



Neuro-Predictive Speed Control for Induction Motor Drives Using Dynamic Neural Networks

M. Zerikat †, M. Guedda ††

† LAAS, Département de génie électrique, ENSET – Oran, B.P. 1523 El Menaouer, Oran- Algérie

†† LAMFA, Université de Picardie Jules Vernes, 33, rue Saint Leu 80039, Amiens- France

Résumé :

On se propose dans cet article, de présenter une technique de régulation en vitesse d'un moteur à courant alternatif de type asynchrone à l'aide d'une stratégie de commande basée sur les réseaux de neurones. L'application d'un schéma de commande prédictive neuronale dans le domaine des entraînements électriques, constitue notre contribution pour l'applicabilité de cette technique et l'influence de l'horizon de prédiction sur la poursuite de trajectoires imposées. L'accent est mis sur la robustesse de l'algorithme utilisé vis-à-vis les variations des paramètres de la machine et des influences externes. Le réseau de neurone dans la structure de commande proposée exécute deux fonctions. La première fonction identifie le système nonlinéaire à tout instant t , et la deuxième fonction assure la commande en vitesse du moteur asynchrone considéré. Les résultats obtenus par simulation numérique démontrent la robustesse de cette commande compte tenu de la complexité et de la présence de fortes nonlinéarités de la machine à induction.

Mots clés : Commande prédictive réseau de neurone dynamique, moteur à induction

Abstract :

Neural network controller for robust accurate tracking of an asynchronous AC-Motor operating in a high performance drives environment is proposed. It has shown to be a very successful controller design strategy in both theory and practice. In this paper one scheme of predictive control using neural networks, is developed and tested for induction motor drive. The neural network in the proposed structure performs two functions. The first is to identify the non-linear system dynamic at all times. The second function of the neural networks is to control the motor speed. Using simultaneous on-line training of the neural network and controller synthesis, the resulting algorithm is adaptive. The developed control algorithm is robust to parameter variations and external influences. It also assures accuracy tracking with the prescribed dynamics. The simulation results demonstrates the effectiveness of the designed controller in the control of AC-motor drive which are considered as highly nonlinear complex system

Keywords : Predictive control , dynamic neural network, induction motor

1. Introduction

In recent years, much research activity has centered around neural networks. It has been recognized that neural networks offer a number of potential benefits for application in the field of control engineering, because many industrial processes such as induction motor control, exhibit a certain type of difficulties. This is due to highly non-linear dynamics of the machine, high coupling and unavailability of measurements of the state variables and usually of some of the controlled outputs, in a typical hardware configuration. In addition to that, all these processes are also very often affected by unknown disturbances and strong variations of machine parameters due to changing thermal conditions, that make the implementation of robust control. To solve problems emerging in the control of such processes, advanced control strategies have been researched. Model predictive control design is shown to be very effective in many cases, as demonstrated in [1],[2] and there is a growing interest even in the field of electrical engineering. This control strategy calculates a control signal such that the predictive process output is close to the desired reference trajectory. The advantages of this method are to provide good results in term of rapidity, disturbances rejection and errors cancellations [2]. However, recently neural networks based control approaches have been developed do not require prior knowledge about the plant to be controlled for the compensation of the nonlinearities caused by the influence of disturbances (i.e. load or parameter variations). Neural network have many attributes that make them an interesting new alternative to control AC-motor drives. The objective of the design of network predictive control systems for unknown complex processes, is to develop algorithms for identification and control through a learning process (Chen, 1990; Hernandez and Arkun, 1992; Hunt and Sbarbaro, 1991; Narendra and Parthasarath, 1990; 1991). This paper presents an application of neural network based predictive control of nonlinear, time invariant processes. The proposed technique is applied to design the speed controller of an indirect field-oriented induction motor drive. The electrical drive is modeled utilizing the input-output model based on Multi-Layer Perceptron (MLP) neural networks. Some simulated results are provided to confirm the effectiveness of the proposed controller. The rest of this paper will follow the structure and mathematical description of multilayer neural

network in section 2. The learning control is reviewed in section 3. The algorithm is then used predictive control in section 4 and the system description is described in section 5. This is followed by computer simulations and concluding remarks in section 6 and 7, respectively.

