Journal of ELECTRICAL ENGINEERING, VOL. 58, NO. 1, 2007, 10–18

A SIMPLE HYSTERESIS PI BASED NEURAL CONTROLLER USED FOR SPEED CONTROL OF AN INDIRECT FIELD ORIENTED INDUCTION MACHINE DRIVE

Abdallah Miloudi [*] — **Eid Al-radadi** [**] — **Azeddine Draou** [**]

This paper presents an original hysteresis PI based neural controller for speed control of an indirect rotor flux oriented controlled (IRFOC) induction motor drive. An original hysteresis PI controller is first proposed. Its simulated performances in speed control and rotor resistance estimation are then compared to those of a classical PI controller. The proposed hysteresis PI controller provides better dynamic performances than the classical PI controller but it has one drawback. The hysteresis PI cannot deal with down step speed tracking below a certain limit. The artificial neural network generalization capacity is then used to deal with this drawback. The simulated input-output non linear relationship of the proposed controller during startup and load disturbance rejection is learned off-line using a feed-forward linear network with one hidden layer. The simulation of the neural network controlled system shows promising results. The motor reaches the reference speed rapidly and without overshoot, step commands are tracked with almost zero steady state error and no overshoot, load disturbances are rapidly rejected and variations of some of the motor parameters are fairly well dealt with.

K e y w o r d s: induction motor drive, vector reference IRFOC, neural controller, speed control

1 INTRODUCTION

With the apparition of the indirect rotor field oriented control (IRFOC), induction machine drives are beginning to become a major candidate in high performance motion control applications. In the complex machine dynamics, this decoupling technique permits independent control of the torque and the flux [1], [6–8].

IRFOC however is parameter sensitive [5–6]. Heating and saturation of the motor causes detuning in the decoupling operation and introduces errors in the torque and field motor output values. The design of robust controllers allowing parameter variation adaptation of the decoupling operation is then necessary.

PID classical controllers find some difficulties in dealing with the detuning problem. Artificial neural networks (ANN) can be used to design numerical controllers in order to maintain high dynamic performances even when detuning occurs.

ANN's have been proven to be universal approximators of non-linear dynamic systems [2]. They are able to emulate any complex non linear dynamic system by using an appropriate multilayer neural network. After being used for many years in pattern recognition and signal and image processing applications, ANN's are now employed in a larger class of scientific disciplines. Many applications have been reported in power electronics, including fault detection and diagnosis in electrical machines, power converter control and the high performance control of electrical drives [3]. This high interest in the use ANN's in the different scientific disciplines is due to their inherent parallelism which allows for high speed processing and permits implementation of real time applications. They are also able to perform in noisy environments and have the capacity of generalization that permits them to be tolerant to faults and missing data [4].

In this paper an original hysteresis PI controller for speed adjustment of a reference voltage indirect rotor field oriented control induction machine drive is presented. This controller provides better dynamic performances than the classical PI controller but it has one drawback. The hysteresis PI controller cannot deal with important down step speed tracking because over certain down step reference values, the hysteresis PI generates a positive command torque that increases the motor speed when we need to decrease it. The generalization capacity of the artificial neural network is then used to generalize the up step speed tracking during start up to the down step speed tracking and eliminate this drawback. The simulated input-output non linear relationship of the proposed controller during startup and load disturbance rejection is learned off-line using an appropriate neural network in order to realize a robust neural controller.

2 INDUCTION MACHINE DYNAMIC MODEL

Figure 1 gives the block diagram structure of an induction motor speed control using vector reference IRFOC scheme. It consists mainly of a squirrel-cage induction motor, a voltage-regulated pulse width modulated inverter, a speed controller and an IRFOC block.

The model of the squirrel-cage induction machine can be expressed in the d-q axes using the following equations:

$$\dot{X} = AX + BU \tag{1}$$

[*] University centre of Saida, BP 138, En-Nasr, Saida 20000, Saida Algeria; [**] Department of Electronics and Control, Madinah College of Technology, Madinah, Saudi Arabia E-mail: adraou@yahoo.com

ISSN 1335-3632

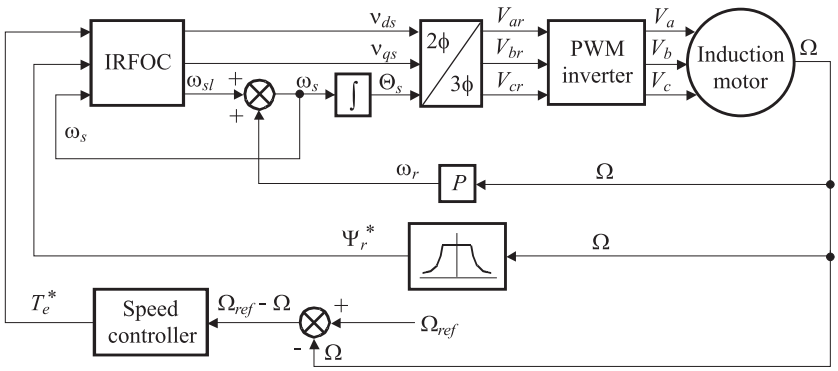

Fig. 1. Indirect rotor field orientation control.

