A new algorithm for MRAC method using a neural variable learning rate

Ayachi ERRACHDI and Mohamed BENREJEB

***Abstract*—**This paper presents a new algorithm for MRAC (Model Reference Adaptive Control) method based-on neural networks using a variable learning rate. The proposed mechanism adaptation algorithm demonstrates that if the learning rate is large, learning may occur quickly, but it may also become unstable or if the learning rate is small learning adapt reliably, but it may take a long time and thus, it can invalidate the purpose of real-time operation. To overcome these problems we propose a neural controller using variable learning rate. This corresponding algorithm depends on the error between the actual plant output and the output of the reference model. The control strategy is based on two-steps; the first is initialization parameters of the neural controller using reduced number of observation. In the second phase, the parameters of the neural controller are directly tuned from the training data via the tracking error. The simulation results show that the proposed algorithm using variable learning rate is simple to implement and may be extended to multivariable system.

***Keywords*—**Nonlinear system; neural network; variable learning rate; adaptive control; model reference

I. Introduction

THE control of complex dynamic plant is a major concern in control theory [1]. In a consequence, a large number of control structures such as direct inverse control [2], model reference control [3, 4], sliding mode control [5], internal model control [6], feedback linearization [7], backstepping [8], indirect adaptive control [2, 4, 8-13], and direct adaptive control [5, 14-17] have been proposed.

One of these methods may be based on Neural Network (NN). The NNs are used for modeling and control of complex physical systems because of their ability to handle complex input-output mapping without detailed analytical models of the systems [18, 19]. The NN controllers have emerged as a tool for difficult control problems of unknown nonlinear systems [20]. There are several control strategies for neural networks which some of them are: feedforward control, direct inverse control, indirect adaptive control based on NN identification, direct adaptive control with guarantied stability, feedback linearization and predictive control [20-22].

In the industrial process there are many systems having nonlinear properties [20, 23-31]. For instance, the systems to be controlled have constant, unknown or slowly-time uncertain parameters [23-31]. Unless such parameter uncertainty is gradually reduced on-line by an appropriate adaptation or estimation mechanism, it may cause inaccuracy or instability for the control systems [32]. For this reason, an adaptive neural networks controller is applied in this paper.

The Model Reference Adaptive Control is a technique well established in the framework of linear systems [33]. In the direct MRAC approach, the parameters of the linear controller are adapted directly to drive the plant output to follow a desired reference model. This structure can be extended by utilizing the nonlinear function approximation capability of feedforward neural networks such as the Multi-Layer Perceptron (MLP). The MRAC have been adopted by many researchers in controlling nonlinear plants [3, 4, 34-37]. It is not only applied with neural networks while it is applied else approaches. The neural networks are widely used methods for the characterization of nonlinear systems [19].

As long as, the MRAC is well used in some plants which are with unknown parameters, partially known or tainted by noise. In this paper, a new algorithm of the MRAC method is proposed for nonlinear system. The new adaptation mechanism of the proposed method is detailed. The neural network provides the capability to describe highly nonlinear plants. One of the neural parameters is the learning rate $0 \le \eta \le 1$. Indeed, the tuning of the weights depends of this parameter. For instance, if the learning rate is large ($\eta \approx 1$), learning may occur quickly, but it may also become unstable or if the learning rate is small ($\eta \approx 0$) learning adapt reliably, but it may take a long time and thus, it can invalidate the purpose of real-time operation. To overcome these problems we propose a neural controller using variable learning rate. The control strategy used to define the adaptation law is based on the tracking error between the actual plant output and target output, which is the response of the reference model. Then, tuning of the weights is based on the standard delta rule or steepest descent algorithm to minimize the tracking error.

This paper is organized as follows. In the second section, the presentation of the MRAC method is presented. In the third section, the proposed adaptation mechanism is showed. An Example is provided in the forth section, and conclusions are given in the last section.

Example is provided in the forth section, and conclusions are given in the last section.

Ayachi ERRACHDI, LARA Automatique, ENIT BP 37, Le Belvédère, Université de Tunis El Manar, Tunis 1002, Tunisia; (errachdi_ayachi@yahoo.fr).

