Journal of Engineering, Sciences, Assiut University, Vol. 35, No. 6, pp. 1481-1493, November 2007

DIFFERENT IDENTIFICATION METHODS WITH APPLICATION TO A DC MOTOR

Mohamed M. Hassan, Amer A. Aly, and Asmaa F. Rashwan
Electrical Engineering Department, Faculty of Engineering, Assiut University, Assiut, Egypt

(Received July 16, 2007 Accepted October 30, 2007)

Neural networks and fuzzy inference systems are becoming well-recognized tools of designing an identifier / controller capable of perceiving the operating environment and human operator with high performance. The purpose of this paper is to identify different models for a DC-servo system using the previous identification methods and linear identification methods as ARX, ARMAX and state space models. The paper compares between these methods and presents the practical results for the application these methods.

***KEYWORDS**: identification, neural network, neuro-fuzzy, DC motor.*

1. INTRODUCTION

In recent years aspects of system identification have been discussed in a multitude of papers, at many conferences and in an appreciable number of university courses. Apparently the interest in this subject has different roots, (e.g. definite needs by engineers in process industries to obtain a better knowledge about their plants for improved control) [1].

1.1 System Identification

System identification is the task of inferring a mathematical description, a model, of a dynamic system from a series of measurements on the system. There are several motives for establishing mathematical descriptions of dynamic systems. Typical applications encompass simulation, prediction, fault diagnostics, and control system design. Figure 1 illustrates the task of system identification. A model shall represent the behavior of a process as closely as possible. The model quality is typically measured in terms of a function of the error between the process output and the model output. This error is utilized to adjust the parameters of the model.

There are three different modeling approaches can be distinguished white box models, gray box models and black box models are solely based on measurement data and no or very little prior knowledge is exploited [2]. This paper deals with black box modeling.

1.2 System Identification Procedures

When attempting to identify a model of a dynamic system it is common practice to follow a procedure similar to the one depicted in Fig. 2.

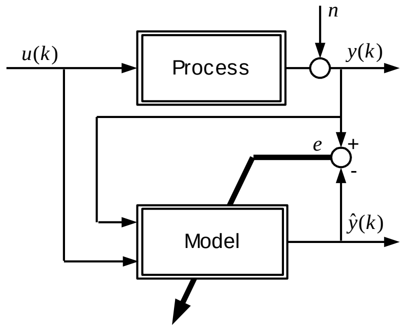


Fig. 1 System identification

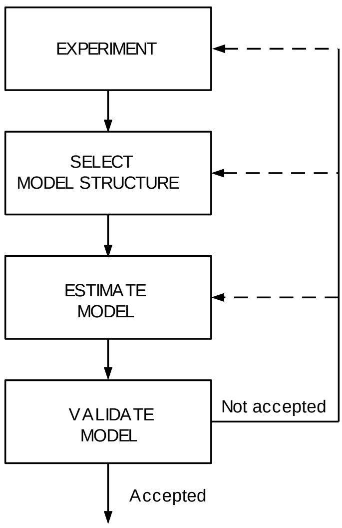


Fig. 2 The basic system identification procedure

The purpose of the experiment is to collect a set of data that describes how the system acts over its entire range of operation. The second step is to select a “family” of model structures considered appropriate for describing the system. The following step will pick the model that performs best according to some type of criterion. In the last step the model it must be evaluated to investigate whether or not it meets the necessary requirements. The paper is organized as follows. Section 2 explains the approaches of the linear system identification. Section 3 gives a brief introduction to an ANN, ANFIS. Section 4 illustrates the identification of the dc-motor and evaluates the performance of each structure. Section 5 presents conclusions.

2. LINEAR DYNAMIC SYSTEM IDENTIFICATION

In this paper two parametric time-domain model representations will be used : the Auto-Regressive with eXogenous input (ARX) model and the Auto-Regressive

Moving Average with eXogenous input (ARMAX) model. These models are special cases of the following general model [3]:

$$A(q)y(k) = \frac{B(q)}{F(q)}u(k) + \frac{C(q)}{D(q)}v(k) \tag{1}$$

Where A(q), B(q), C(q), D(q) and F(q) are polynomials in the shift operator q, defined as

$$A(q) = 1 + a_1 q^{-1} + a_2 q^{-2} + \ldots + a_{na} q^{-na}$$

$$B(q) = \quad + b_1 q^{-1} + b_2 q^{-2} + \ldots + b_{nb} q^{-nb}$$

$$C(q) = 1 + c_1 q^{-1} + c_2 q^{-2} + \ldots + c_{nc} q^{-nc}$$

$$D(q) = 1 + d_1 q^{-1} + d_2 q^{-2} + \ldots + d_{nd} q^{-nd}$$

$$F(q) = 1 + f_1 q^{-1} + f_2 q^{-2} + \ldots + f_{nf} q^{-nf}$$

By making certain choices for these polynomials, the different model representations are obtained.

