed the following NARMAX model of the controlled plant

$$y(k+1)=f(\boldsymbol{y},\boldsymbol{u})+d(k+1), \tag{1}$$

where f stands for a nonlinear function, $\boldsymbol{y}=[y(k),y(k-1),\ldots,y(k-n+1)]^T$, $\boldsymbol{u}=[u(k),u(k-1),\ldots,u(k-m+1)]^T$, where $y(k)$ and $u(k)$represent samples, at the discrete time instant k, of the plant output and input, respectively, n denotes length of the vector $\boldsymbol{y}$, m denotes length of the vector $\boldsymbol{u}$, and $d(k)$ represents an effect of the slowly varying disturbances to the plant. Also, that effect could be consequence of a constant disturbances at the input or output of the plant, in the case of a stable plant model, and/or possible modeling errors. The NNs in control system applications are used to estimate model (1) based on available input and output data. In most cases of so called "neuro" control algorithms, some kind of the preceding NN training has been required [11], [12], [13], [14].. Further on, we consider an algorithm by which the identification and control are carried out in real time, simultaneously.

As we know from adaptive control systems based on linear models, the identification part of the adaptive control algorithms must have fast convergence [10]. It has been demonstrated through many examples that the used SGD algorithm with momentum parameter exhibits very good performance [8], [9], [15]. It is also worth mentioning that this is achieved partially due to the appropriate NN architecture, i.e. the FCRBFN in our case.

B. An adaptive estimation algorithm based on the FCRBFN

For the sake of completeness and easier tracking of the results outlined below, we shall give here only the basic terms relating to the on-line estimation algorithm. The FCRBFN is used as an internal plant model [7], [8], [9], whose output is a prediction of the plant given by (1) at sample instant k+1

$$\hat{y}(k+1)=\boldsymbol{W}^T\boldsymbol{\Phi}(\boldsymbol{x},\boldsymbol{c},\boldsymbol{\sigma}), \tag{2}$$

where $\boldsymbol{x}=[\boldsymbol{y}^T,\boldsymbol{u}^T]^T$ is the input vector to the network, $\boldsymbol{c}$ and $\boldsymbol{\sigma}$ are the vectors of centers and spreads of activation functions of neurons in the hidden layer, respectively, $\boldsymbol{\Phi}$ is the vector composed of activations of hidden layer neurons and $\boldsymbol{W}$ is the weight vector of connections between hidden layer neurons of the FCRBFN model and its output. Activation function of neurons in the hidden layer of the FCRBFN is Gaussian function

$$\varphi(x)=e^{-\frac{(x-c_x)^2}{\sigma_x}} \tag{3}$$

where c_x denotes center and σ_x denotes spread of the activation function.

Hidden layer neurons of the FCRBFN are clustered according to its inputs, which implies that total number of clusters is equal to $n+m$. The ranges of the possible values of each FCRBFN input are known in advance and it is given as

$$wr_x=\max(x)-\min(x) \tag{4}$$

where x is substituted by y and u, respectively. Therefore, it is necessary to decide about their range expansions e_x (usually expressed in percents), as well as, to decide on the number of neurons n_{cy} in each of n clusters related to each of the plant output values $y(k),y(k-1),\ldots,y(k-n+1)$ used as the first n network inputs, and the number of neurons n_{cu} in each of m clusters related to each of the plant input values $u(k),u(k-1),\ldots,u(k-m+1)$ used as the next m network inputs. Choice of other parameters and distribution of activation functions within neuron clusters in the FCRBFN have been discussed in [7], [15].