2. Multilayer Neural Network

A general architecture graph of MultiLayer Perceptron (MLP) is shown in figure 1. This network which can be multiplexed for each output of the controller has been found to possess acceptable performance in many industrial application. The feed-forward topology shown in the network of Fig.1. offers the advantage of simplicity and ease programming. Such a neural network contains three layers, input layer, hidden layers and output layer. Each layer is composed of several neurons. The number of the neurons in the output and output layers depends on the number of the selected input and output variables. The number of hidden layers and the number of neurons in each depend on the system dynamic and the desired degree of accuracy.

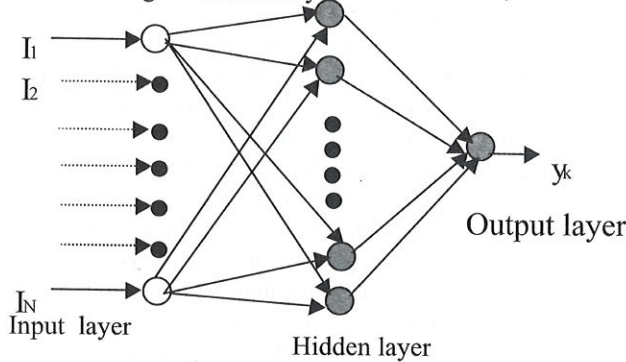


Fig.1. Architecture of Multilayer Neural Network

The block-diagram of Fig. 2. Shows the model of a neuron, which performs two functions. The first is to sum all the inputs from the upper layer based on their weighting factors in equation (1.a). The second is to process this sum by a nonlinear sigmoidal function in equation (1.b)

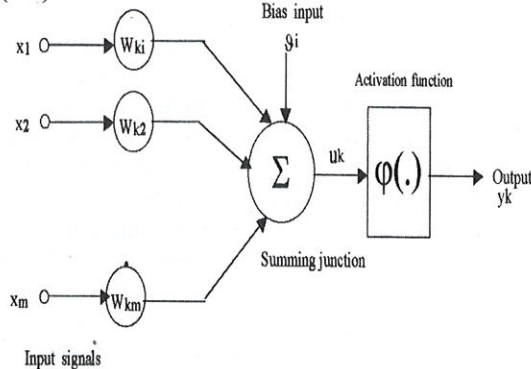


Fig.2. Basic model of neuron

The basic equations describing the dynamics of each neuron are

$$u_k = \sum_{j=1}^m w_{ji} x_j \quad (1.a)$$

$$y_k = \Phi(u_j + \vartheta_j) \quad (1.b)$$

where w_{ji} design the synaptic weight between the j th neuron and the i th neuron in two adjacent layers.

$\Phi(\cdot)$ is the activation function. The neural network has two phases of operations, training and testing. In the training phase, the weights of the neural network are adjusted to map the input of the system to its output. In the testing phase, the neural network should predict the correct system output for a given input, even if the input was not used in training. Here for generality, the scalar-weighted summing of the input array x_i is distorted by a linear function $\phi(\cdot)$ which is usually sigmoidal (e.g. tanh function) to facilitate the gradient search techniques used in the training procedure. An Artificial Neural Networks (ANNs) is made up of many such neurons arranged in a variety of architectures. The feed-forward architecture graph shown in Fig.1. offers the advantage of simplicity and ease of programming.

3. Neural Network Training

The most common method of neural network is error back-propagation algorithm (Rumelhart et al., 1986). The algorithm is based on the gradient descent search technique that minimizes a cost function of the mean square errors. The minimization process is done by adjusting the weighting vector of the neural network. Several training algorithms have been proposed to adjust the weight values in dynamic recurrent neural network. Examples for these methods are the dynamic back-propagation from Narendra and parthasarathy; 1991; Williams and Ziepsner, 1995; among others. The cost function being minimized is the error between the network output and the desired output given by equation (2).

$$E = \frac{1}{2} \sum_j e_j^2(k) = \frac{1}{2} \sum_j [y_j^* - y_j(k)]^2 \quad (2)$$

where $y_j(k)$ is the output of neuron j and $y_j^*(k)$ is the desired pattern for that neuron.