2.1 Auto-Regressive With Exogenous Input Model

The ARX model can be obtained from the general model (1) by choosing $C(q) = D(q) = F(q) = 1$, and $A(q)$ and $B(q)$ arbitrary polynomials [4]

$$A(q)y(k) = B(q)u(k) + v(k) \tag{2}$$

Since the noise enters directly in the equation, the model is of the class of equation error models.

2.2 Auto-Regressive Moving Average with Exogenous Input Model

The ARMAX model can be derived from the general model (1) by choosing $D(q) = F(q) = 1$ and $A(q), B(q)$ and $C(q)$ arbitrary polynomials [4]

$$A(q)y(k) = B(q)u(k) + C(q)v(k) \tag{3}$$

2.3 State-Space Model

The state-space model takes the following form:

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) + Ke(k) \\ y(k) &= Cx(k) + Du(k) + e(k) \end{aligned} \tag{4}$$

The easiest and most straightforward way to obtain a state space model from data is to estimate an input/output model.

The state-space model can be obtained by choosing A, B, C, D and K arbitrary matrices.

3. NONLINEAR DYNAMIC SYSTEM IDENTIFICATION

3.1 Multilayer Perceptron (MLP) Network

A feed forward Multilayer perceptron [4,5] is layered network made up of one or more hidden layers between the input and output layers. Each layer consists of several percepton neurons are used in parallel and connected to the neurons in adjacent layers. The input layer acts as an input data holder that distributes the inputs to the first hidden layer. The output from the first layer nodes then becomes input to the second layer, and so on. The last layer acts as the output layer; see Fig.3. In basis function formulation the MLP can be written as

$$\hat{y} = \sum_{i=0}^{M} w_i \phi_i \left(\sum_{j=0}^{P} w_{ij} u_j \right) \qquad \text{with } \phi_\circ(\cdot) = 1, \tag{5}$$

where M as the number of neurons in the hidden layer, w_i as output layer weights, and w_{ij} as weights of the hidden layer [6].

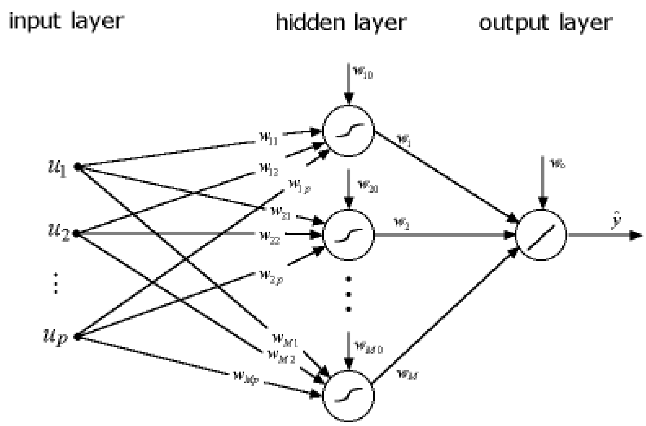


Fig. 3 A multilayer perceptron network

3.2 Neuro-Fuzzy Modeling

ANFIS Architecture

For simplicity, we assume the fuzzy inference system under consideration has two inputs x and y and one output z . Suppose that the rule base contains two fuzzy if-then rules of Takagi and Sugeno's type [7,8]:

Rule 1: If x is A_1 and y is B_1, then $f_1 = p_1 x + q_1 y + r_1$

Rule 2: If x is A_2 and y is B_2, then $f_2 = p_2 x + q_2 y + r_2$

then the type-3 fuzzy reasoning is illustrated in Fig.4(a), and the corresponding equivalent ANFIS architecture (type-3 ANFIS) is shown in Fig.4(b) [9,10,11]. The ANFIS has 5 layers and functions of these layers are explained below:

Layer 1: In this layer where the fuzzification process takes place, every node is adaptive. Outputs of this layer form the membership values of the premise part.

Layer 2: In contrary to layer 1 the nodes in this layer are fixed. Each node output represents a firing strength of a rule.

Layer 3: In this layer where the normalization process is performed, the nodes are fixed as they are in Layer 2. The ratio of the *i*th rule's firing strength to the sum of all rule's firing strength is calculated for the corresponding node.