Total number of neurons in hidden layer is equal to

$$n_{tn}=n\cdot n_{cy}+m\cdot n_{cu}\,. \tag{5}$$

Performance criterion of the adaptive estimation algorithm defined as a square of the model output estimation error

$$\varepsilon(k)=y(k+1)-\hat{y}(k+1)=y(k+1)-\boldsymbol{W}^T\boldsymbol{\Phi}(\boldsymbol{x},\boldsymbol{c},\boldsymbol{\sigma}), \tag{6}$$

is given by

$$J(k)=\tfrac{1}{2}(\varepsilon(k))^2. \tag{7}$$

To adapt weights of the FCRBFN , according to the basic SGD algorithm, an increment of the weight vector should be

$$\Delta\boldsymbol{W}(k)=-\eta\nabla J(k)=\eta\boldsymbol{\Phi}(\boldsymbol{x},\boldsymbol{c},\boldsymbol{\sigma})\varepsilon(k), \tag{8}$$

where η denotes the learning rate parameter.

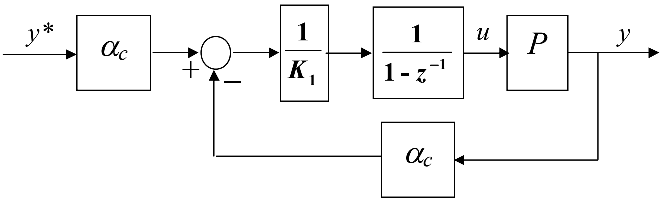

Figure 1. The classical control structure with integral controller

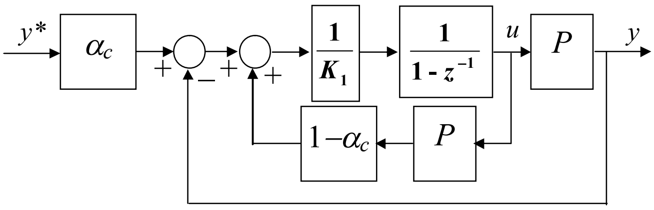

Figure 2. An equivalent of the classical control structure from Fig.1

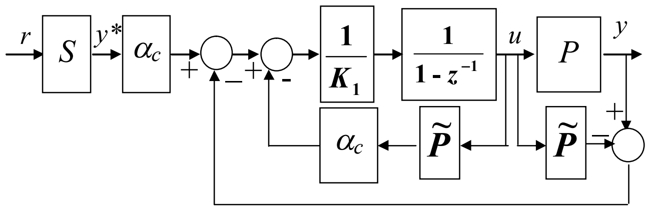

Figure 3. The proposed control structure for the adaptive neuro control

Within the analysis based on Taylor series expansion of the vector function $\boldsymbol{\Phi}(\boldsymbol{x}+\boldsymbol{\Delta x},\boldsymbol{c},\boldsymbol{\sigma})$, we have shown that in order to assure convergence of the algorithm, the learning rate parameter should be chosen according to [15]

$$\eta < \frac{2}{nn_{cy}+mn_{cu}}. \tag{9}$$

In order to improve performance of the plant model estimation in the cases where the noises and outliers are present in data, we apply the SGD algorithm with the momentum term given by

$$\Delta\boldsymbol{W}(k) = \alpha\Delta\boldsymbol{W}(k-1) + \eta\boldsymbol{\Phi}\boldsymbol{\varepsilon}(k), \tag{10}$$

where α $(0<\alpha<1)$ determines a filtering characteristic of the algorithm, and it should be appropriately chosen.

III. A Neuro Adaptive IMC Controller

A. *Embedding integral action into IMC structure*

Taking into account the fact that PID control is the ubiquitous controller applied in industry [4], and that integral action is almost unavoidable in it, we start our consideration in this section by classical control structure shown in Fig.1. In the case of stable process given by transfer funcion $P(z)$, with appropriate choice of parameters K_1 and α_c this control structure provides offset-free control for constant reference signal r and disturbance d, if the DC gain of the set point filter is $S(1)=1$. On the other hand, an equivalent block diagram of that in Fig.1 is shown in Fig.2. Finally, assuming that instead of true plant model P, on our disposal is only its approximation $\tilde{P}$ we obtain the control structure shown in the Fig.3.