Let $\eta_{ji}(k)$ denote the learning rate parameter assigned to synaptic weight $w_{ji}(k)$ at iteration number k . Minimizing equation (2) leads to a sequence of update of the weight vector. The weights of the interconnections between two adjacent layers can be update based on the following formula [10,11]

$$w_{ji}(k+1) = w_{ji}(k) - \eta_{ji}(k+1) \frac{\partial E(k, w)}{\partial w_{ji}(k)} + \alpha \Delta w_{ji}(k) \quad (3)$$

α is the momentum gain, is susceptible to local minima and needs additional computation for gradient evaluation, and $\Delta w_{ji}(k)$ is weight change based on gradient of the cost function $E_{k,w}$ and k is the iteration number.

4. Neural Network Controller

The function of the neural network in the control mode is to achieve a trajectory tracking of the motor speed and / or position. The Neural network here is acting as an adaptive controller. One commonly used method of adaptive control is the model reference. By this method, the adaptive controller compares the performance of the system under control to the performance of a desired reference model. If the system under control deviates from the reference model, the adaptive controller steers the system back to back the reference model. The block diagram shown in Fig. 3, depicts the general structure of the consider speed control system.

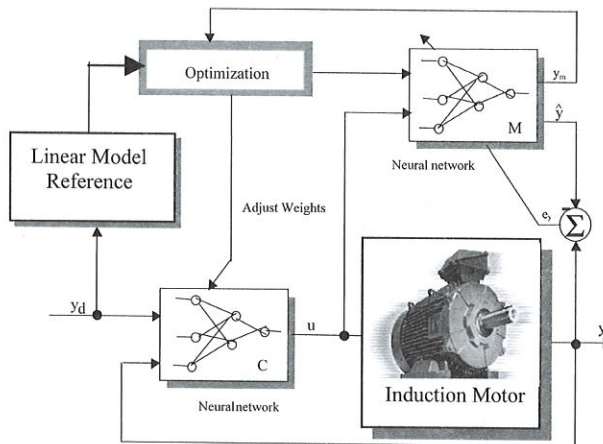


Fig. 3. Configuration of the proposed controller

This scheme includes the system under control, the reference model and the trained neural network. The basic idea of this structure is to equate the dynamics of the reference model to that of the system under control. The neural control algorithm is provided with the predicted system via a recursive on-line identification based on the static feedforward linear network with one hidden layer illustrated by fig. 1. In this approach a neural network provides a prediction of a process future response over a specified horizon. The predictions supplied by the network as passed to a numerical optimization routine which attempts to minimise a specified performance criterion J in the calculation of a suitable control signal.

5. GPC algorithm

Detailed description of the GPC algorithm can be found in [1,2]. Here, it is described only briefly. The principle of predictive control consists in calculating the control input by minimizing a cost function related to desired performances through the choice of some tuning parameters like predictive and control horizons, control weighting and reference model. This cost function, which can be minimized under some constraints on the input and output signals, is expressed in terms of output predictions Fig. 4. Explains the basic idea of model based predictive control. At time t the future control sequence $u(t+k), \dots, u(t+k+N-1)$ is optimized to such that the future output, $y(t+k+j)$ is as close as possible to

reference signal $r(t+k+j)$, measured in the performance index J and subject to constraints. At time t only the first element of the optimal sequence is applied to the real process, then the horizon is shifted and a new optimization at time $t+1$ is done. The reference values are assumed to be a constant sequence as long as additional information is available. The system response is based on future control, models parameters and the actual system state. The predictive control law generates a control sequence, which forces the futures system response to be equal to the reference values. All calculations are done at time t , especially the calculations concerning the future of the system.

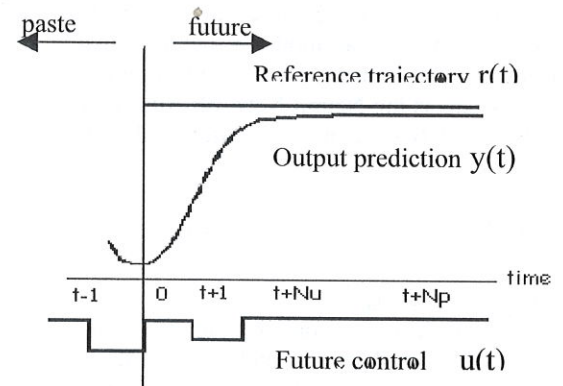


Fig.4. Basic principle of the Generalized Predictive Control

The GPC algorithm uses the CARIMA process model prediction will be deduced by considering a discrete linear system operating in stochastic environment which be represented by the following CARIMA form (Controlled Auto-Regressive and Integrated Moving Average) introduced by Clarke et al. [1,2].

$$A(q^{-1})y(t) = B(q^{-1})u(t-k) + \frac{C(q^{-1})}{\Delta(q^{-1})}e(t) \quad (4)$$

where A, B, C are polynomials in the q^{-1} ; q^{-1} is the backward shift operator ($q^{-1}y(t) = y(t-1)$) and k is the assumed minimum value of the delay-time.