Layer 4: Since the nodes in this layer operate as a function block whose variables are the input values, they are adaptive. Consequently the output of this layer forms TSK outputs and this layer is referred to as the consequent part.

Layer 5: This is the summation layer. Which consist of a single fixed node. It sums up all the incoming signals and produces the output.

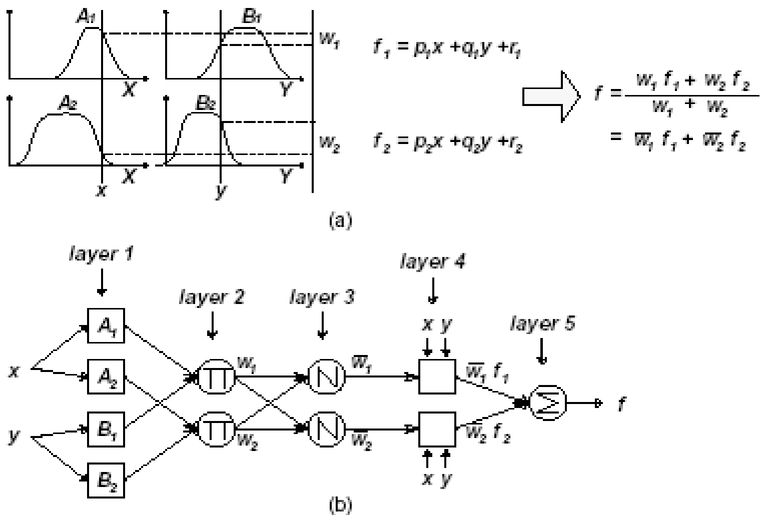


Fig. 4 (a) Type-3 fuzzy reasoning; (b) equivalent ANFIS (type-3 ANFIS)

3.3 Type of Models

In analogy to linear system identification [3], a nonlinear dynamic model can be used in two configurations: for prediction and for simulation. Prediction means that on the basis of previous process inputs $u(k-1)$ and process outputs $y(k-1)$ the model predicts one or several steps into the future. A requirement for prediction is that the process output is measured during operation. In contrast, simulation means that on the basis of previous process inputs $u(k-1)$ only the model simulates future outputs. Thus, simulation does not require process output measurements during operation. Fig.5 compares the model configuration for prediction (a) and simulation (b). In former linear system identification literature and in the context of neural networks, fuzzy systems and other modern nonlinear models the one-step prediction configuration is called a series-parallel model while the simulation configuration is called a parallel model.

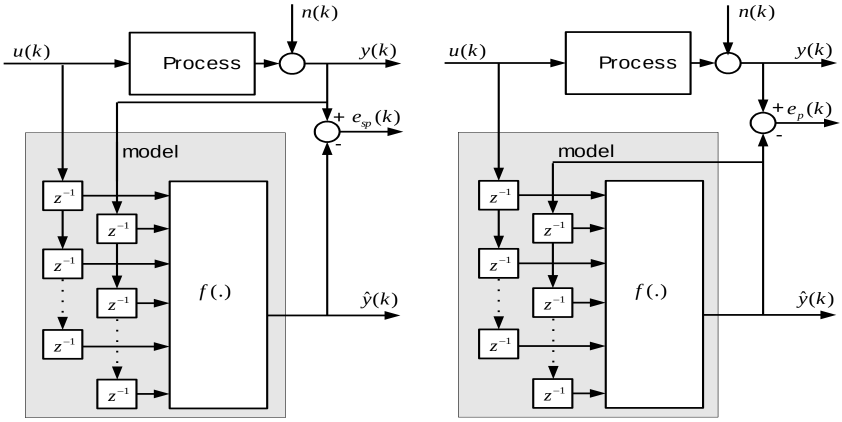


Fig. 5 (a) One-step prediction with a series parallel model

(b) simulation with a parallel model

For a second order model, the one-step prediction is calculated with the previous process outputs as

$$\hat{y}(k) = f(u(k-1), u(k-2), y(k-1).y(k-2)) \quad (6)$$

while the simulation is evaluated with the previous model outputs as

$$\hat{y}(k) = f(u(k-1), u(k-2), \hat{y}(k-1).\hat{y}(k-2)) \quad (7)$$

4. DC SERVO MOTOR IDENTIFICATION

4.1 System Model

The DC motor dynamics are given by the following equations [12]:

$$\begin{aligned} K\omega(t) &= -R_a i_a(t) - L_a[di_a(t)/dt] + V_c(t) \\ Ki_a(t) &= J[d\omega(t)/dt] + D\omega(t) + T_L(t) \end{aligned} \quad (8)$$