Mohamed BENREJEB, LARA Automatique, ENIT BP 37, Le Belvédère, Université de Tunis El Manar, Tunis 1002, Tunisia (mohamed.benrejeb@enit.rnu.tn).

II. PRESENTATION OF MRAC METHOD

In figure 1, a multilayer Perceptron is taken in order to concept a nonlinear controller which based on neural network [38]. The adopted general structure of MRAC method is showed here. The nonlinear plant is time-varying system. Figure 1 shows the configuration of the MRAC control system. The used controller is a multilayer Perceptron (MLP) and it contains three layers: the input layer contains N_1 neurons; the hidden layer contains N_2 neurons and one neuron in the output layer. Each neuron of each layer is connected to all neurons of the following layer.

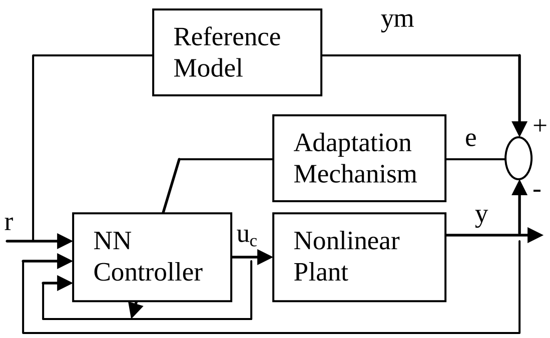

Fig. 1. Model reference adaptive control

A nonlinear system given by the following form:

$$y(k+1)=s\ \ y(k),y(k-1),...,u(k),u(k-1),...,$$

with:

s : unknown function of nonlinear plant

$u_c(k)$: input vector of nonlinear plant

$y(k+1)$: output vector of nonlinear plant

$ym(k+1)$: output vector of reference model

r : input vector of neural networks controller

The output of the l^{th} node, of the hidden layer, is given by the following equation, ($l=1,\dots,N_2$):

$$f\,(h_l)=f\,(\sum_{j=1}^{N_1} w_{lj}x_j) \tag{1}$$

with:

$$h_l=\sum_{j=1}^{N_1} w_{lj}x_j$$

The output of the controller is given by the following equation:

$$\begin{aligned} u_c(k) &= \lambda f\,(\sum_{l=1}^{N_2} f\,(\sum_{j=1}^{N_1} w_{lj}x_j)z_l) \\ &= \lambda f\,(\sum_{l=1}^{N_2} f\,(h_l)z_l) \end{aligned} \tag{2}$$

or in the compact form:

$$u_c(k)=\lambda f\left[z^T F(Wx)\right] \tag{3}$$

with:

$x=\left[x_j\right]^T \in R^{N_1\times 1};\ j=1,...,N_1$

$Z=\ z_l\ ^T \in R^{N_2\times 1};l=1,...,N_2$

$W=\left[w_{lj}\right]\in R^{N_2\times N_1};l=1,...,N_2\ \ and\ j=1,...,N_1$

λ: a scaling coefficient used to expand the range of NN output,

$F(Wx)=\ \ f(h_l)\ ^T \in R^{N_2\times 1};l=1,...,N_2$

The output of the controller $u_c(k)$ is a law control which is used as an input signal for the nonlinear plant. The used training is based on the descent gradient method in order to minimize a function cost (E) and the tuning of the synaptic weights of the neural controller is based on the standard delta rule defined as [19].

$$\begin{aligned} \Delta w_{lj} &= -\frac{\partial E_i(k)}{\partial w_{lj}(k)}=\frac{\partial u_c(k)}{\partial w_{lj}(k)}e(k) \\ &= \lambda f\,'(h_l)\frac{\partial(z_l^T F(Wx))}{\partial w_{lj}(k)}e(k) \end{aligned} \tag{4}$$

$$\begin{aligned} \Delta z_l &= -\frac{\partial E_i(k)}{\partial z_l(k)}=\frac{\partial u_c(k)}{\partial z_l(k)}e(k) \\ &= \lambda f\,'(h_l)\frac{\partial(z_l^T F(Wx))}{\partial z_l(k)}e(k) \end{aligned} \tag{5}$$

with:

$$E=\frac{1}{2}\sum_{k=1}^{N}(e(k))^2=\frac{1}{2}\sum_{k=1}^{N}(y(k)-ym(k))^2$$

In these expressions, η is a positive constant value which represents the learning rate $(0\le\eta\le 1)$ and $F\,'(Wx)$ represents Jacobian matrix of $F(Wx)$.