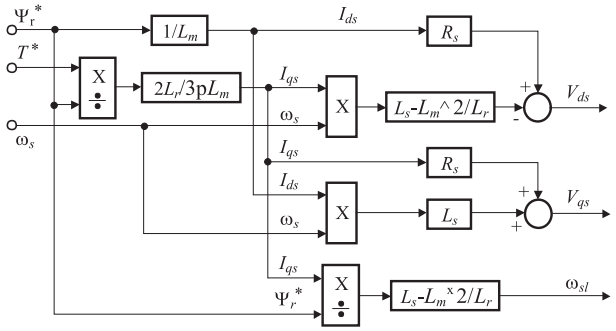

Fig. 2. IRFOC structure.

Where:

$$A=\begin{bmatrix} -\frac{1}{\sigma T_s} & \omega_s+\frac{1-\sigma}{\sigma}\omega_r & \frac{L_m}{\sigma L_s T_r} & \frac{L_m}{\sigma L_s}\omega_r \\ -\omega_s-\frac{1-\sigma}{\sigma}\omega_r & -\frac{1}{\sigma T_s} & -\frac{L_m}{\sigma L_s}\omega_r & \frac{L_m}{\sigma L_s T_r} \\ \frac{L_m}{\sigma L_r T_s} & -\frac{L_m}{\sigma L_r}\omega_r & -\frac{1}{\sigma T_r} & \omega_s-\frac{1}{\sigma}\omega_r \\ \frac{L_m}{\sigma L_r}\omega_r & \frac{L_m}{\sigma L_r T_s} & -\omega_s+\frac{1}{\sigma}\omega_r & -\frac{1}{\sigma T_r} \end{bmatrix}, \quad (2)$$

$$B=\frac{1}{\sigma L_s}\begin{bmatrix} 1 & 0 \\ 0 & 1 \\ -\frac{L_m}{L_r} & 0 \\ 0 & -\frac{L_m}{L_r} \end{bmatrix}, \; X=\begin{bmatrix} i_{ds} \\ i_{qs} \\ i_{dr} \\ i_{qr} \end{bmatrix}, \; U=\begin{bmatrix} \nu_{ds} \\ \nu_{qs} \end{bmatrix}. \quad (3)$$

The electromagnetic torque and the mechanical equations can be written as follows:

$$T_e=\frac{3}{2}pL_m(i_{dr}i_{qs}-_i qr_i ds\,, \quad (4)$$

$$J\frac{\mathrm{d}\Omega_r}{\mathrm{d}t}+f\Omega_r=T_e-T_L \quad (5)$$

where J is the moment of inertia, f the viscous friction coefficient and T_L the load torque.

3 VOLTAGE REFERENCE IRFOC MODEL

In the voltage reference IRFOC scheme (Fig. 1), the command values of the electromagnetic torque, the rotor flux and the stator frequency are delivered to the IRFOC block to generate command values of the frequency and the reference voltage vector d-q frame components. The slip frequency is added to the rotor frequency and the result is integrated to evaluate the stator angle. The reference voltage vector d-q frame components along with the stator angle are delivered to the Park inverse transformation block to evaluate the reference voltage vector three phase system components. These components are delivered to a sine triangle PWM to generate pulses to control the power switches in the inverter.

The d-q equations of the motor in the synchronous reference frame are given by:

$$R_r i_{qr}+\omega_{sl}\psi_{dr}=0\,, \quad (6)$$

$$R_r i_{dr}+\frac{\mathrm{d}}{\mathrm{d}t}\psi_{dr}=0\,, \quad (7)$$

$$L_m i_{qs}+L_r i_{qr}=0\,, \quad (8)$$

$$L_m i_{ds}+L_r i_{dr}=\psi_{dr}\,, \quad (9)$$

$$\nu_{ds}=R_s i_{ds}-\sigma L_s\omega_s i_{qs}+\sigma L_s\frac{\mathrm{d}i_{ds}}{\mathrm{d}t}+\frac{L_m}{L_r}\frac{\mathrm{d}\psi_{dr}}{\mathrm{d}t}-\frac{L_m}{L_r}\omega_s\psi_{qr}\,, \quad (10)$$

$$\nu_{qs}=R_s i_{qs}+\sigma L_s\omega_s i_{ds}+\sigma L_r\frac{\mathrm{d}i_{qs}}{\mathrm{d}t}+\frac{L_m}{L_r}\frac{\mathrm{d}\psi_{qr}}{\mathrm{d}t}+\frac{L_m}{L_r}\omega_s\psi_{dr} \quad (11)$$

where R_s, R_r L_r, L_s, L_m are motor parameters, i_{dr}, i_{qr}, i_{ds}, i_{qs}, ψ_{dr}, ψ_{ds} are motor currents and fluxes, and ω_{sl} is slip frequency. Under field orientation condition ($\psi_{dr}=\psi_r$ and $\psi_{qr}=0$) and using equations (4), (6)–(9) we can obtain the following equations:

$$\omega_{sl}=\frac{L_m}{\psi_r}\Big(\frac{R_r}{L_r}\Big)i_{qs}\,, \quad (12)$$

$$T_e=\frac{3}{2}p\frac{L_m}{L_r}\psi_r i_{qs}\,, \quad (13)$$

$$\Big(\frac{L_r}{R_r}\Big)\frac{\mathrm{d}\psi_r}{\mathrm{d}t}+\psi_r=L_m i_{ds}\,. \quad (14)$$

In the steady state mode equation (14) becomes

$$\psi_r=L_m i_{ds} \quad (15)$$

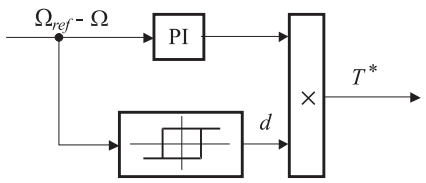

Fig. 3. Hysteresis PI controller structure.