where $\omega, V_c, i_a, R_a, L_a, D, J, K$ and T_L are the rotor speed, terminal voltage, armature current, armature resistance, armature inductance, damping constant, rotor inertia, torque, back emf constant and load torque, respectively. From the previous equations, the overall transfer function of the motor system can therefore be written as:

$$\frac{\omega(s)}{V_c(s)} = \frac{k_t}{s^2 + a_1 s + a_2} \quad (9)$$

where $k_t = k_m \times k_a$ and k_a is the servo amplifier gain, constant a_1, a_2 and k_m depend on the motor parameters, can be written in the form of difference equation as:

$$V_c(k) = \alpha\omega(k) + \beta\omega(k-1) + \zeta\omega(k-2) \quad (10)$$

where α, β and ζ are constants and their values depend on the motor parameters and the sampling time.

4.2 Identification of DC Motor Using Linear Method

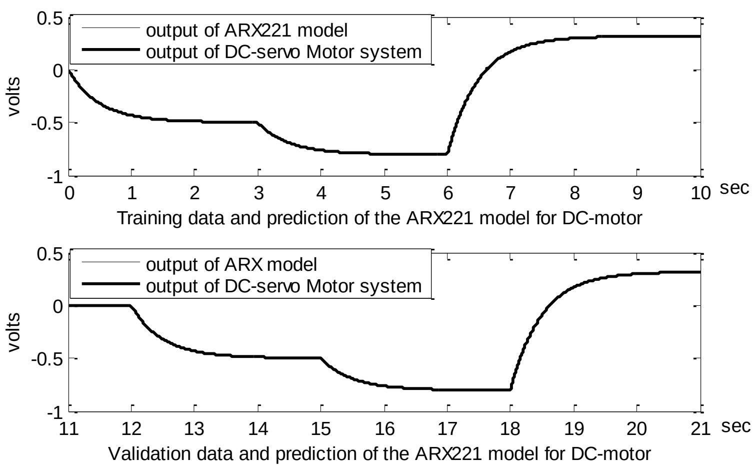
Equation (10) which is approximated via previous linear approaches using some input/output data for estimate the model and another data for validate this model. The input to linear model is $V_c(k)$ and its output is $\omega(k)$.the first 2001 pairs (training data set) were used for training the models, while the remaining 2000 pairs (checking data set) were used for validating the model identified. Table 1 demonstrates The values of the parameters of the linear models and Fig.6 illustrates the results of identification.

Experimental Results

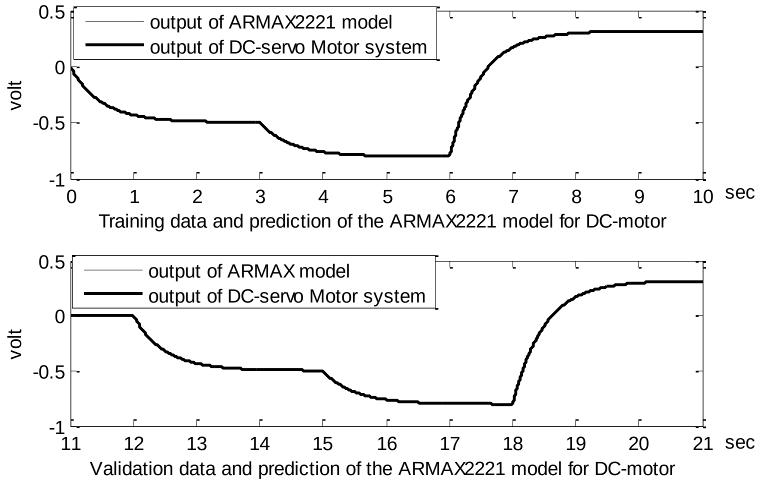
The experiments were carried out by supplying a square wave input in Fig.7 and a sawtooth wave input in Fig.8. Fig.9 and Fig.10 show the motor response, the ARX model response to a square wave and a sawtooth wave inputs, respectively and the error between the motor output and the ARX model output.

Table 1 The values of the parameters of the linear models.

