A new algorithm of neural internal model controller using variable learning rate

Ayachi ERRACHDI and Mohamed BENREJEB

***Abstract*—**In this paper is proposed a new algorithm of internal model controller using a proposed variable learning rate. This method overcomes the difficulty to choose a fixed parameter $(0 \le \eta \le 1)$ when we apply an Adaptive method for Internal Model based on neural network Control (AIMC) for nonlinear systems. This work clarify that 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. The proposed method is dependent on the availability of the inverse neural model of the system and the availability of the internal neural model in each moment. These two neural models use the proposed variable learning. The bloc of on-line inverse model and on-line internal model will be used as a bloc of neural controller. This bloc of controller tries to minimize the error between the system output and the internal model output. The adjustment of the controller bloc runs in each moment. The robustness of the proposed adaptive internal model neural network control strategy is investigated in threes cases; firstly when the system has time-invariant parameters, secondly when it has a time-varying parameters and finally when it's a noisy time-varying system. The proposed strategy is compared with the Adaptive Direct Inverse Control (ADIC). From the experiments, it is showing that the performance of the AIMC method is much better than the ADIC method. Two different reference command signals are used to test the control system performance, and it is noted that an excellent tracking response is exhibited in the presence of disturbance.

Keywords — **Nonlinear system, neural networks, variable learning rate, adaptive control, internal model, inverse control.**

I. Introduction

IT is well known nonlinearities, as time-varying parameters, may cause instability and poor performance of practical systems, which have driven many researches to study the problem of nonlinear time-varying systems during the recent years [1].

Until these researches the Neural Network (NN) is well used in modeling and control of nonlinear systems [2] and nonlinear time-varying systems [1]. NN has the advantages to learn sufficiently accurate models [3-8] and provide good nonlinear control when model equations are not known or only partial state information is available [9]. In process control applications, NN can be incorporated in the control strategy via either the direct or indirect methods [10]. In the direct method that is proposed in this paper, a NN is trained to represent the on-line controller bloc of the nonlinear time-varying system. In all application neural modeling and neural control the learning rate is used a constant $(0 \le \eta \le 1)$. However, it is so hard to find a suitable parameter to find the best model or the acceptable controller. Indeed, 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 the difficulty to choose a fixed parameter we propose a new algorithm of internal model controller using a proposed variable learning rate.

In [9, 11-25], a non adaptive Internal Model Control and a non adaptive Direct Inverse Control are applied for nonlinear time-varying systems and they are based on fixed neural controller and they presented them limits. In [26], an online Direct Inverse Control is applied for nonlinear time-varying systems but this method is used in opened loop. In [25], an adaptive Internal Model Control is applied for nonlinear systems and they are based on a neural network with fixed learning rate.

This paper is organized as follows. The problem formulation is briefly discussed in second section. In third section, the procedure for obtaining an on-line inverse model is developed. In the forth section, the Adaptive Internal Model Control method is presented. Examples are provided in section five, and conclusions are given in the latest section.

II. Problem Formulation

It has been shown in [11, 25] that using a fixed neural controller in internal model control can't overcome the problem of changing suddenly of the system parameters. Indeed, the proposed strategy of system control is being in three steps. The first is applied to model the nonlinear systems; the second step is called to find a neural controller finally in the third step the obtained fixed neural controller is used in off-line whereas this strategy is insufficient in the nonlinear time-varying systems. Also, it has been verifying in [26] that

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).

using an adaptive direct inverse control in opened loop is still insufficient to control the nonlinear time-varying systems. In [11] the internal model control is used applying a neural network with fixed learning rate. These motivate us to apply an adaptive internal model control based on neural network using adaptive learning rate to control nonlinear time-varying systems.

III. ON-LINE NEURAL INVERSE MODEL

On the basis of the input and output relation of a system, the above nonlinear system can be expressed by the following form:

$$y(k+1) = f\,(y(k),...,y(k-N_y+1),u(k),...,u(k-N_u+1)) \qquad (1)$$

where f is a nonlinear mapping, $u(k)$ and $y(k)$ are the input and output vector, N_u and N_y are the maximum input and output lags, respectively.