$\Delta(q^{-1}) = (1 - q^{-1})$ is the differential operator, $y(t) \in \mathbb{R}^p$, $u(t) \in \mathbb{R}^p$, $e(t) \in \mathbb{R}^p$, represent the output vector, the input vector and a noise sequence with a zero average and a finite variance, respectively. The considered model leads to two very important features. It enables to easily introduce an integrator in the regulation system to eliminate the influence of the steady state perturbation and improve the quality of the estimation of model parameters of the process. Note that t can be replaced by a discrete time. The optimal predictive controller output vector is determined by minimization of the performance criterion (5).

$$J(r, \hat{y}, \hat{u}) = \sum_{j=N_i}^{N_p} (r(t+j+k) - \hat{y}(t+j+k))^2 + \lambda \sum_{j=1}^{N_u} \Delta u^2(t+j-1) \quad (5)$$

$r(t)$ is the reference trajectory, $\Delta u(t)$ is the process control increment signal, N_i is the minimum cost

horizon, N_p is the prediction horizon, N_u design the control horizon and λ is the control weighting. The first term in the performance criterion refers to the square variation of the predicted process output from the desired reference trajectory, while the second term is added in order to limit the controller output, greater λ yields less active controller output. By minimizing the performance (5), optimal controller output vector u_{opt} is obtained.

6. Neural Network based model of system

The use of a neural network to represent a nonlinear dynamic system is based on the assumption that a nonlinear input/output discrete time model describes the system. A general model structure suitable for representing the dynamics of a wide range of nonlinear systems is the Neural Network Auto-Regressive eXogenous (NNARX) model written in the following form.

$$y(t+1) = f[y(t), \dots, y(t-n+1); u(t), \dots, u(t-m+1)] \quad (6)$$

thus $y = w_r$; $u = i_{qs}$
 where f represents the nonlinear input-output map of the network, n and m are numbers of past output w_r and past input i_{qs} respectively. The non-linear mapping f can be approximated by a feed-forward neural network (e.g. Multi-Layer Perceptron network MLP)

$$y_{mod}(t+1) = f_N[y(t), \dots, y(t-n+1); u(t), \dots, u(t-m+1)] \quad (7)$$

A feed-forward neural network $f_N(\cdot)$ represents the nonlinear input/output map of the network which approximates the plant mapping $f(\cdot)$. Note that the input to the network includes the past values of the plant output but not the past values of the network output. The mode of operation of the NNARX model given in (6), predicts the process output only one step in the future and is used during the network training. In predictive control design long-range, prediction is required and the NNARX model has to operate in the following mode.

$$\hat{y}_{mod}(t+1) = f_N[\hat{y}(t), \dots, \hat{y}(t-n+1); u(t), \dots, u(t-m+1)] \quad (8)$$

A feed-forward neural network $f_N(\cdot)$ is configured to represent the NNARX model by applying the m delayed values of the process input and n delayed values of the process output to the network and assigning its output $\hat{y}(t)$ to be $y(t)$. The Multi-Layer network with one hidden layer is used to approximate the nonlinear function $f_N(\cdot)$. An efficient Newton type training algorithm [Ref], is employed to adjust network weights. The objective of learning process is to found a set of controller weights that minimizes the

error function which is an average of $\|y_d - \hat{y}_t\|^2$, over the set of initial states y_0 .