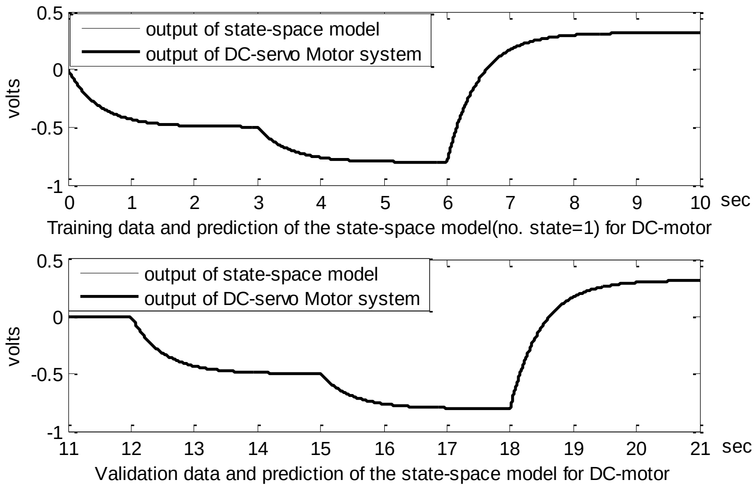
The models	The coefficients					
	a1	a2	b1	b2	c1	c2
ARX	-0.49	-0.495	0.008856	0.004428		
ARMAX	-0.49	-0.495	0.008856	0.004428	0.3414	0.3536
	The matrices					
	A	B	C	D		K
State-Space	0.99	0.0010782	8.2135	0		0.10196



(a) DC-servo motor system and ARX model outputs.



(b) DC-servo motor system and ARMAX model outputs.



(c) DC-servo motor system and state space model outputs.

Fig. 6

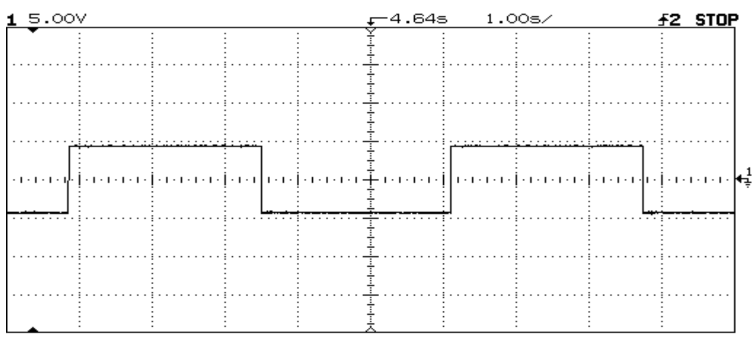


Fig. 7 A square wave input.

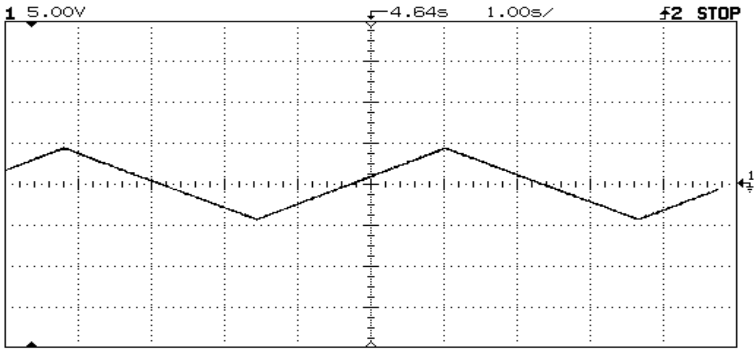


Fig. 8 A sawtooth wave input.

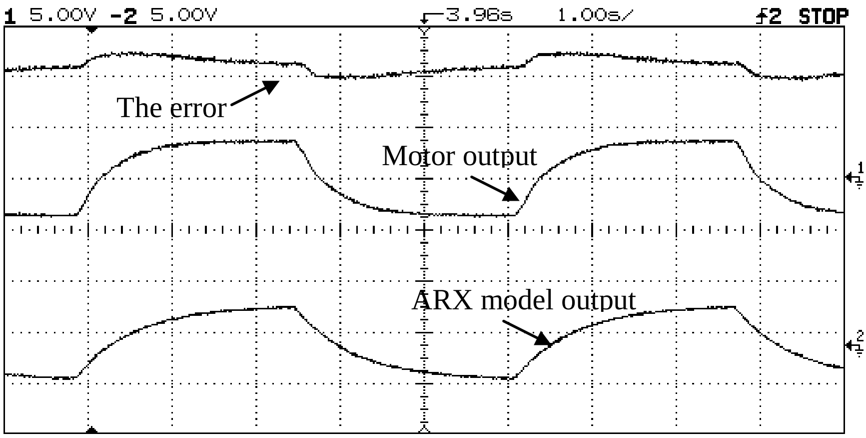


Fig. 9 The experimental motor and the ARX model outputs to a square wave input.