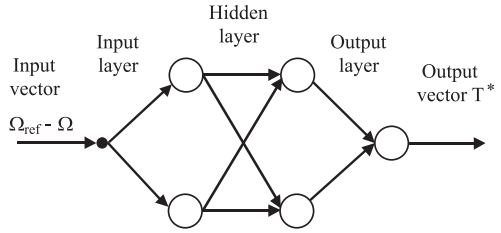

Fig. 4. Neural controller structure.

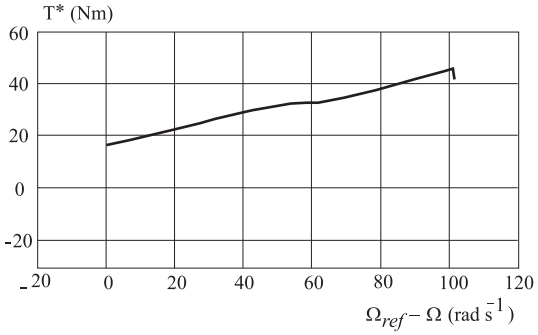

Fig. 5. Hysteresis PI controller Input-Output simulated relationship.

and equations (10) and (11) become:

$$\nu_{ds} = R_s i_{ds} - \sigma L_s \omega_s i_{qs}\,, \tag{16}$$

$$\nu_{qs} = R_s i_{qs} + L_s \omega_s i_{ds}\,. \tag{17}$$

From equation (12), (13), (15), (16) and (17), the IRFOC block structure can be obtained as shown in Fig. 2.

The Park inverse transformation equations are given by:

$$V_{ar} = A\cos(\theta_s + \varphi)\,, \tag{18}$$

$$V_{br} = A\cos(\theta_s + \varphi - 2\pi/3)\,, \tag{19}$$

$$V_{cr} = A\cos(\theta_s + \varphi + 2\pi/3)\,. \tag{20}$$

Where V_{ar}, V_{br} and V_{cr} are the three phase system reference voltage vector components, A is the amplitude and φ is the angle of the d-q frame reference voltage vector.

$$A = \sqrt{\nu_{ds}^2 + \nu_{qs}^2} \text{ and } \varphi = \begin{cases} \operatorname{atan}\frac{\nu_{ds}}{\nu_{qs}} & \text{if } \nu_{ds} \geq 0 \\ \operatorname{atan}\frac{\nu_{ds}}{\nu_{qs}} + \pi & \text{if } \nu_{ds} < 0 \end{cases} \tag{21}$$

4 HYSTERESIS PI CONTROLLER STRUCTURE

To replace the speed controller of the IRFOC block diagram structure given in Fig. 1, we propose the use of a hysteresis PI controller presented in Fig. 3.

When the motor starts, the error is positive making the hysteresis output equal to $+1$. The hysteresis PI controller acts then like a classical PI controller by decreasing the value of the speed error towards the hysteresis lower limit. When crossing this limit, the hysteresis output becomes -1 and changes the command torque sign. This will in turn change the sign of the sliding frequency ω_{sl} and the quadrature current i_{qs}, resulting in an instantaneous decrease in the amplitude and angle of the reference voltage vector. The speed error is then forced to increase towards the hysteresis upper limit. When crossing this limit, the hysteresis output value becomes $+1$ and the controller acts again like a classical PI controller decreasing the speed error back towards the hysteresis lower limit. This forward and backward crossing of the hysteresis limits will go on until stabilization of the speed occurs. This original proposed controller can adjust the speed of the motor only at starting mode or when load disturbances occur, it cannot deal with important down step speed tracking operation because, as we are going to prove, under a down step reference speed limit, the hysteresis PI generates a positive command torque leading to an increase in the motor speed when we need to decrease it.

Consider $u(t)$ to be the output from the PI controller:

$$u(t) = K_p\big(\Omega_{ref} - \Omega(t)\big) + K_i \int_0^t \big(\Omega_{ref} - \Omega(\tau)\big)\mathrm{d}\tau\,. \tag{22}$$

If u_k is a sample value of $u(t)$ with sampling period T_s, then for a step reference speed an approximation of u_k is given by:

$$u_k = K_p\big(\Omega_{ref} - \Omega_k\big) + T_s K_i\Big(k\Omega_{ref} - \sum_{i=0}^{k-1}\Omega_i\Big). \tag{23}$$

If the reference speed is reached at $I = n$ then for $k \geq n$ the hysteresis PI controller maintains the motor speed in the vicinity of the reference speed so that we have $\Omega_k \approx \Omega_{ref}$. Therefore the output from the PI controller is maintained nearly constant and given by:

$$u^* \approx T_s K_i\Big(n\Omega_{ref} - \sum_{i=0}^{n-1}\Omega_i\Big). \tag{24}$$

For a new step speed reference Ω_{ref1} which occurs at $t = (k+1)T_s$ we can use equation (22) to obtain:

$$u_{k+1} = u^*\big(K_p + T_s K_i\big)\big(\Omega_{ref1} - \Omega_{ref}\big)\,. \tag{25}$$