$$F\,'(Wx)=diag\left[\ f\,'(\sum_{j=1}^{N_1} w_{lj}x_j)\right]$$

with

$$f'(\sum_{j=1}^{N_1} w_{lj}x_j) = \frac{\partial f(\sum_{j=1}^{N_1} w_{lj}x_j)}{\partial(\sum_{j=1}^{N_1} w_{lj}x_j)}; l = 1,..., N_2$$

III. The Proposed Adaptation Mechanism

The aim of the controller is to find the suitable control law which is given by the following equation:

$$u_c(k) = f\ u_c(k-1),...,u_c(k-n_1), r(k), y(k-1), ..., y(k-n_2) \quad (6)$$

Although the changing of the parameters model, the control law must be suitable in order to let the output of plant follow the required trajectory of the model reference, i.e. the convergence of the error between the actual output of plant and the reference model is zero, this condition is given by the following equation.

$$\lim_{k\to\infty} e^{(k+1)} = \lim_{k\to\infty}(ym^{(k+1)} - y^{(k+1)}) \approx 0 \quad (7)$$

At time instant $(k+1)$, is introduced a new data $(u_c^{(k+1)}, y^{(k+1)}, r^{(k+1)})$, if

$$\left\| ym^{(k+1)} - y^{(k+1)} \right\| < \varepsilon \quad (8)$$

If the condition (8) is not satisfied, $\left\| e^{(k+1)} \right\| > \varepsilon$, the tuning of the synaptic weights of the neural controller is necessary in order to reduces the error. The updates of the synaptic weights are given by the equation (9) and (10) [19, 38].

$$w_{lj}^{(k+1)} = w_{lj}^{(k)} + \eta \Delta w_{lj}^{(k+1)} \quad (9)$$

$$z_l^{(k+1)} = z_l^{(k)} + \eta \Delta z_l^{(k+1)} \quad (10)$$

with:

$$\Delta w_{lj} = \lambda f'(h_l) F'(Wx) z_l x^T e(k)$$

$$\Delta z_l = \lambda f'(h_l) F(Wx) e(k)$$

$$\eta = 1/(\lambda^2 f'^2(h_l)\left[F^T(Wx)F(Wx) + z_l^T x^T F'(Wx) F'(Wx) z_l x^T x\right]^T)$$

Algorithm:

1. *Initialize the parameters of the neural model of the controller using M observations,*

2. *Test the condition* $\left\| ym^{(k+1)} - y^{(k+1)} \right\| < \varepsilon$,

3. *if this condition is satisfied, apply the control law* $u_c(k) = f\ u_c(k-1),...,u_c(k-n_1),..,r(k), y(k-1), ..., y(k-n_2)$ *and the increment the time k,*

4. *otherwise return to step 1 and look for the appropriate model using M observations,*

5. *if at the instant $(k+1)$, a package is inserted,*

6. *testing if the error $e_c^{(k+1)}$ converges to zero or not,*

7. *if yes, increment the time,*

8. *otherwise do an update parameters using those obtained at the time k,*

9. *apply whenever the control law,*

10. *end.*

It's clear that the proposed algorithm is simple to implement, but it requires an initialization phase. This step is necessary to find the initialization parameters neural controller like the number of neuron in each layer and the synaptic weights w_{lj} and z_l. This step proceeds in off-line training.

IV. Results and Discussion

In this section, a nonlinear time-varying system is used to study the performance of the proposed MRAC.

A. Example of time-varying system

The time-varying nonlinear system is described by the input-output model in the following equation.

$$y(k+1) = \frac{y(k)y(k-1)y(k-2)u(k-1)(y(k-2)-1) + u(k)}{1 + a_0(k)y^2(k-1) + a_1(k)y^2(k-2)} \quad (11)$$

$$\begin{cases} a_0(k) = 1 + 0.2\cos(k) \\ a_1(k) = 1 - 0.2\sin(k) \end{cases} \quad (12)$$

The trajectory of $a_0(k)$ and $a_1(k)$ are given in the following figure.