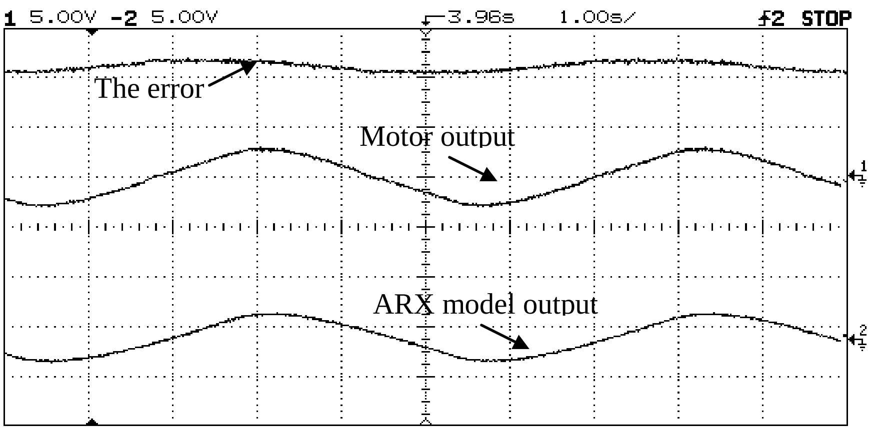


Fig. 10 The experimental motor and the ARX model outputs to a sawtooth wave input.

4.3 Identification Of DC Motor Using Nonlinear Method

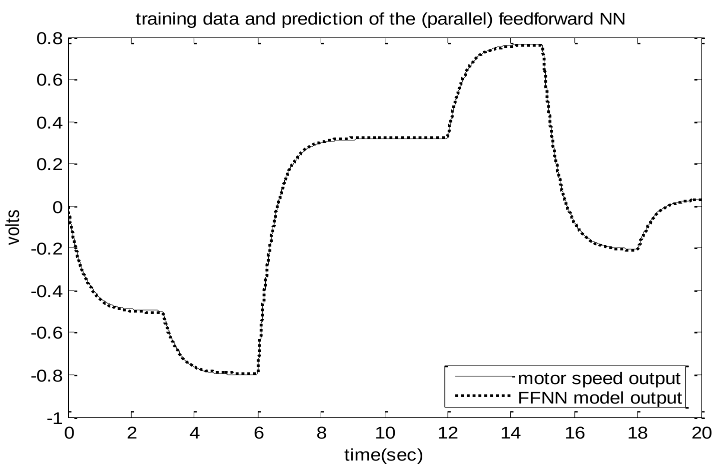
Equation (10) can be written in the form

$$\omega(k) = f\left(V_c(k), \omega(k-1), \omega(k-2)\right) \qquad (11)$$

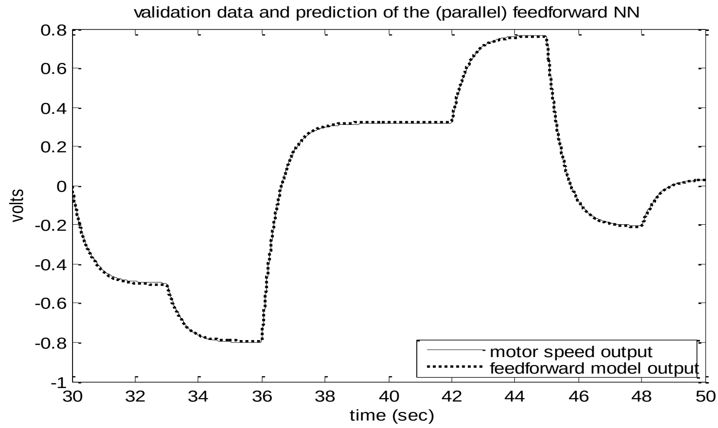
in the identification model, a feedforward neural network belonging to the class $\pi^2_{3,2,1}$ is used to approximate the function $f(\cdot)$. The inputs to a neural network are $V_c(k), \omega(k-1)$ *and* $\omega(k-2)$ and its output is $\omega(k)$. A neuro-fuzzy is used to approximate the function $f(\cdot)$ with the number of rules is 6. the first (4001, 2001) pairs (training data set) were used for training neural network and ANFIS respectively, while the remaining (4001, 2000) pairs (checking data set) were used for validating the model identified. During the identification process a series-parallel model is used, but after the identification process is terminated the performance of the model is studied using a parallel model. Fig.11 (a, b) shows the training data and the validation data of a dc-motor system with the prediction output of a neural network model and Fig.12(a, b) shows the training data and the validation data of a dc-motor system with the prediction output of a neuro-fuzzy model [12].