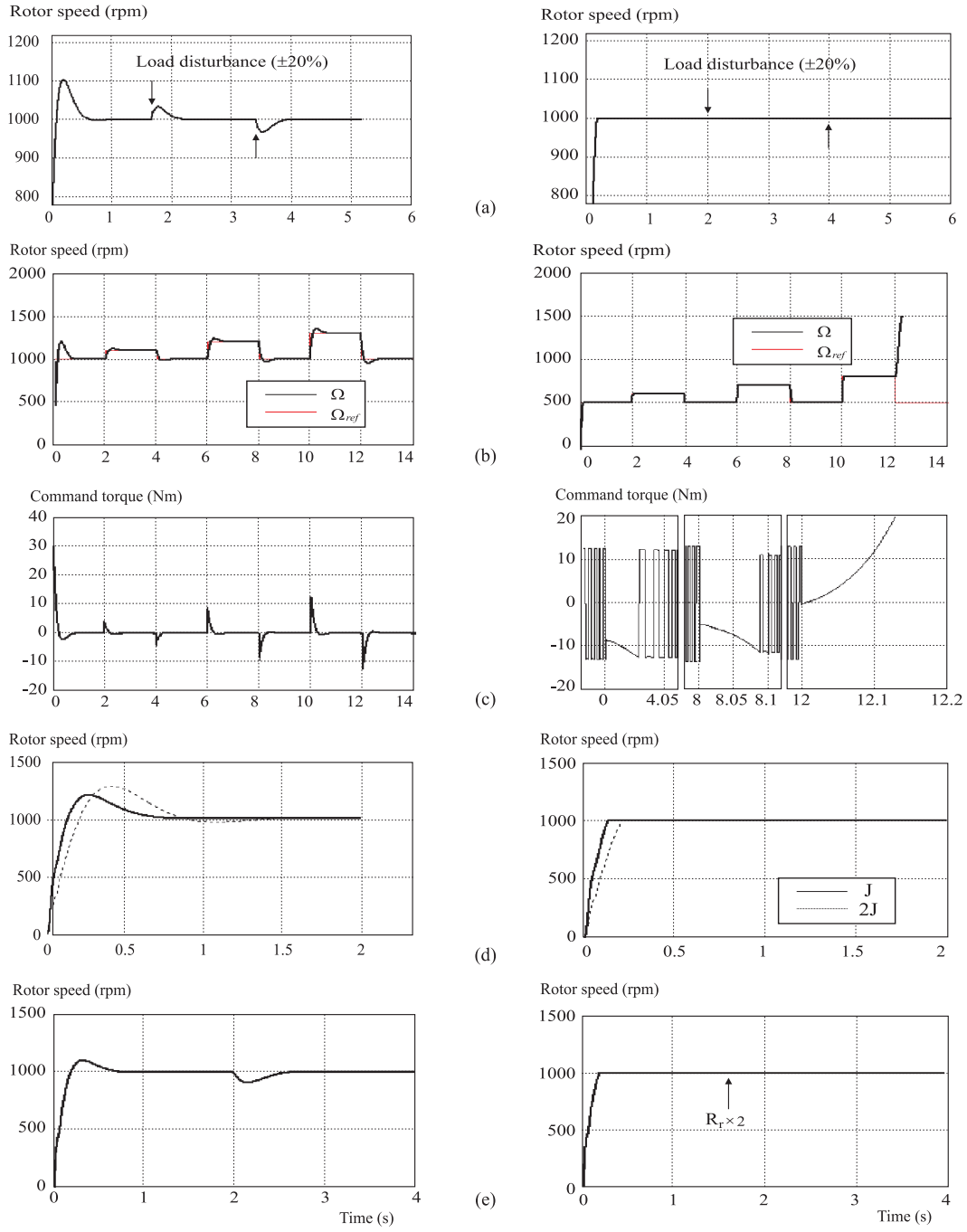

Fig. 6. Performance comparison between a classical PI controller and a hysteresis PI controller in speed control of a voltage reference IRFOC induction motor drive.

If $T_s \ll 1$ then

$$u_{k+1} \approx u^* + K_p \Delta\Omega_{ref} \,. \tag{26}$$

For no load speed tracking, a step decrease of the reference speed with $\Delta < -u^*/K_p$ leads to a negative output from the PI controller. Since the speed error is negative, the hysteresis PI controller generates a positive command torque that increases the motor speed drifting it far away from the reference speed. Therefore the hysteresis PI controller cannot deal with an important down step speed tracking.

However, the controller input-output data obtained during simulation of the start up (up step speed tracking) and load disturbance rejection could be used to design a neural network based controller whose generalization capacity would permit it to deal with down step speed tracking operation.

5 NEURAL CONTROLLER STRUCTURE

Neural networks can be employed in advanced intelligent control applications by making use of their non

linearity learning, parallel processing and generalization capacities [4, 5, 9].

A neural network is constituted of densely interconnected neurons. A neuron is a computing node. It performs the multiplication of its inputs by constant weights, sums the results, shifts it by a constant bias and maps it to a non linear activation function before transferring it to its output.

A feed-forward neural network is organized in layers of neurons: an input layer, one or more hidden layers and an output layer. The inputs to each neuron of the input layer are the inputs to the network. The inputs to each neuron of the hidden or output layer are the outputs from the neurons of the preceding layer.

The mathematical model of a neuron is given by:

$$y = \Big(\sum i = 1^n w_i x_i + b\Big). \tag{27}$$

Where y is the output from the neuron, $(x_1, x_2, \ldots, x_n)$ are the inputs to the neuron, $(w_1, w_2, \ldots, w_n)$ are the corresponding weights, and b is the bias of the neuron. The activation function f is generally the logarithmic or tangent sigmoidal function. For a logarithmic sigmoidal activation function the output from the neuron is given by:

$$y = \frac{1}{1 + e^{-\left[\sum_{i=1}^{n} w_i x_I + b\right]}}\,. \tag{28}$$

In a supervised off-line control, the proposed original hysteresis PI controller can be replaced by a neural network that learns the mapping form of the controller input-output relationship by adapting its parameters to a training set of examples of what it should do.