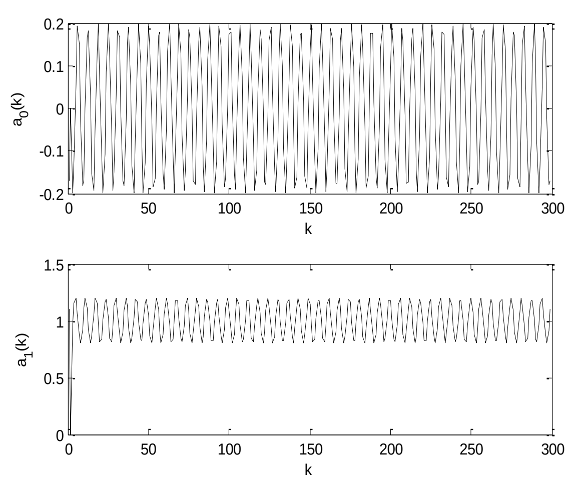

Fig. 2. $a_0(k)$ and $a_1(k)$ trajectories

The multilayer Perceptron network topology with sigmoid activation function was chosen. The variation of error with number of hidden neurons is shown in the following figure.

The lowest error corresponds to 13 neurons in the hidden layer. Hence it is selected as optimal architecture of RNN. The RNN selected here consists of five neurons in the input layer, 13 neurons in the hidden layer and one neuron in the output layer.

The neural model of the nonlinear time-varying time-delay system is presented in figure 4.

The model reference is given by the following equation.

$$y_r(k)=(1+\alpha_1+\alpha_2)y_c(k)-\alpha_1 y_r(k-1)-\alpha_2 y_r(k-2) \quad (13)$$

with y_c is a setpoint sequence, $\alpha_1 = 0.0693$ and $\alpha_2 = 0.0286$.

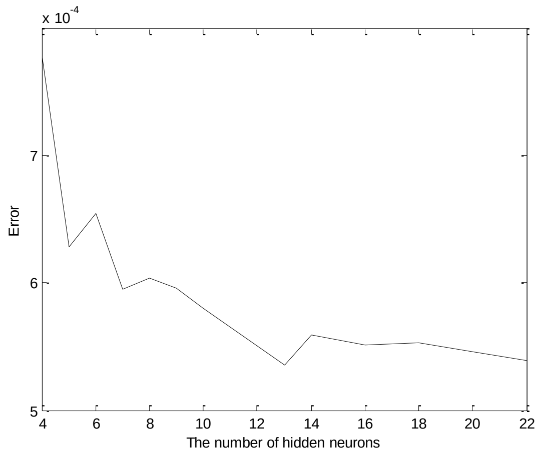

Fig. 3. Variation of error with hidden neurons

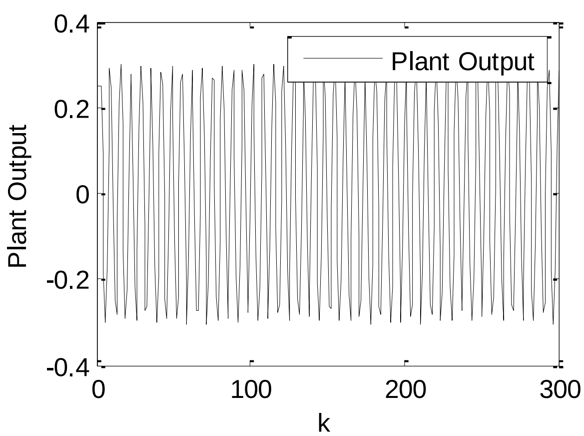

Fig. 4. The time-varying plant output

The output of the reference model and the output of the nonlinear plant are presented in figure 5. The error between the plant output and the model reference output is showed in figure 6. The control law is presented in figure 7.