Industrial processes are characterised by unmeasured disturbances, nonlinear dynamics, resolution and sensitivity limits, valve nonlinearities, and time varying process parameters. Therefore, it is needed to undertake appropriate control action to cope with such complex circumstances. In that sense we propose the control structure shown in the Fig.3 as one of possible approach to resolve these complex control problems. Wthin this control structure the FCRBFN equipped with the SGD learning algorithm (10) is used to implement the internal model of the plant $\tilde{P}$.

B. *The adaptive neuro control design*

According to the control structure in the Fig.3 the control signal follows

$$u(k) = u(k-1) + \frac{\alpha_c y^*(k) - y(k)}{K_1} - \frac{(\alpha_c - 1)\hat{y}(k)}{K_1}. \tag{11}$$

Here is $y^*(k) = S(q)r(k)$. It is clear from (11) that implementation of the proposed control algorithm requires only one FCRBFN.

Value of the parameter α_c $(0<\alpha_c<1)$ can be tuned according to stability analysis of the linearised plant model [6] and K_1 is set as an upper bound of the static gain of the control plant in its operating range.

Implementing control law defined by (11) using the FCRBFN with the on-line tuning of its weight vector given by (10) we obtained the AIMNCI controller.

Remark 1. It should be noted that in [6] value of K_1 ,the first derivative of the NN output with respect to control input, has been calculated on the base of the pre-trained NN plant model. Accuracy of calculation, mentioned in Sec. I, as well as changing plant dynamics has more deteriorating effect on the value of K_1 than on approximation of the plant model.

IV. ILLUSTRATIVE EXAMPLES AND PERFORMANCE COMPARISON

In the next examples taken from [4] (Ch.17, Data-Driven PID Controller, written by T. Yamamoto) we illustrate performance of the adaptive neural controllers proposed in this paper.
Example 1: The true plant is given by the following Hammestein models
System 1

$$y(k) = 0.6y(k-1) - 0.1y(k-2) + 1.2x(k-1) - 0.1x(k-2) + \\ + d(k) + \xi(k), \\ x(k) = 1.5u(k) - 1.5u^2(k) + 0.5u^3(k), \tag{12}$$

System 2

$$y(k) = 0.6y(k-1) - 0.1y(k-2) + 1.2x(k-1) - 0.1x(k-2) + \\ + d(k) + \xi(k), \\ x(k) = 1.0u(k) - 1.0u^2(k) + 1.0u^3(k), \tag{13}$$

where compareing to the example in [4], a disturbance term $d(k)$ is added.
The variable static gains corresponding to control signal u of System 2 are larger than those of System 1 at u>0.6 [4].
We considered the case where the true plant model changes from (12) to (13) at k=70, when the white Gaussian noise $\xi(k)$ with zero mean and a variance 0.01^2 and disturbance $d(k)$ as shown on Fig.4 were present. With a fixed PID controller with parameters: K_P=0.486, K_I=0.227, K_D=0.122, T_S=1.0, [4] the control performance becomes oscillatory after k=100 for the changes of the reference signal as shown on the Fig.4.
On the other hand, when we have applied the proposed AIMNCI algorithm to the same plant model as it is described above, the obtained control performance are shown on the Fig.4 and Fig.5. Corresponding parameters of the used FCRBFN have been as follow: $n = 2$, $n_{cy} = 2$, $\max y = 2.5$, $\min y = 0$, $e_y = 0.5$, $sr_y = 1$, $m = 2, n_{cu} = 2$, $\max u = 2.5$, $\min u = 0$, $e_u = 0.5$, $sr_u = 1$, and the total number of weight parameters was equal to 9, because we applied here the method "one in the data vector" [10]. Also parameters of the SGD learning algorithm with the momentum term were fixed at: $\eta = 0.08, \alpha = 0.3$.
The values of the other parameters of the AIMNCI algorithm were: $\alpha_c = 0.5$ and K_1 =5. The set point filter S was as follows

$$S(z^{-1}) = \frac{0.7473z^{-2}}{1-0.271z^{-1}+0.0183z^{-2}}. \tag{15}$$

As it can be seen from the Fig.4 and Fig.5. we obtained the satisfactory performance of the proposed neuro adaptive control design and very similar results as with the Data-Driven PID Controller (DD-PID) [4], but with simpler and less demanding computation in the AIMNCI algorithm.