To design a neural network for a supervised off-line control, the following steps are necessary: Selection of the network structure: The number of layers, the number of neurons for each layer and the number of inputs to the network.

- Presentation of the training data: The network input and the target output vectors.
- Learning: Adaptation of the network parameters (Weights and bias of each neuron) in such a way that the network output gets as close as possible from the target output. Most of the learning algorithms perform the adaptation of weights and biases of the network iteratively until the error between the target vector and the output of the network becomes less than an error goal.

The structure of the neural controller chosen in this paper is given by Fig. 4. The controller is a three layers feed-forward linear network with two neurons in the input and hidden layer and one neuron in the output layer. The speed error is the only input to the controller. The activation functions are logarithmic sigmoid for the input and hidden layer neurons and linear for the output neuron.

The set of examples used to train the network is composed of the original hysteresis PI controller simulated input and output values, obtained during starting up and load disturbance of the motor. The training rule used to adapt the network neurons weights and biases is that of Levenberg Marquardt.

6 SIMULATION RESULTS

In order to test the dynamic performances and the robustness of the proposed controller in speed control of an indirect field oriented induction machine drive, some simulation works have been performed. The parameters of the motor used in the simulation are given in Table 1.

6.1. Hysteresis PI controller Performances

By using the reference voltage IRFOC induction machine drive control structure illustrated by Fig. 1 where the controller block is replaced by the hysteresis PI controller given by Fig. 3, some simulations have been executed. The values of the PI controller gains used are $K_p = 0.4$ and $K_i = 2$ and the hysteresis limits are ± 0.001. Simulations have been performed using a speed sampling frequency of 10 KHz.

Figure 6 shows a performance comparison between a classical PI controller and a hysteresis PI controller in speed control of a voltage reference IRFOC induction motor drive.

Figure 6a shows the controllers settling performance and the disturbance rejection capability. Initially the machine is started up with a 10 Nm load. At 2 s, a 2 Nm load disturbance is applied during a period of 2 s.

For the classical PI controller, the speed of the motor reaches Ω_{ref} at 1.4 s after making an overshoot of 10.31 %. The controller rejects the 2 Nm load disturbance after 0.65 s with a maximum speed dip of 32.8 rpm (3.28 %).

For the hysteresis PI controller, the speed of the motor reaches Ω_{ref} at 0.2 s with almost no overshoot. It then begins to oscillate inside a 0.4 % error strip around Ω_{ref}. The load disturbance application has almost no effect on the speed of the motor which stays inside the 0.4 % error strip.

Figure 6b shows the controllers speed tracking performance under no load. Seven tracking tests have been performed. The reference speed set used is $\Omega_{ref} = 1000$, 1100, 1000, 1200, 1000, 1300 and 1000 rpm at $t = 0$, 2, 4, 6, 8, 10 and 12 s. From these tests, it is clear that the hysteresis PI controller cannot deal with a 300 rpm decrease in the reference speed. The command torque figure shows a negative command torque output when the 100 and 200 rpm down step speed tracking are applied, while it shows a small positive command torque which increases rapidly when the 300 rpm down step speed tracking is applied.

Figure 6c shows the controllers reaction to moment of inertia variation. The motor's speed is simulated, under no load, for moments of inertia equal to J and $J \times 2$.

Table 1. Induction Machine Parameters

2 pairs of poles, 50 Hz	$R_s = 4.85\,\Omega$	$L_s = 274$ mH
220/380 V, 6.4/3.7 A	$R_r = 3.805\,\Omega$	$L_r = 274$ mH
2 hp, 1420 rpm	$L_m = 258$ mH	
$J = 0.031\,\mathrm{kgm}^2$	$f = 0.00114$ Nms	

In the case of the classical PI controller, the simulation results show that multiplying J by 2 affects both the time to peak and the overshoot values. The time to peak value changes from 0.269 s to 0.415 s and the overshoot value changes from 20 % to 28 %. In the case of Hysteresis PI controller, only the speed settling time is affected. Its value changes from 0.15 s to 0.22 s.

Figure 6d shows the controllers reaction to rotor resistance variation. The motor is started up with a load of 10 Nm. The rotor resistance is supposed to double at 2 s.

The classical PI controller rejects the rotor resistance disturbance after 1.4 s with a speed dip of 90.5 rpm (9.05 %). For the hysteresis PI controller, rotor resistance variation seems to have no effect on the speed of the motor which stays inside the 0.4 % error strip.

By comparing these results, one can say that adding a hysteresis to a classical PI controller transforms it to a high performance robust controller. Its only drawback is its incapacity to deal with down step speed tracking.

This drawback could be eliminated by using the generalization capacity of the artificial neural network. A neural network controller is designed by learning off line the simulated input-output non linear relationship of the hysteresis PI controller during startup and load disturbance rejection. This neural controller uses the startup operation, which is considered to be an up step speed tracking, in order to generalize it to the down step speed tracking.

6.2. Neural controller design and performances

Figure 5 shows the hysteresis PI controller input-output relationship during start up and load disturbance rejection.