7. Induction motor model

Using the Park transformation (e.g. Vas., 1990), the three-phase stator windings (sA,sB,sC) can be transformed into equivalent quadrature-phase windings (Sd,Sq). The AC motor dynamic models are described by a set electrical and mechanical non-linear differential equations (Marino, Peseda and Valigi, 1993).

$$\frac{dw_r}{dt} = \frac{n_p M}{J L_r} (\psi_{rd} i_{sq} + \psi_{rq} i_{sd}) - \frac{T_L}{J} \quad (9)$$

$$\frac{di_{sd}}{dt} = \frac{M R_r}{\sigma L_s L_r^2} \psi_{rd} + \frac{n_p M}{\sigma L_s L_r} w_r \psi_{rd} - \frac{M^2 R_r + L_r^2}{\sigma L_s L_r^2} i_{sd}$$

$$\frac{di_{sq}}{dt} = \frac{M R_r}{\sigma L_s L_r^2} \psi_{rq} - \frac{n_p M}{\sigma L_s L_r} w_r \psi_{rd} - \frac{M^2 R_r + L_r^2}{\sigma L_s L_r^2} i_{sq} + \frac{1}{\sigma L_s} u_{sq} \quad (10)$$

$$\frac{d\psi_{rd}}{dt} = -\frac{R_r}{L_r} \psi_{rd} - n_p w_r \psi_{rq} + \frac{R_r}{L_r} M i_{sd}$$

$$\frac{d\psi_{rq}}{dt} = -\frac{R_r}{L_r} \psi_{rq} + n_p w_r \psi_{rd} + \frac{R_r}{L_r} M i_{sq} \quad (11)$$

$$\sigma = 1 - \frac{M^2}{L_s L_r}$$

where i , u , ψ denote current, voltage and flux linkage respectively. Subscripts r and s stand for rotor and stator. w_r is the rotor speed, d and q denote direct and quadrature components of the vectors with respect to the fixed stator reference frame, L and R are the autoinductances and resistances, M is the mutual inductance and T_L is the load torque. In the following, we plan to design an neuro-predictive controller which should stabilize the system and cause $y(t)$ to track a bounded reference $r(t)$ asymptotically. In order words, the aim is to determine the control input $u(t)$ for all $t \geq t_0$, so that

$$\lim_{t \rightarrow \infty} \|r(t) - y(t)\| \leq \varepsilon, \text{ for some specified constant } \varepsilon \geq 0$$

8. Simulation results

In this section, we present some simulation results to evaluate the effectiveness of the proposed control scheme for an induction motor. To test the fidelity of the proposed control scheme, the following numerical values of the system parameters were used:

Rated data of the simulated induction motor

Rated values	Power	4	kW
	Frequency	50	Hz

	Voltage Δ/Y	220/380	V
	Current Δ/Y	15/8.6	A
	Speed	1440	rpm
	Pole pair (p)	2	
Rated parameters	R_s	1.2	Ω
	R_r	1.8	Ω
	L_s	0.1554	H
	L_r	0.1564	H
	M	0.1500	H
Constant	J	0.013	kg.m ²

In each of the simulations, the neural network have been chosen with one hidden layer with five neurons. A bias has been used in the hidden layer. The initial values of the weights were chosen randomly in the interval $[0,1]$. The parameters of the neural network are adjusted using the back-propagation learning algorithms and the following parameters were chosen constants as $\eta = 0.2$, $\alpha = 0.5$. The results obtained using a computer simulation, are interesting, considering the relative complexity of the system and the presence strong nonlinearities. The performances of the neuro-predictive controller are evaluated under a variable speed reference signal and the tracking ability of the controller is well demonstrated by plots. In all cases, the GPC parameters were chosen as $N_i=1$; $N_p=10$; $N_u=2$ and $\lambda=0.01$. The presented in Figs. 5,6,7,8,9,10,11, show that the proposed control system works correctly. The plots of these figures show the performance as the neural network controller for variety of step changes in the desired set point. It can be observed that, other than the presence of the noise in the output, the control and setpoint following is satisfactory. To demonstrate the robustness of the proposed controller, a different type of trajectory was combined in this test (square-triangular and square-sinusoidal reference speed track). One can see from the Figs. 9,10, that the results were very successful. From these results, it is apparent the prediction control based on neural network models scheme was superior than that the conventional controllers, such as the Proportional Integral and Differential (PID). The controller envisaged is capable of maintaining a high tracking accuracy even in the presence of sudden disturbances such as load of electric transients.