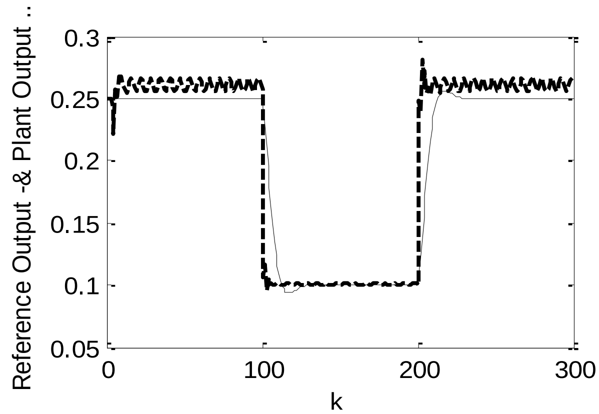

Fig. 5. The plant output and the reference model

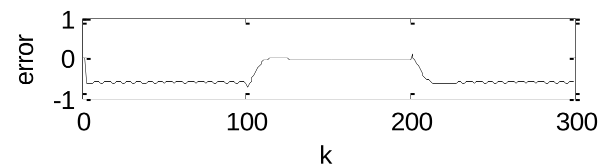

Fig. 6. The error estimation

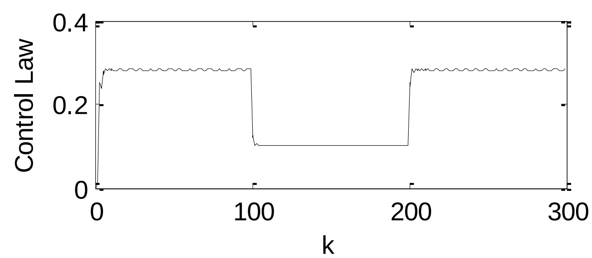

Fig. 7. The control law

B. Effect of disturbances

In this section a noise ξ is added to the output of the plant in order to test the effectiveness of the proposed algorithm.

To measure the correspondence between the system output and the estimated output, a Signal Noise Ratio (SNR) is taken by the following equation:

$$SNR = \frac{\sum_{k=0}^{N} (y(k) - \bar{y})}{\sum_{k=0}^{N} (\xi(k) - \bar{\xi})} \quad (14)$$

with $\xi(k)$ is noise of measurement of symmetric terminal δ, $\xi(k) \in -\delta,\delta$, $\bar{y}$ and $\bar{\xi}$ are an output average value and a noise average value respectively. In this paper, the taken SNR is 5%.

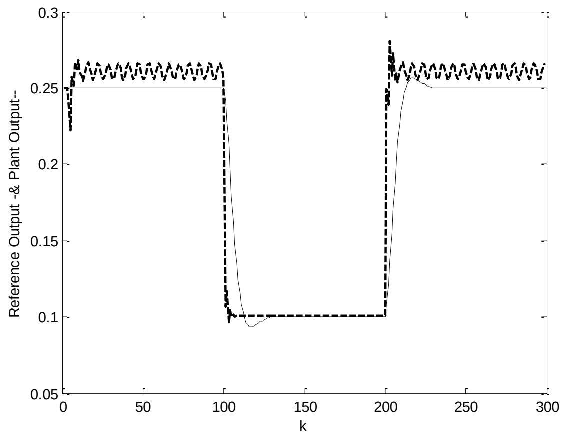

Fig. 8. The plant output and the reference model

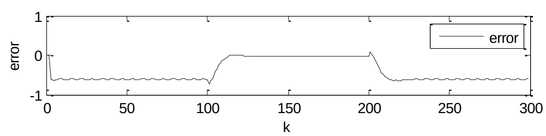

Fig. 9. The error estimation

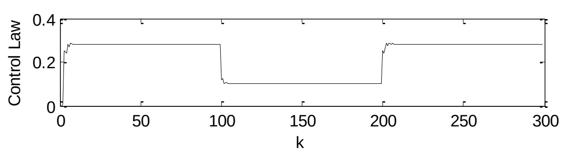

Fig. 10. The control law

The output of the reference model and the output of the nonlinear plant are presented in figure 8. The error between the plant output and the model reference output is showed in figure 9. The control law is presented in figure 10.

In all figures, it is clear that the plant output follows the reference model output although the time-varying parameters and the added noise. This simulation result shows the efficiency of the proposed algorithm, and its simplicity to treat complex nonlinearity.

V. CONCLUSION

This paper has presented a new algorithm for model reference neural network adaptive controller for different cases of nonlinear system with and without noise. The proposed neural controller is based on a variable learning rate. The proposed mechanism adaptation is based on the convergence of the error between the actual output of plant and the output of the model reference. The tuning of the synaptic weight depends on the variation of the parameters of the plant. The simulation results conforms the effectiveness.