The goal of the neural network is to track the above system so that at each time the model mapping is as a close as possible to the system input-output mapping. The model used is described by the following equation:

$$ym(k+1) = \lambda s(\sum_{l=1}^{N_1} s(\sum_{j=1}^{N_0} w_{lj} x_j) z_l) \qquad (2)$$

The training algorithm is a gradient descent algorithm, a version of the back propagation algorithm, driven by the above error function. As it will be shown in the simulation examples, the gradient descent algorithm is very efficient in minimizing the cost (error) criterion.
The objective of this algorithm is to minimize the sum of the squares of the errors J_m.

$$J_m = \frac{1}{2}\sum_{k=0}^{N}(e_m(k))^2 = \frac{1}{2}\sum_{k=0}^{N}(y(k) - ym(k))^2 \qquad (3)$$

The derivate of (J_m) with respect z_l and w_{lj} gives the equation for updating the synaptic weights of the internal model:

$$\begin{aligned} z_l(k) &= z_l(k-1) + \eta_1 \lambda s'(h_l) S(Wx) e_m(k) \\ w_{lj}(k) &= w_{lj}(k-1) + \eta_1 \lambda s'(h_l) S'(Wx) z_l x^T e_m(k) \end{aligned} \qquad (4)$$

with (η_1) is a variable learning rate given by the following equation :

$$\eta_1(k) = \frac{1}{\lambda^2 s'^2(\sum_{j=1}^{N_0} w_{lj} x_j)\left[S^T(Wx)S(Wx) + z_l^T S'(Wx) S(Wx) z_l x^T x\right]} \qquad (5)$$

The on-line identification algorithm is given in the [27]. However, in the other hand, to find an inverse model for a complex nonlinear system using conventional methods, the structure of neural inverse model is showing in figure 1.

The network is fed with past inputs, past outputs and present outputs. The network then predicts the controller outputs. The network then predicts the controller output $u(k)$.

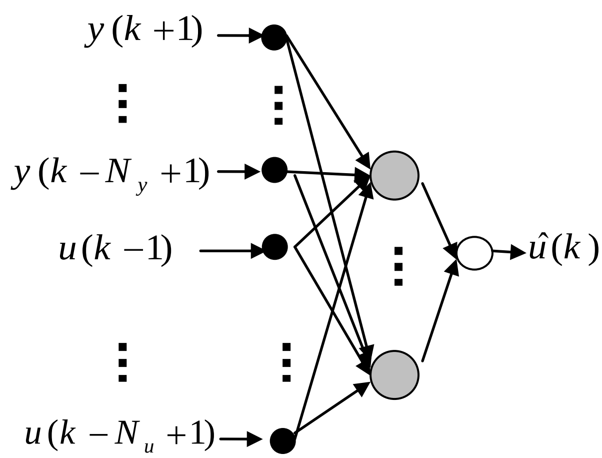

Fig. 1. The Inverse Neural Network Model

The final network representation of the inverse model is given by:

$$u(k) = f^{-1}(y(k+1),...,y(k-N_y),u(k-1),...,u(k-N_u)) \qquad (6)$$

The Recurrent Neural Network (RNN) used in the nonlinear system identification and control is a multilayer neural network with tapped delay lines of both input and output. The delay elements are used to introduce delayed inputs and outputs that are then fed to a static network as the regressor vector so that the predicted RNN out will follow the target output [27].

The output of the inverse neural network model $\hat{u}(k)$ is given by the following equation.

$$\hat{u}(k) = \lambda s(\sum_{l=1}^{N_1} s(\sum_{j=1}^{N_0} w_{lj}^{(1)} x_j) z_l^{(1)}) \qquad (7)$$

The same training algorithm is a gradient descent algorithm, and its objective is to minimize the sum of the squares of the error J_{mi}.

$$J_{mi} = \frac{1}{2}\sum_{k=0}^{N}(e_{mi}(k))^2 = \frac{1}{2}\sum_{k=0}^{N}(u(k) - \hat{u}(k))^2 \qquad (8)$$

The derivate of (J_{mi}) with respect $z_l^{(1)}$ and $w_{lj}^{(1)}$ gives the equation for updating the synaptic weights of the internal model:

$$\begin{aligned} z_l^{(1)}(k) &= z_l^{(1)}(k-1) + \eta_2 \lambda s'(h_l) S(Wx) e_{mi}(k) \\ w_{lj}^{(1)}(k) &= w_{lj}^{(1)}(k-1) + \eta_2 \lambda s'(h_l) S'(Wx) z_l^{(1)} x^T e_{mi}(k) \end{aligned} \qquad (9)$$