Experimental Results

Figures 13 and Fig.14 show the motor response, the neural network model response to a square wave and a sawtooth wave inputs, respectively and the error between the motor output and the neural network model output. We note that the maximum error in the neural network model is less than the maximum error in the ARX model.

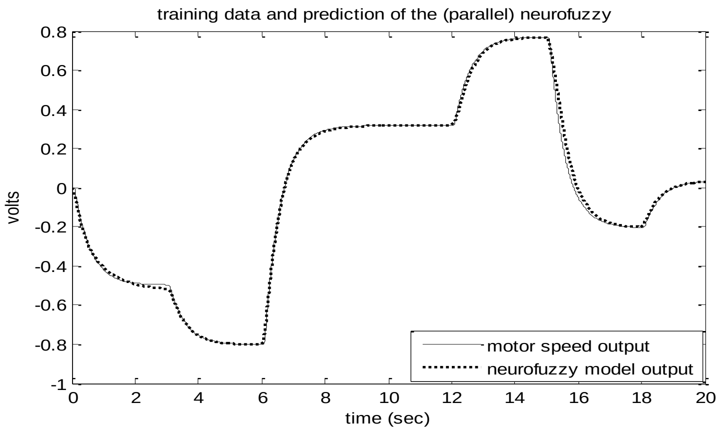


(a)

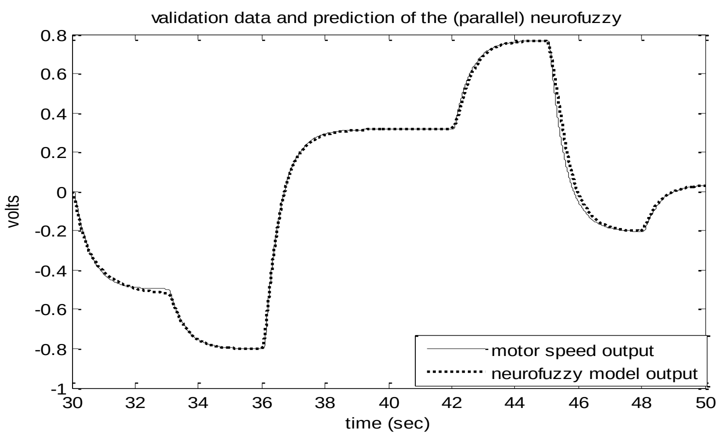


(b)

Fig. 11 DC-servo motor system and feedforward network (FFNN)



(a)



(b)

Fig. 12 DC-servo motor system and ANFIS model

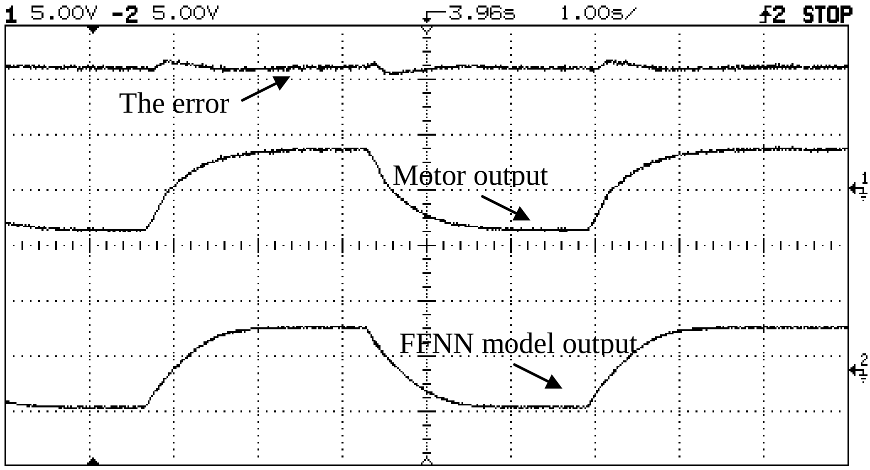


Fig. 13 The experimental motor and the neural network model outputs to a square wave input.