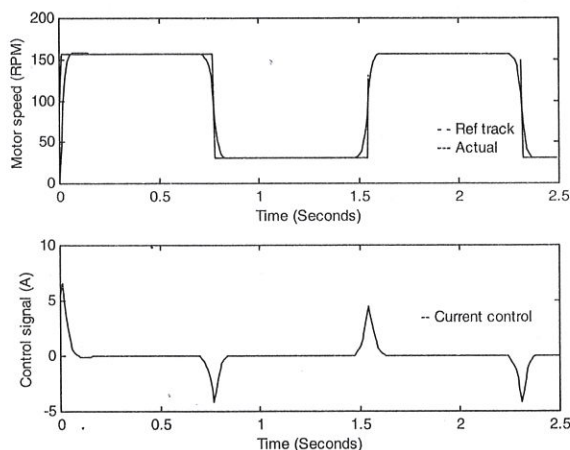


Fig. 5. Speed control of square-wave reference track with neural network controller ($N_i=1$; $N_p=10$; $N_u=2$; $\lambda=0.01$ no-load)

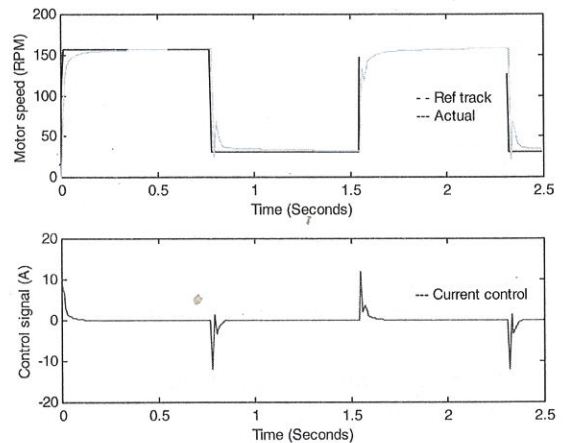


Fig. 6. Speed control of square-wave reference track using a neuro-predictive controller with load $T_r=2Nm$.

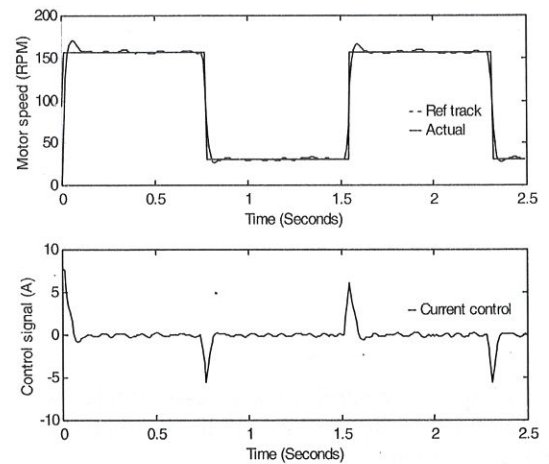


Fig. 7. Speed control of square-wave reference track with neural network controller in the presence of noise

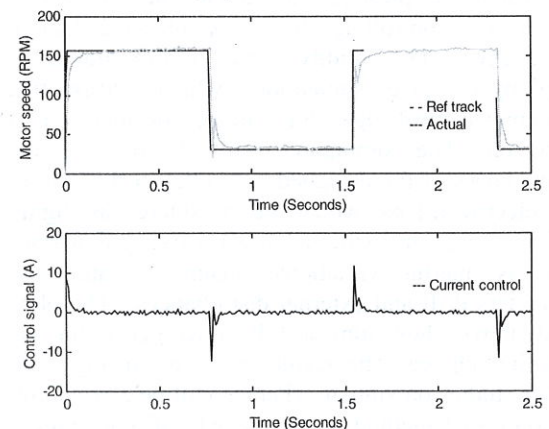


Fig. 8. Speed control of square-wave reference track using a neuro-predictive controller with stochastic load changes

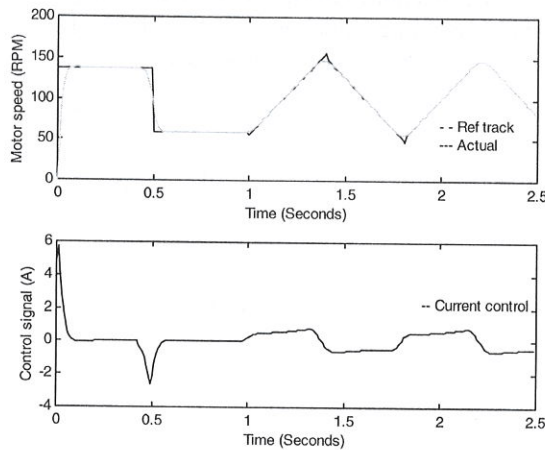


Fig. 9. Speed control of square-triangular reference track using a neuro-predictive controller ($N_i=1$; $N_p=10$; $N_u=2$; $\lambda=0.01$ no-load)