The need for using a variable learning rate is to have a fast training, we have:

ISSN: 2313-0563

$$e_{mi}(k+1) - e_{mi}(k) = u(k+1) - \hat{u}(k+1) - u(k) + \hat{u}(k) \quad (10)$$

we suppose that

$$\Delta u(k+1) = u(k+1) - u(k) \quad (11)$$

And

$$\Delta \hat{u}(k+1) = \hat{u}(k+1) - \hat{u}(k) \quad (12)$$

By application of [26]

$$\|\Delta u(k+1)\| \ll \|\Delta \hat{u}(k+1)\| \quad (13)$$

The

$$\begin{aligned} e_{mi}(k+1) - e_{mi}(k) &\approx -\Delta \hat{u}(k+1) \\ &= \eta_2 \frac{\partial \hat{u}(k)}{\partial w_{lj}^{(1)}(k)} e_{mi}(k) \\ &= -\lambda \Delta s(h_l) \end{aligned} \quad (14)$$

$$\begin{aligned} & e_{mi}(k+1) - e_{mi}(k) \approx -\lambda^2 \eta_2 s'^2(h_l) * \\ & \left[S^T(Wx)S(Wx) + z_l^T S'(Wx) S'(Wx) z_l^{(1)}{}_l x^T x \right] e_{mi}(k) \end{aligned} \quad (15)$$

So we find that

$$e_{mi}(k+1) - e_{mi}(k) = -\eta_2 \xi_2(k) e_{mi}(k) \quad (16)$$

With

$$\begin{aligned} & \xi_2(k) \approx \lambda^2 s'^2(h_l) * \\ & \left[S^T(Wx)S(Wx) + z_l^{(1)T} S'(Wx) S'(Wx) z_l^{(1)} x^T x \right] e_{mi}(k) \end{aligned} \quad (17)$$

from where

$$e_{mi}(k+1) - e_{mi}(k) = \ 1 - \eta_2 \xi_2(k) \ e_{mi}(k) \quad (18)$$

To ensure convergence, i.e., $\lim_{k \to \infty} e_{mi}(k) = 0$ it is necessary that $\|e_{mi}(k)\| < 1$ is satisfied.

This condition proves that $0 < \eta_2 < 2\xi_2^{-1}(k)$. For that and in order to have a variable learning rate is necessary that:

$$\eta_2 = \xi_2^{-1}(k) \quad (19)$$

$$\eta_2(k) = \frac{1}{\lambda^2 s'^2 (\sum_{j=1}^{N_0} w_{lj}^{(1)} x_j) \left[S^T(Wx)S(Wx) + z_l^{(1)T} S'(Wx) S(Wx) z_l^{(1)} x^T x \right]} \quad (20)$$

The proposed on-line training algorithm is implemented by the following steps [26].

Algorithm:

1. *Using M observations $(M \ll N)$, in off-line algorithm to find a reduced inverse model given the equation (7).*
2. *At time instant $(k+1)$, is introduced a new data $(u^{(k+1)}, y^{(k+1)})$, if $e_{mi}^{(k+1)} < \varepsilon$ (given threshold) is satisfied then the model approaches sufficiently the inverse of the dynamic nonlinear system and increment the time.*
3. *If the absolute value of the error $(e_{mi}^{(k+1)})$ for this data pair is greater than the threshold then an update of the synaptic weights $z_l^{(1)}$ and $w_{lj}^{(1)}$ is necessary which are given by the equation (9).*
4. *Generate a new data pair and test the Step. 2 and Step. 3.*
5. *End.*

IV. Adaptive internal model control for nonlinear system

The Direct Inverse Control (DIC) is well applied in off-line control [6-24]. In different studies it is demonstrated that this strategy of control have a problems [11]. For instance, it is ever impossible to obtain a suitable process model due to the complexity of the underlying process and the lack of knowledge of its critical. For instance, an Adaptive Direct Inverse Control (ADIC), given by the figure 2, is proposed and demonstrated in [26].