Remark 2 The DD-PID design was proposed as one approach to reduce the large computational cost at using NNs and GAs in the control applications of the nonlinear processes. Nevertheless, the DD-PID requries some clustering and removing the redudant data. Additionally, it is well known that in some switching control schemes from one to the other linear controllers instability can arise. In this regard, the DD-PID controller has a disadvantage compared to the AIMCNI that continually adapts its parameters.

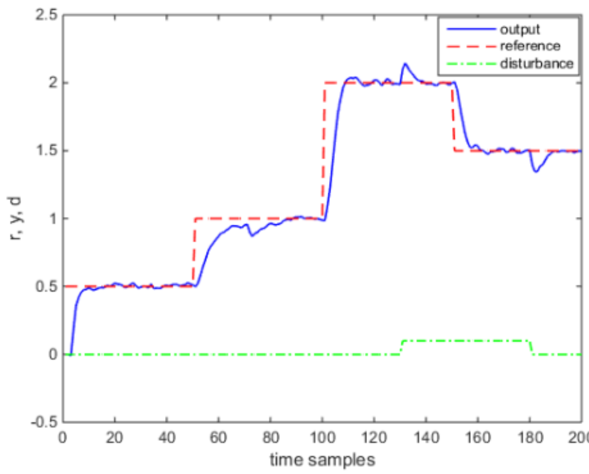

Figure 4. The reference and output in the adaptive system with the AIMNCI algorithm (Ex.1)

Example 2: In this example another type of nonlinear plant model was tested [4], [11]. In this model the following equation applies

$$y(k) = \frac{y(k)y(k-1)[y(k)+2.5]}{1+y^2(k)+y^2(k-1)} + u(k) \tag{16}$$

In this model, there is a kind of hysteresis appearing between y=0 and y=2.5. Thus, the reference signal was chosen in order to illustrate behavior of the adaptive system with the AIMNCI algorithm in the case of this type of nonlinearity.
The reference model,set point filter, for overall dynamics of the control system with the structure shown on Fig.3 was given as in the Example 1 (15).
In this example, parameters of the used FCRBFN have been as follow: $n = 2$, $n_{cy} = 2$, $\max y = 2.5$, $\min y = -1.0$, $e_y = 0.5$, $sr_y = 1$, $m = 1, n_{cu} = 2$, $\max u = 0.3$, $\min u = -1.5$, $e_u = 0.5$, $and\ sr_u = 1$. The learning rate and momentum term parameter of the SGD learning algorithm were fixed at: $\eta = 0.05, \alpha = 0.3$, respectively.
Also, the other parameters of the AIMNCI algorithm used in this example were: $\alpha_c = 0.3$ and K_1=1.
The plant output and control signal for the given changes of the reference are shown on the Fig.6 and Fig.7, respectively. It is seen from these figures that we obtained the satisfactory

results in spite of the strong nonlinearity within controlled plant. Control performance is comparable with that obtained using the DD-PID Controller.

From Figs.6, 7 we can notice the particular changes in the control signal and output during a transient of the reference from 2.5 to -1. This transient corresponds to the upper part of the hysteresis nonlinearity within plant dynamics (16). At the same time, the internal model of the plant implemented by the FCRBFN changes in such way to reduce the output prediction error to zero. This is illustrated by the Fig.8, where the changes of the FCRBFN weight parameters are shown.