This relationship could be represented by a one variable non linear function symbolized by IOF (Input Output Function) and given by:

$$T^* = IOF\big(\Omega_{ref} - \Omega\big)\,. \tag{29}$$

An approximation of this function could be obtained using the neural network illustrated by Fig. 4. The parameters of the network (Neurons weights and biases) have been obtained using Levenberg Marquardt backpropagation training rule and are given by:

$$W_1 = \begin{bmatrix} -743.031935 \\ 0.0033127 \end{bmatrix}, \qquad B_1 = \begin{bmatrix} -3.884927 \\ 1.760102 \end{bmatrix},$$

$$W_2 = \begin{bmatrix} -131.599808 & -26.131695 \\ -121.959881 & 55.252943 \end{bmatrix}, B_2 = \begin{bmatrix} 25.161492 \\ -44.227922 \end{bmatrix}, \tag{30}$$

$$W_3 = \begin{bmatrix} -201.392981 & 236.099844 \end{bmatrix}, B_3 = -16.736723\,.$$

The resulting neural network is used as a numerical controller to replace the hysteresis PI controller in the IRFOC induction machine drive control structure illustrated by Fig. 1.

Figure 7 gives a dynamic performance comparison between the hysteresis PI controller and the designed neural controller during start up and load disturbance rejection. A 10 KHz speed sampling frequency has been used.

Initially the machine is started up with a load of 10 Nm. At 1 s, a 5 Nm load is applied during a period of 1 s.

Figure 7a shows the dynamic performances of the hysteresis PI controller. As it is shown by the figure and as given by Table 2, speed error crossings of the hysteresis limits will result in changing the hysteresis output from $+1$ to -1. This will result in changing the sign of the command torque steady state value, and then the sign of i_{ds} and ω_{sl}. This will then lead to an instantaneous decrease in ω_s and ν_{qs}, and an instantaneous increase

Table 2. Hysteresis PI controler steady state performance Table

	T_e^* (N.m)	i_{qs} (A)	ω_{sl} (rad/s)	ω_s (rad/s)	ν_{ds} V	ν_{qs} V	A V	φ ∘
$d = 1$	16.6	8.43	43.5	253	−53.2	227.6	233.7	103.2
$d = -1$	−16.6	−8.43	−43.5	166	56.5	81.7	99.3	55.3
Mean Value	10	5.1	26.2	235.5	−31.9	198.4	207.3	93.4

Table 3. Neural controller steady state performance Table

	T_e^* (N.m)	i_{qs} (A)	ω_{sl} (rad/s)	ω_s (rad/s)	ν_{ds} V	ν_{qs} V	A V	φ ∘
$\varepsilon_\Omega > 0^+$	16.7	8.50	43.8	253.2	−53.8	228.1	234.4	103.3
$\varepsilon_\Omega < 0^-$	−16.7	−8.53	−43.95	165.5	56.9	80.8	98.8	55.9
Mean Value	10	5.1	26.3	235.6	−31.6	198.6	207.2	93.6