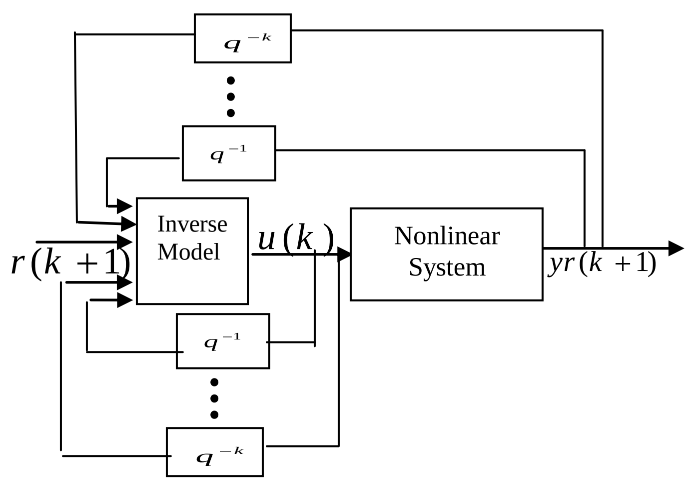

Fig. 2. The **A**daptive **D**irect **I**nverse **C**ontrol (**ADIC**)

A promising way to overcome these problems the adaptive internal model control is proposed which presented in figure 3.

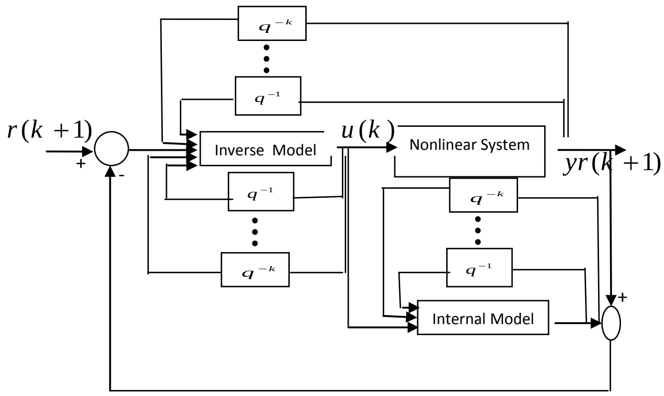

Fig. 3. Adaptive Internal Model Control (AIMC)

This type of control system is composed in two blocs: the inverse model and the internal model. The well principle of this kind of system control is to find an on-line inverse model and an on-line internal model in each second. $r(k+1)$ is the reference signal and $\hat{u}(k)$ is the control signal.

In the whole design process, no constraints and prior knowledge of the controlled plant are required, and the stability of the convergence of the control system can guaranteed.

The above training algorithm is implemented by the following steps.

Algorithm:

1. *Using M observations $(M \ll N)$, in off-line algorithm to find a reduced inverse model (equation (7)) and the reduced internal model (equation (2)).*

2. *at time instant $(k+1)$, is introduced a new data $(u^{(k+1)}, y^{(k+1)})$, if $e_{mi}^{(k+1)} < \varepsilon$ (given threshold) is satisfied then the on-line inverse model approaches sufficiently the inverse of the dynamic nonlinear system.*

3. *Calculate the control law if not update the synaptic weights.*

4. *If the on-line internal model approaches sufficiently the dynamic system then increment the time $(k+1)$.*

5. *Test the error $(e(k+1) = y(k+1) - ym(k+1))$, if this gap is under a given threshold $(e(k+1) < \varepsilon)$, increment the time $(k+1)$.*

6. *If the absolute value of the error $e_{mi}^{(k+1)}$ for this data pair is greater than the threshold then an update of the synaptic weights $z_l^{(1)}$ and $w_{lj}^{(1)}$ is necessary which are given by the equation (9) an (10).*

7. *Test the Step 3 and Step 4.*

8. *Generate a new data pair and test the Step. 2 and Step. 3.*

9. *End.*

V. Experimental Result

In this section, an example of nonlinear system with three cases is taken. Indeed, in the first case, the system parameters are constants. Secondly, the time-varying parameters are taken and finally a noise $v(k)$ is added in the time-varying system output.

The applied example is a nonlinear system proposed by Narendra and Parthasarathy [28], is given by the following equation.

$$y(k+1) = \frac{y(k)y(k-1)y(k-2)u(k-1)(y(k-2)-1)+u(k)}{a_0(k)+a_1(k)y^2(k-1)+a_2(k)y^2(k-2)} \quad (21)$$

with

$$u(k) = 0.325\sin(0.95k)\cos(0.55k) \quad (22)$$

Square (1) and sinusoidal (2) reference command signals are used to test the control system performance and are represented in figure 5 and 6.