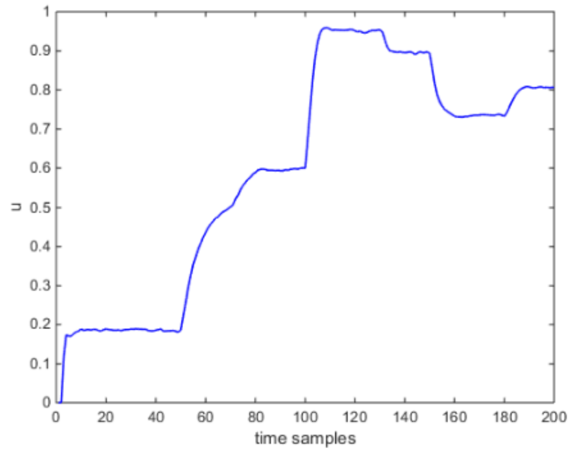

Figure 5. Control signal in the adaptive system with the AIMNCI algorithm (Ex. 1)

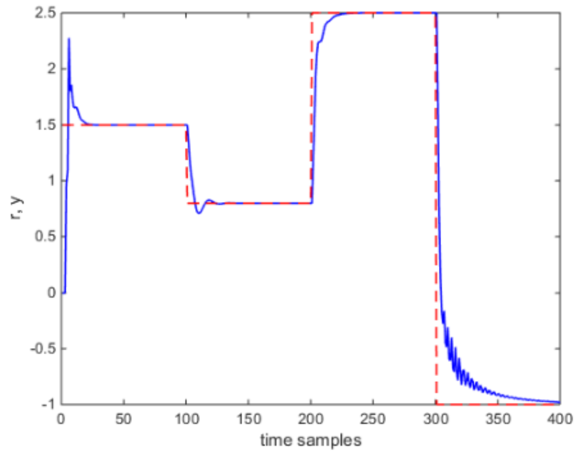

Figure 6. The reference and output in the adaptive system with the AIMNCI algorithm (Ex. 2)

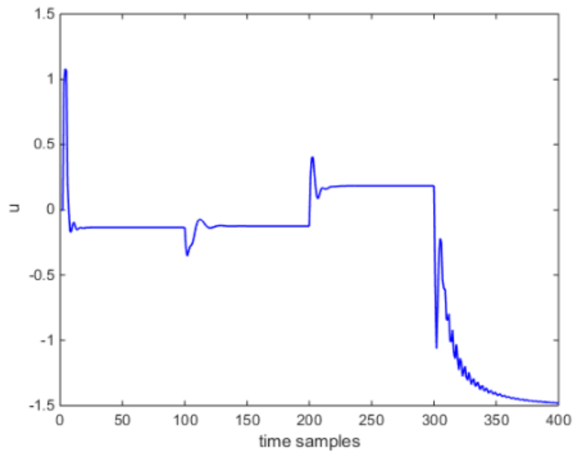

Figure 7. Control signal in the adaptive system with the AIMNCI algorithm (Ex. 2)

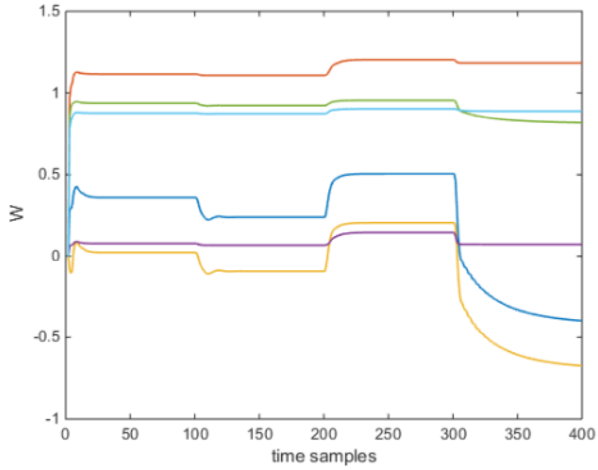

Figure 8. Weight parameters W in the adaptive system with the AIMNCI algorithm (Ex. 2)