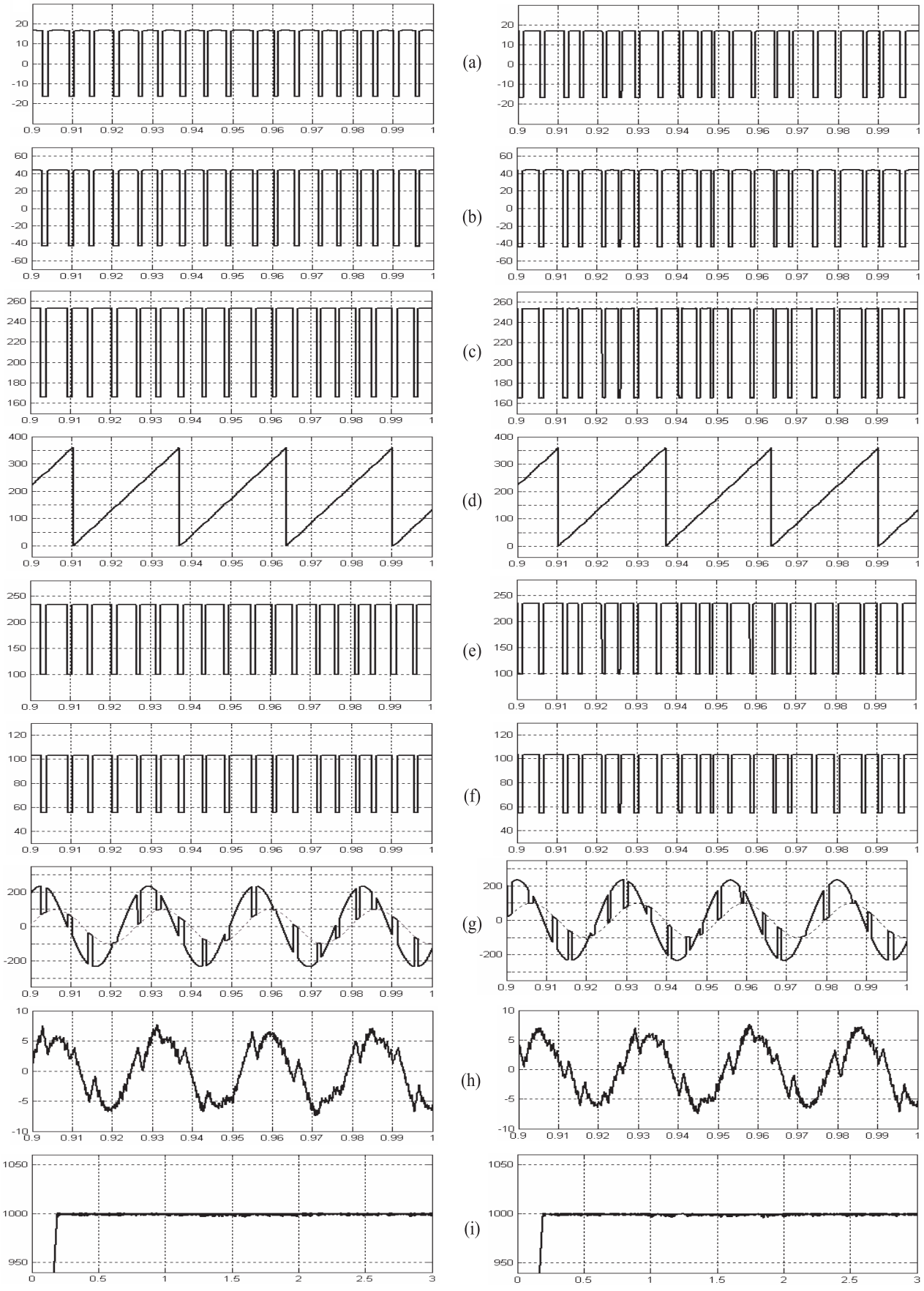

Fig. 7. Dynamic performance comparison between the hysteresis PI controller (left) and the neural controller (right) in speed control of the voltage reference IRFOC motor drive. From the top, in both columns: (a) — command torque (Nm), (b) — slip frequency (rad/s), (c) — stator frequency (rad/s), (d) — stator angle (deg), (e) — reference voltage, d-q frame amplitute (V), (f) — reference voltage, d-q frame angle (deg), (g) — phase "a" of he line reference voltage (V), (h) — stator current (A), (i) — rotor speed (rpm).

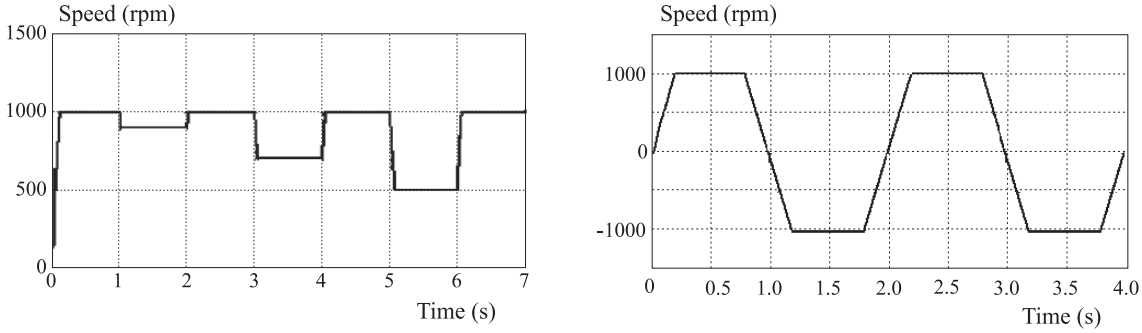

Fig. 8. Hysteresis PI based Neural Controller Tracking Performances.

in ν_{ds}. The reference voltage vector amplitude and angle will then decrease instantaneously.

Depending on the value of the hysteresis output, the first of the three phase components of the reference voltage vector is shown to go along different sinusoidal curves (equ. 18 where A and φ are given by Table 2). The instantaneous change of the reference voltage vector has a direct effect on the stator current which is shown to be distorted when the hysteresis output is equal to -1. This adequate distortion of the stator current forces the rotor speed to stay glued to the reference speed. The speed of the motor reaches Ω_{ref} at 0.194 with 0.07 % overshoot. It then begins to oscillate inside a 0.3 % error strip around Ω_{ref}. The load disturbance application has almost no effect on the speed of the motor which stays inside a 0.4 % error strip. We can say then that the hysteresis PI controller generates appropriate stator current distortion in order to obtain high performance induction motor speed control.

Figure 7b shows the dynamic performances of the neural controller. As it is shown by the figure and as given by Table 3, it is evident that the neural controller has perfectly learned the dynamic performances of the hysteresis PI. The steady state values of the system parameters are almost the same as obtained using the hysteresis PI. The only difference is the switching time sequence of the command torque which results in a slightly better speed control. The speed of the motor reaches Ω_{ref} at 0.195 s with 0.07 % overshoot. It then begins to oscillate inside a 0.2 % error strip around Ω_{ref}. The load disturbance application has almost no effect on the speed of the motor which stays inside a 0.4 % error strip.

Figure 8 shows the speed tracking performance of the neural controller under no load. In Fig. 8a, seven step tracking tests have been performed. The reference speed set used is $\Omega_{ref} = 1000$, 900, 1000, 700, 1000, 500 and 1000 rpm at $t = 0$, 1, 2, 3, 4, 5 and 6 s. From this figure it is clear that the neural controller has generalized the 100 rpm down step speed tracking to the 300 and 500 rpm speed tracking. Using a neural controller to learn the dynamic performances of the hysteresis controller has then solved its down step tracking problem.

In Fig. 8a, a 5000 rpm/s slope trapezoidal command speed is used. The speed of the motor tracks the trapezoidal command speed with almost zero speed error since start up. The neural controller seems to accomplish a very good speed tracking.

Simulation given by Fig. 9 examines the robustness of the neural controller to machine parameters variations.

Figure 9a shows the neural controller reaction to IR-FOC detuning. The motor is started up with a load of 10 Nm. At 1 s, the rotor's resistance value is supposed to double. The motor's speed makes a maximum drop of 3 rpm (0.3 %) and then returns inside a 0.2 % error strip after 0.045 s.