A. Nonlinear system with time-invariant parameters

In this case, the nonlinear system defined by the equation (11) in which their parameters are given by the following equation:

$$\begin{cases} a_0(k) = 1 \\ a_1(k) = 1 \\ a_2(k) = 1 \end{cases} \quad (23)$$

The figure 4 presents the on-line inverse neural model for the nonlinear system. In the first phase of learning, a set of input-output training data ($M = 4$) are used us to find the initialization parameters of model. The obtained parameters are used to adjust on-line the parameters of the whole model.

In all figures (Fig. 5 to Fig. 10), solid line is the reference signal (1) or (2), dashed line is the system output using the ADIC method however the pointed line is the system output using the AIMC method.

In figure 5 (respectively figure 6), the reference signal (1) (respectively the reference signal (2)) is presented with the output system using the ADIC method and the AIMC method. An excellent correspondence between these signals is noticed.

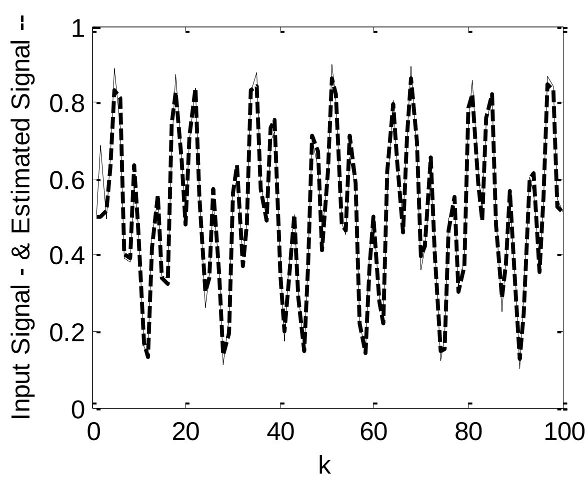

Fig. 4. The On-line Inverse Neural Model

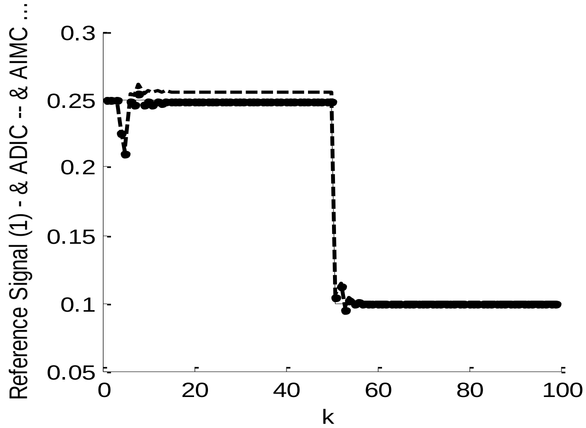

Fig. 5. The Output System using the ADIC, the AIMC and the Reference Signal (1)

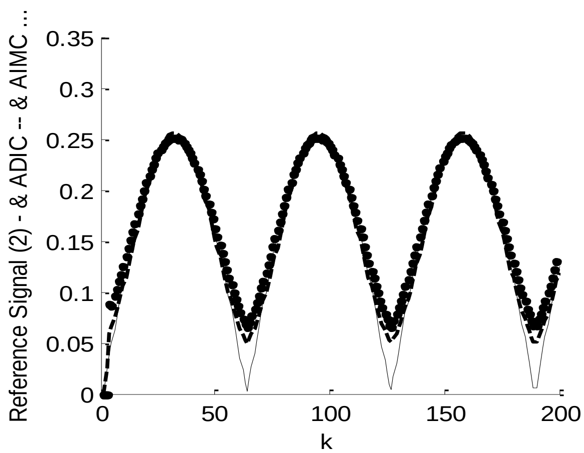

Fig. 6. The Output System using the ADIC, the AIMC and the Reference Signal (2)

B. Nonlinear system with time-varying parameters

In this case, the nonlinear system defined by the equation (11) in which their parameters are given by the following equation:

$$\begin{cases} a_0(k) = 1 \\ a_1(k) = 1 - \cos(95k) \\ a_2(k) = 1 - \sin(95k) \end{cases} \quad (24)$$