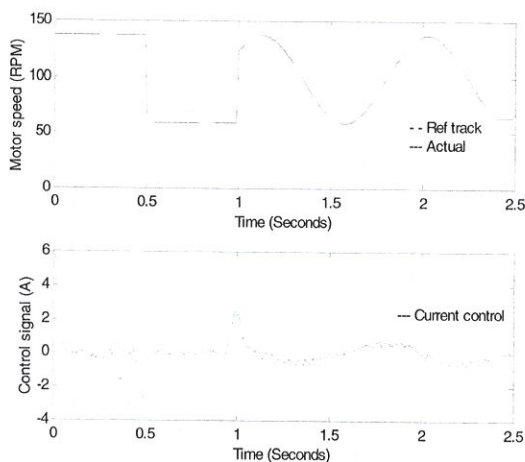


Fig. 10. Speed control of square-sinusoidal wave reference track using a neuro-predictive controller in the presence of noise

9. Concluding remarks

This paper has presented a neural network based scheme for controlling the speed of an induction motor drive. The validity of the control strategy is tested by computer simulations, which confirm that predictive control algorithm can be considered as a proper one. The simulation results demonstrates the effectiveness of the designed controller in the control for electric drives, which are considered as highly nonlinear systems with unknown varying dynamics such as inertia variations, magnetic saturation, parameters drift and external disturbances. The plots listed above show almost full convergence for the controller choice. The resulting are promising and further studies on similar schemes will be carried out. The proposed method it is believed will constitute a major step in the evolution of intelligent control of complex mechatronic systems.

10. References

- [1] Clarke, D.W, Mohtadi, C. and Tuffs P.S (1987). Generalized Predictive Control – Part I and II, *Automatica*, vol.23,2,pp.137-160.
- [2] Clarke, D.W and Mohtadi, C. (1989). Properties of Generalized Predictive Control. *Automatica*, vol.25,6,pp.859-875.
- [3] Soeterboek, R. (1992). Predictive Control: A unified approach. Prentice Hall, Englewood Cliffs.
- [4] Tonn, J.J. and Boom, Van Den. (1996). Model Based Predictive Control : Status and Perspectives. *IFAC World Congress*.pp1-10.
- [5] Liaw, C.M, Kung, Y.S and Wu, C.M. (1991). Design and implementation of a high-performance field-oriented induction motor drive. *IEEE Trans. Ind. Electron.*,vol.38,4,pp.275-282.
- [6] Wishart, M.A, and Harley, R.G. (1995). Identification and control of induction machines using artificial neural networks. *IEEE Trans. Ind. Applicat.*, vol.31,pp.612-619.
- [7] Kung, Y.S, Liaw, C.M and Ouyang, M.S. (1995). Adaptive speed control for induction motor drives using neural networks. *IEEE Trans. Ind. Electron.* Vol.42,1,pp.25-32.
- [8] Chen, T.C. and Sheu, T.T (1992). Robust speed-controlled induction motor drive based on model reference with neural networks. *Inter. Journ. Of Knowledge Based Intelligent Engineering System*. Vol.3,3,pp.162-171
- [9] Narendra, K.S and K. Parthasarathy (1990). Identification and control for dynamical systems using neural networks . *IEEE Trans. Neural Networks*, NN-1,1,4-27.
- [10] Narendra, K.S. (1996). Neural networks for control : Theory and practice. *Proceedings of the IEEE*, 84,10,1385-1406.
- [11] Chen, F.C. (1990). Back-propagation neural networks for nonlinear self-tuning adaptive control. *IEEE. Control System Magazine, Special Issue on neural Networks for control*.pp.45-48
- [12] McClellan, J. L., Rumelhart, D. E. and PDP research group. (1986). Parallel distributed processing : Exploitation of the microstructure of cognition.Vol.2, Badford books, Cambridge, MA