REFERENCES

[1] G. Debbache and A. Bennia, “Neural network-based MRAC control of dynamic nonlinear systems”. Int. J. Appl. Math. Compt. Sci., 6(2) : 219-232, 2006.

[2] G.L. Plett, “Adaptive inverse control cf linear and nonlinear systems using dynamic neural control networks”. IEEE Trans. Neural Network, 14(2) : 360-376, 2003.

[3] K.S. Narendra and K. Parthasarthy, “Identification and control of dynamical systems using neural networks”. IEEE Trans. on Neural Networks, 1(1) : 4-27, 1990.

[4] H.D. Patino and D. Liu, “Neural network-based model reference adaptive control system”. IEEE Trans. Syst. Man Cybern., 30(1) : 198-204, 2000.

[5] M. Zhihong, H.R. Wu and M. Palaniswami, “An adaptive tracking controller using neural networks for a class cf nonlinear systems”. IEEE Trans. on Neural Networks, 9(5) : 947-955, 1998.

[6] I. Rivals and L. Personnaz, “Nonlinear internal model control using neural using neural networks: application to processes with delay and design issues”. IEEE Trans. Syst. Man. Cybern., 35(6) : 1311-1316, 2005.

[7] A. Yesildirek and F.L. Lewis, “Feedback linearization using neural networks”. Automatica, 31(11) : 1659-1664, 1995.

[8] Y. Zhang, P.Y. Peng and and Z.P. Jiang, “Stable neural controller using design for unknown nonlinear systems”. IEEE Trans. on Neural Networks, 11(6) : 1347-1359, 2000.

[9] Y-C. Chang and H-M. Yen, “Adaptive output feedback tracking control for a class cf uncertain nonlinear system using neural networks”. IEEE Trans. on Neural Networks, 1: 4-26, 1990.

[10] J.C. Neidhoefer, C.J. Cox and R.E. Saeks, “Development and application of a Lyapunov synthesis based neural adaptive controller”. IEEE Trans. Syst. Man. Cybern., 33(1) : 125-137, 2003.

[11] A.S. Poznyak, W. You, E.E. Sanchez and J.P. Perez, “Nonlinear adaptive trajectory tracking using dynamic neural networks”. IEEE Trans. Syst. Man. Cybern., 35(6) : 1311-1316, 2005.

[12] S. Seshagiri and H. Khalil, “Output feedback control cf nonlinear systems using RBF neural networks”. IEEE Trans. Neural Network. 11(1) : 69-79, 2005.

[13] S-H. Yu and A.M. Annaswamy, “Adaptive control cf nonlinear dynamic systems using adaptive neural networks”. Automatica, 33(11) : 1975-1995. 1997.

[14] S.S. Ge, C.C. Hang and T. Zhang, “A direct method for robust adaptive nonlinear control with guaranteed transient performance”. Syst. Contr. Lett., 37 : 275-284. 1999.

[15] S.N. Huang, K.K. Tan and T.H. Lee, “Nonlinear adaptive control cf interconnected systems using neural networks”. IEEE Trans. Neural Network., 17(1) : 243-246. 2006.

[16] R.M. Sanner and J.E. Slotine, “Gaussian networks for direct adaptive control”. IEEE Trans. Neural Network., 3(6) : 837-863. 1992.

[17] J.T. Spooner and K.M. Passino, “Stable adaptive control using fuzzy systems and neural networks”. IEEE Trans. Neural Net. 4(3) : 339-359. 1996.

[18] P. Borne, M. Benrejeb et J. Haggège, “Les réseaux de neurones. Présentation et applications”. Editions Technip, 2007.

[19] A. Errachdi, I. Saad and M. Benrejeb, “Neural modeling of multivariable nonlinear system. Variable learning rate case”. 18th Mediterranean Conference on Control and Automation, 557-562, 2010.

[20] M. Karadeniz, I. Iskender and S. Yuncu, “Adaptive neural network control of a DC motor”. Gazi University, Faculty of Engineering & Architecture, Department of Electrical & Electronics Engineering, 06570 Maltepe, Ankara Turkiye.