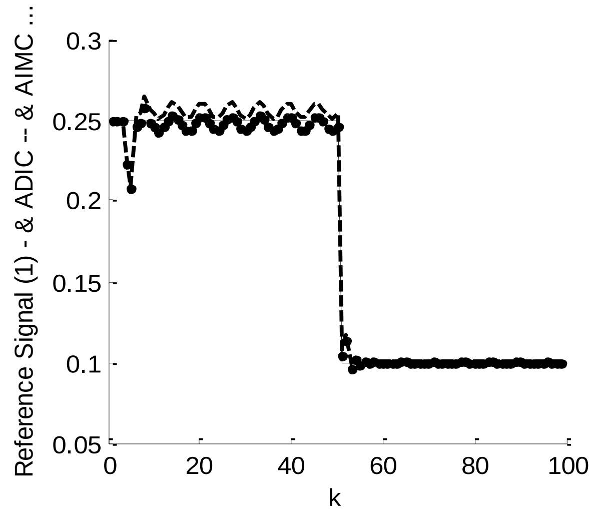

Fig. 7- The Output System using the ADIC, the AIMC and the Reference Signal (1)

Although the variation of the system parameters an excellent correspondence between the output system using the AIMC method which presented in figures 7 and 8.

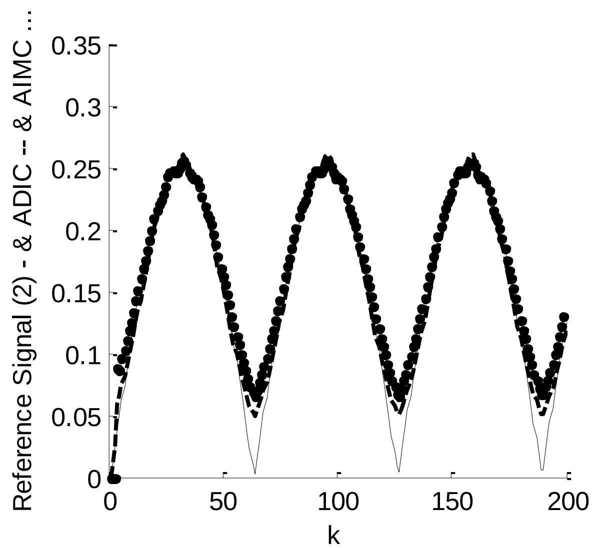

Fig. 8- The Output System using the ADIC, the AIMC and the Reference Signal (2)

C. Noisy nonlinear system with time-varying parameters

In this case, the time-varying system is used with an added noise in the output of the system.

$$y(k+1) = \frac{y(k)y(k-1)y(k-2)u(k-1)(y(k-2)-1)+u(k)}{a_0(k)+a_1(k)y^2(k-1)+a_2(k)y^2(k-2)} + v(k) \quad (25)$$

The influence of this disturbance was measured by the Signal Noise to Ratio (SNR) which is given by the following equation:

$$SNR = \frac{\sum_{k=1}^{N} (y(k) - \bar{y})^2}{\sum_{k=1}^{N} (v(k) - \bar{v})^2} \quad (26)$$

where $\bar{y}$ and $\bar{v}$ are respectively the output average value and noise average value.

In figure 9 and 10, the noisy system output $SNR = 10\%$ and the two references signals ((1) and (2)) are presented using the ADIC method and the AIMC method.

In these two figures, the plant output follows the reference signals ((1) and (2)) although the disturbances which injected in the output of the system. In these two examples, the results simulation proves that the suggested algorithm is able to control all nonlinear systems.

Furthermore, the parameters of the controller like learning rate can be adjusted on-line with the other parameters. This avoids the divergence of the developed algorithm and allows a robustness, flexibility and stability of the AIMC method in each time.