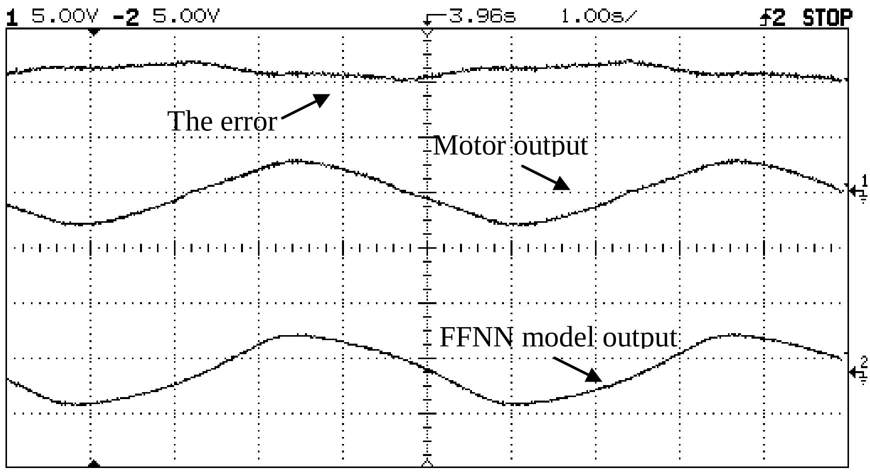


Fig. 14 The experimental motor and the neural network model outputs to a sawtooth wave input.

5. CONCLUSION

Linear and nonlinear identification have been compared, referring to their usability as black box model for DC-servo motor. The identification of DC-motor has been successfully captured by an artificial neural networks (ANNs) and neuro-fuzzy.

6. REFERENCES

[1] K.J. Åström and P.E. Eykhoff, "System Identification-A Survey" , Automatica, Vol.7, PP. 123-162, 1971.

[2] Sjöberg, J.; Zhang, Q.; Ljung, L.; Benveniste, A.; Delyon, Glorennec, P.-Y.; Hjalmarsson, H.; Judisky, A.: "Nonlinear Black-Box Modeling In System

Identification: A Unified Overview”. In: Automatica, Vol. 31, No. 12, 1995, PP. 1691-1724.

[3] P.P.J. Van Den Bosch; A.C. Van Der Klauw: “Modeling, Identification And Simulation Of Dynamical Systems”. 1994 by CRC Pess, Inc.

[4] Oliver Nelles: “Nonlinear System Identification From Classical Approaches To Neural Networks And Fuzzy Models”. Springer-Verlag Berlin Heidelberg 2001.

[5] Ali Gulbag and Fevzullah Temurtas “A Study On Quantitative Classification Of Binary Gas Mixture Using Neural Networks And Adaptive Neuro-Fuzzy Inference Systems” Sensors and Actuators B 115 (2006) 252–262.

[6] Tim Callinan (August 2003) "Artificial Neural Network Identification And Control Of The Inverted Pendulum" http://www.eeng.dcu.ie/~brutonj/Reports/TCallinan_MEng_03.pdf

[7] J.-S. R. Jang and C.-T. Sun, “Neuro-Fuzzy Modeling And Control," The Proceedings of the IEEE, vol. 83, pp. 378-406, Mar. 1995.

[8] J.-S. R. Jang, “ ANFIS: Adaptive-Network-Based Fuzzy Inference Systems," IEEE Trans. on Systems, Man, and Cybernetics, vol. 23, pp. 665-685, May 1993.

[9] R. Babuska; H. Verbruggen "Neuro-Fuzzy Methods For Nonlinear System Identification" Annual Reviews in Control, Volume 27, Number 1, 2003, pp. 73-85(13).

[10] Cheng-Jian Lin, Cheng-Hung Chen "Identification And Prediction Using Recurrent Compensatory Neuro-Fuzzy Systems" Fuzzy Sets and Systems 150 (2005) 307–330.

[11] Andreas Nürnberger, Arne Radetzky, Rudolf Kruse "Using Recurrent Neuro-Fuzzy Techniques For The Identification And Simulation Of Dynamic Systems" Neurocomputing 36 (2001) 123-147.

[12] S. Weerasooriya and M.A. El-Sharkawi, “Identification And Control Of A Dc Motor Using Back-Propagation Neural Networks”, IEEE Transaction on Energy Conversion, pp.663-669, December1991.

الطرق المختلفة للتعريف مع تطبيقها علي محرك تيار مستمر

هذا البحث يعرض مقدمة مختصرة عن أنظمة التعريف الخطية كنظام الارتداد الذاتي بالإدخال الخارجي, نظام الارتداد الذاتي للمتوسط المتحرك بالإدخال الخارجي, نظام المدى و أنظمة التعريف اللاخطية كالشبكة العصبية للتغذية الأمامية, الشبكة العصبية الغيمية. نقوم باعتبار محرك التيار المستمر نظام مجهول يتم تعريف نموذج له باستخدام الطرق السابقة في التعريف ثم نقوم بالمقارنة بينهم.