[21] Y.K. Choi, S.K. Lee and Y.C. Kay, “Design and implementation of an adaptive neural network compensator for control systems”. IEEE Trans. Industrial Elect. 48(2). 2001.

[22] N. Magnus, “Neural networks for modeling and control of dynamic systems”. Springer, Berlin. New Yor 2000.

[23] R.D. Braatz and M. Morari, “On the stability of systems with mixed time-varying parameters”. Int. J. of Robust and Nonlinear Control, 7 : 105-112. 1997.

[24] J-X. Xu, Y-J. Pan and T-H. Lee, "VSS identification schemes for time-varying parameters". The 15th Triennial World Congress, IFAC 2002.

[25] W-S. Chen, R-H. Li and J. Li, "Observer-based adaptive iterative learning control for nonlinear systems with time-varying delays". Int. J. Automation and Computing. 2009.

[26] R-H. Chi, S-L. Sui and Z-S. Hou, "A new discrete-time adaptive ILC for nonlinear systems with time-varying parametric uncertainties". ACTA Automatica Sinica, 34(7), 2008.

[27] M.M. Arefi and M.R. Jahed-Motlagh, "Adaptive robust synchronization of rossler systems in the presence of unknown matched time-varying parameters". Commmun Nonlinear Sci Numer Simulat 15: 4149-4157. 2010.

[28] S.S. Ge and J. Wang, "Robust adaptive tracking for time-varying uncertain nonlinear systems with unknown control coefficients". IEEE Trans. Automatic control, 48(8) : 1463-1469. 2003.

[29] Y. Tan, "Time-varying time-delay estimation for nonlinear systems using neural networks". Int. J. Appl. Math. Comput. Sci., 14(1) : 63-68. 2004.

[30] J. Wang and Z. Qu, "Robust adaptive control of a class of nonlinearly parameterized time-varying uncertain systems". IET Control Theory and Applications. 2008.

[31] Y. Zhang, B. Fidan and P.A. Ioannou, "Backstepping control of linear time-varying systems with known and unknown parameters". IEEE Trans. Automatic Control, 48(11). 2003.

[32] Y-W. Cho, K-S. Seo and H-J. Lee, "A direct adaptive fuzzy control of nonlinear systems with application to robot manipulator tracking control". I. Journal of Control, Automation and Systems, 5(6) : 630-642. 2007.

[33] Y.D. Landau, "Adaptive Control: the model reference approach". New York : Marcel Dekker. 1979.

[34] H. Husain, M. Khalid and R. Yusof, "Direct model reference adaptive controller based on neural fuzzy techniques for nonlinear dynamical systems". American Journal of Applied Sciences 5(7) : 769-776, 2008.

[35] J.M. Skowronski, "Adaptive nonlinear model following and avoidance under uncertainty". IEEE Trans. Circuits and system, 35 : 1082-1088, 1988.

[36] J. Lu and T. Yahagi, "Applications of neural networks to nonlinear adaptive control systems". Proceeding of ICSP200 : 252-257, 2000.

[37] H.K. Lam and F.HF. Leung, "On design of a switching controller for nonlinear systems with unknown parameters based on a model reference approach" . IEEE Int. Conference on Fuzzy systems, 1263-1266. 2001.

[38] B. Minaoui, "Contrôle du convertisseur monophasé par réseau de neurones". Afrique Science, 03(1) : 24-36, 2007.

Ayachi ERRACHDI received his engineer diploma in Electrical Engineering and his Master in Automatic and Industrial Maintenance from National School of Engineers of Monastir (ENIM) - Tunisia in 2005 and 2007 respectively. He received the Ph.D. degree in Automatic from National School of Engineers of Tunis (ENIT) – Tunisia in 2012. His research interests include Neural Networks, Dynamical Systems, and Adaptive Control.

Mohamed BENREJEB obtained the Diploma of "Ingnieur IDN" (French "Grande Ecole") in 1973, The Master degree of Automatic Control in 1974, the PhD in Automatic Control of the University of Lille in 1976 and the DSc of the same University in 1980. Full Professor at National School of Engineers of Tunis (ENIT) since 1985 and at "Ecole Centrale de Lille" since 2003, his research interests are in the area of analysis and synthesis of complex systems based on classical and non conventional approaches.