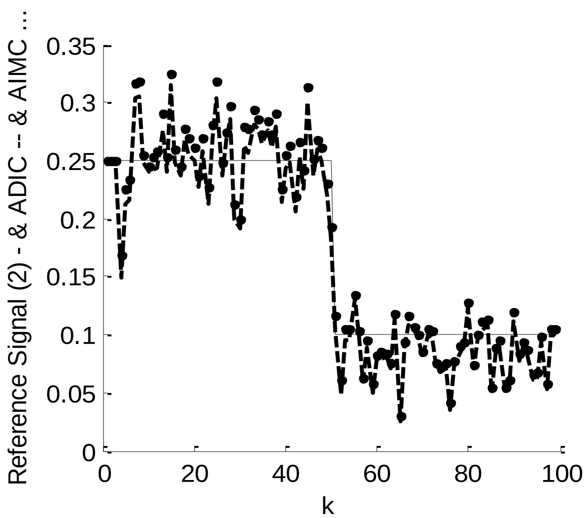

Fig. 9- The Noisy Output System using the ADIC, the AIMC and the Reference Signal (1)

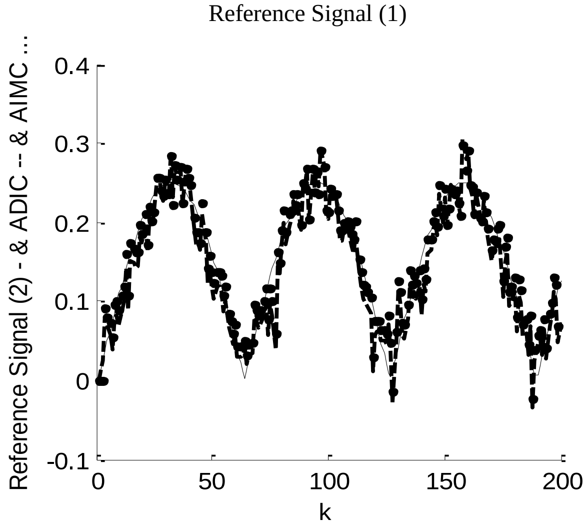

Fig. 10- The Noisy Output System using the ADIC, the AIMC and the Reference Signal (2)

VI. CONCLUSION

In this paper, a new algorithm of internal neural model controller, using variable learning rate, is proposed.

The proposed strategy for adaptive internal model control of nonlinear time-varying systems guaranties that closed loop system is stable. The advantage of this method with respect to direct inverse control is that in direct inverse control we use just a neural controller with the nonlinear time-varying system without closed loop but in the proposed strategy a closed loop which used including an error between the system output and the internal model output.

In comparison of obtained results here for this method with obtained results in [11, 25], we can see that not only the online calculation of the control law but also its robustness against uncertainty in system parameters, because the adjustment of the control law is being in each second. Another advantage of the above method is its simplicity in implementation.

Two different reference command signals (square and sinusoidal) are used to test the control system performance, and it is noted that an excellent tracking response is exhibited in the presence of disturbance. Future work is directed to extend this control design to multivariable nonlinear time-varying system.

REFERENCES

[1] Q. Fang, C. Boatong and J. Yan, “Further results on robust stability of neural system with mixed time-varying delays and nonlinear perturbations”. Nonlinear Analysis: Real World Applications, pp. 895-906, 2010.

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

[3] J. Richalet, A. Rault, J.L. Testud and J. Papon, “Model predictive heuristic control: applications to industrial processes”, Automation 14, Issue 5, pp. 413-428, 1978.

[4] Gowrishankar and P.S. Satyanarayana, “Recurrent Neural Network Based Bit Error Rate Prediction for 802.11 Wireless Local Area Network”, IJCSES, Vol. 2, No. 3, pp. 171-177, 2008,

[5] M. A. M. Abdeen and H. Hodhod, “Experimental Investigation and Development of Artificial Neural Network Model for the Properties of Locally Produced Light Weight Aggregate Concrete”. Engineering, 2, 408-419, 2010.

[6] M-C. Hwang, "Adaptive Neural Network Impedance Control for Robotic Manipulators Driven by Current-Fed Induction Motors ", IJCSES, Vol. 3, No. 2, pp. 143-149, 2009.

[7] G. S. Uttreshwar, and A. A. Ghatol, "Hepatitis B Disease Diagnosis Using Artificial Neural Network", IJCSES, Vol. 3, No. 4, pp. 325-333, 2009.

[8] H. Hossam and A.M. Mostafa Abdeen, “Experimental comparative and numerical predictive studies on strength evaluation of cement types: effect of specimen shape and type of sand”. Engineering, 2, 559-572, 2010.