IV. CONCLUSION

Starting from the classical control structure with an integral controller through the equivalent block diagram manipulation we came up to the IMC structure with embedded integral action. Based on this control structure and the on-line plant model estimation using the FCRBFN equipped with the SGD learning algorithm, in this paper we proposed the adaptive NN controller design (AIMCNI). Control performance using the proposed AIMCNI controller is illustrated by two examples of the nonlinear controlled plants with variable and strong nonlinear dynamics. In the first example we have demonstrated the robustness of the AIMCNI controller in the case of the strong changes of the nonlinear plant dynamics and step changes of the output disturbance. By the second example we have shown that using the proposed control algorithm the satisfactory results were obtained in spite of the strong nonlinearity within control plant which cannot be controlled by application of classical control algorithms otherwise. Simulation results confirm the validity of the proposed design algorithm in the control applications of the nonlinear and variable industrial processes.

REFERENCES

[1] M. Morari and E. Zafiriou, *Robust Process Control*, Prentice-Hall Int, 1989.

[2] J. M. Sanches and J. Rodellar, *Adaptive Predictive Control: From the concepts to plant optimization*, Prentice Hall, 1996.

[3] I.Rivals, and L.Personnaz, "Nonlinear internal model control using neural networks: Application to processes with delay and design issues," *IEEE Transactions Neural Networks,* vol. 11, no. 1, pp. 80–90, January, 2000.

[4] R. Vilanova, A. Visioli, *PID Control in the Third Millennium: Lessons Learned and New Approaches*, Springer-Verlag London Limited 2012.

[5] M. Mohammadzaheri, L. Chen, and S. Grainger, "A critical review of the most popular types of neuro control*", Asian Journal of Control,* vol. 14, no. 1, pp. 1-11, January 2012.

[6] H.-X. Li, and D. Hua, "An Approximate Internal Model-Based Neural Control for Unknown Nonlinear Discrete Processes," *IEEE Trans. on Neural Networks,* vol. 17, no. 3, pp. 659-670, May, 2006.

[7] D. Kosic, "Fast Clustered Radial Basis Function Network as an adaptive predictive controller", *Neural Networks*, *Elsevier*, no 63, pp. 79-86, 2015.

[8] M. Bozic, J. Igic and I. Krcmar, "Adaptive Internal Model-based Neural Controllers with the SGD learning algorithm", Proceedings of 3rd IcETRAN 2016, Zlatibor, Serbia, pp. AUI2.1.1-6, 2016.

[9] M. Bozic, P. Maric and J. Igic, "A Neuro-Adaptive Control of Nonlinear Slow Processes", Proc. XI International Symposium on Industrial Electronics - INDEL, pp.1-6, 2016.

[10] K.J. Astrom, B. Wittenmark, "Adaptive control", (2.ed.), Addison-Wesley, 1995

[11] K.S. Narendra and K. Parthasarathy, "Identification and control of dynamical systems using neural networks," *IEEE Trans. Neural Networks*, vol. 1, pp. 1–27, 1990.

[12] J.B.D. Cabrera, and K.S. Narendra, "Issues in the application of neural networks for tracking based on inverse control*", IEEE Transactions on Automatic Control,* vol. 44, no. 11, pp. 2007–2027, November 1999.

[13] P.M. Mills, A.Y. Zomaya and M.O. Tade, *Neuro-Adaptive Process Control: A Practical Appoach*, John Willey&Sons, 1996.

[14] K. Patan, "Neural Network-Based Model Predictive Control: Fault Tolerance and Stability", *IEEE Transactions on Control Systems Technology*, vol. 23, no. 3, pp. 1147-1155, 2015.

[15] M.Bozic, J.Igic and I.Krcmar, "An Adaptive Neural IMC Design of Nonlinear Dynamic Processes", *Proc. XV INFOTEH Conference*, Jahorina, vol.15, pp. 752–757, 2016.