[9] D.C. Psichogios and L.H. Ungar, “Direct and indirect model-based control using artificial neural networks”. Industrial and Engineering Chemistry Research, 30, 2564, 1991.

[10] W. Daosud, P. Thitiyasook, A. Arpornwichanop, P. Kittisupakorn and M.A. Hussain, “Neural network inverse model-based controller for the

control of a steel pickling process". Computers & Chemical Engineering, pp. 2110-2119, 2005.

[11] A. Errachdi, I. Saad and M. Benrejeb, "Internal model control for nonlinear time-varying system using neural networks". 11th International conference on Sciences and Techniques of Automatic Control computer engineering, Monastir, 19-21, 2010.

[12] M.A. Hussain, "Review of the applications of neural networks in chemical process control-simulation and on-line implementation". Artificial Intelligence in Engineering, 13, 55, 1999.

[13] M.A. Hussain and L.S. Kershabaem, "Implementation of inverse-model-based control strategy using neural networks on a partially simulated exothermic reactor". Transactions of the Institution of Chemical Engineers (Part A), 20, 1039, 2000.

[14] M.A. Hussain, "Adaptive inverse model control of a continuous fermentation process using neural networks". In I. M. Mujtaba, & M. A. Hussain (Eds.), Application of neural networks and other learning technologies in process engineering (pp. 199-216). London: Imperial College Press, 2001.

[15] N. Tang, L. Chunxiu, and S. Zuoliang, "Artificial Neural Networks Model on the Heating Performance of the Seawater Desalination Brine by Solar Energy", IJCSES, Vol. 4, No. 2, pp. 89-93, 2010.

[16] D.B. Anuradha, G.P. Reddy and J.S.N. Murthy, "Direct inverse neural network control of q continuous stirred tank reactor (CSTR)". Proceedings of the International MultiConference of Engineers and Computer Scientists, Vol. 3, 2009.

[17] P.Q Xia, "An inverse model of MR damper using optimal neural network and system identification". Journal of Sound and Vibration 266, pp. 1009-1023, 2003.

[18] A. Rahideh, H.M. Shaheed and A.H. Bajodah, "Adaptive non-linear model inversion control of a twin rotor multi-input multi-output system using artificial intelligence". J. Aerospace Engineering, JAERO 155, pp. 343-351, 2007.

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

[20] B. Widrow, "Adaptive inverse control". IFAC Adaptive Systems in Control and Signal Processing, Lund, Sweden, 1986.

[21] M. Salem, D.E. Chaouch et M.F. Khelfi, "Commande neuronale inverse des systèmes non linéaires". 4th International Conference Computer Integrated Manufacturing CIP'2007.

[22] G. Puscasu and V. Palade, "An improved inverse neural control structure for nonlinear dynamic systems". 0-7803-5682-9/99/$10.00@ IEEE 1999.

[23] J. Lobato, P. Canizares, M.A. Rodrigo, C.G Piuleac, S. Curteanu and J.J Linares, "Direct and inverse networks modelling applied to study the influence of the gas diffusion layer properties on PBI-based PEM fuel cells". Int Journal of Hydrogen Energy, pp. 1-9, 2010.

[24] M. Bayat, M. Sedighizadeh and A. Rezazadeh, "Wind energy conversion systems control using inverse neural model algorithm". International Journal of Engineering and Applied Science (IJEAS), Vol 2, Issue 3, pp. 40-46, 2010.

[25] I. Rivals and L. Personnaz, "Internal model control using neural networks". Proceedings of the IEEE International Symposium on Industrial Electrons. 1, pp. 109-114, 1996.

[26] A. Errachdi, I. Saad and M. Benrejeb, "On-line direct inverse neural network control for nonlinear system". 13th International conference on Sciences and Techniques of Automatic Control computer engineering, Monastir, 17-19, 2012.

[27] A. Errachdi, I. Saad and M. Benrejeb, "On-line Identification Method Based on Dynamic Neural Network". International Review of Automatic Control, Vol 3, N°5, pp. 474-479, 2010.

[28] K.S. Narendra and K. Parthasarathy, "Identification and control of dynamical systems using neural networks". *IEEE* Transactions on Neural Network*s*, Vol. 1 (Issue 1):4-27, 1990.

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.
from the biography.