Figure 9b shows the neural controller reaction to the variation of the moment of inertia. The motor's speed is simulated, under no load, for moments of inertia equal to J and $J \times 2$. Simulation results show that multiplying J

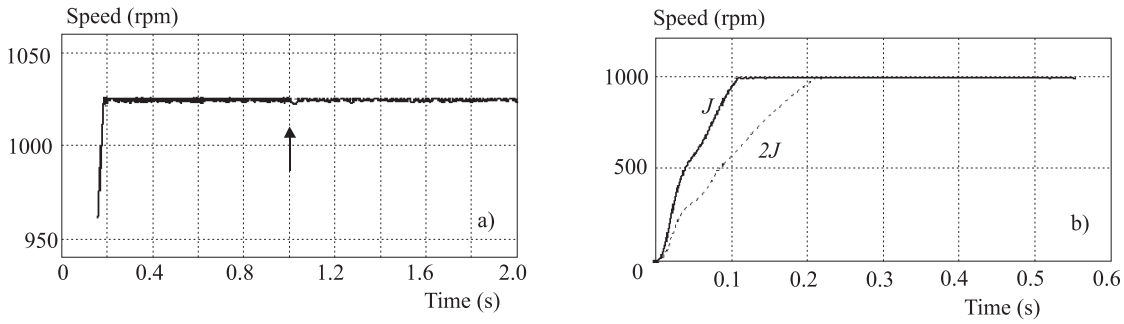

Fig. 9. Hysteresis PI based Neural Controller reaction to machine parameters variation.

by 2 affects only the speed settling time which changes from 0.125 s to 0.235 s.

7 CONCLUSION

An original hysteresis PI speed controller for voltage reference IRFOC induction machine drive control has been presented. This controller generates appropriate stator current distortion in order to obtain high performance induction motor speed control.

A dynamic performance comparison with the classical PI controller showed that a simple hysteresis has changed the classical PI controller to a high performance controller. The only drawback is that the hysteresis PI controller cannot deal with important down step speed tracking.

The input output relationship of the proposed controller has been used to design an artificial neural network based controller whose generalization capacity got rid of the down step tracking problem.

Simulation results show that the designed neural controller realizes a good dynamic behavior of the motor, with a rapid settling time, no overshoot, almost instantaneous rejection of load disturbance, a perfect speed tracking and it deals well with parameter variations of the motor. It seems to be a high- performance robust controller.

References

[1] LEONHARD, W.: Control of Electrical Drives, Springer-Verlag, 1985.

[2] NARENDA, K. S.—PARTHSARATHY, K.: Identification and Control of Dynamical Systems using Neural Networks, IEEE Trans, Neur netw **1** (March 1990), 4–27.

[3] FILIPPETTI, F.—FRANCESCHINI, G.—TASSONI, C.: Neural Network Aided Online Diagnostics of Induction Motor Rotor Faults, IEEE Trans. Ind. Applicat **3** No. 4 (June 1995), 892-899.

[4] BA-RAZZOUK—CHERITI, A.—OLIVIER, G.: Artificial Neural Networks Rotor Time Constant Adaptation Indirect Field Oriented Control Drives, IEEE PESC, Baveno (June 1986), 701-707.

[5] BUH, M. R.—LORENZ, R. D.: Design and Implementation of Neural Network for Digital Current Regulation of Inverter Drives, Conf. Record of IEEE, IAS Annual Meeting, pp. 415-421 1991.

[6] HO, E. Y. Y.—SEN, P. C.: Decoupling Control of Induction Motor Drives, IEEE Trans. on Ind. Electron. **35** No. 2 (May 1988), 253-262.

[7] BOSE, B. K.: Technology Trends in Microcomputer Control of Electrical Machines, IEEE Trans. Ind. Electron. **35** No. 1 (Feb 1988), 160-177.

[8] LEONHARD, W.: Adjustablespeed AC Drives, Proc. of IEEE, **76** No. 4 (April 1988), 455-471.

[9] PSALTI, D.—SIDERIS, A.—YAMAMURA, A.: Neural Controllers, Proceedings of 1st International Conference on Neural Networks, vol. 4, San Diego USA, pp. 551-558, 1987.

[10] JANG, J.-S. R.—SUN, C.-T.: Neuro-Fuzzy Modeling and Control, Proc. IEEE **83** (1995), 378-406.

[11] KAZMIERKOWSKI, M. P.—KASPROWICZ, A.: Improved Direct Torque and Flux Vector Control of PWM Inverter-Fed Induction Motor Drives, IEEE Trans. Ind. Electron. **45** (Aug 95), 344-350.

[12] NASH, J. N.: Direct Torque Control, Induction Motor Vector Control without an Encoder, IEEE Trans. Ind. Applicat. **33** (Mar/Apr 1997), 333-341.

[13] DAMIANO, A.—VAS, P. *et al*: Comparison of Speed-Sensorless DTC Induction Motor Drives, in Proc. PCIM, Nuremberg, Germany, 1997, pp. 1-11.

[14] BUJA, G.: A New Control Strategy of the Induction Motor Drives: The Direct Flux and Torque Control, IEEE Ind. Electron. Soc. Newslett. **45** (Dec 1998), 14-16.

[15] VAS, P.: Sensorless Vector And Direct Torque Control, Oxford Univ. Press, Oxford, U.K., 1998.

[16] CASADEI, D.—SERRA, G.—TANI, A.: IEEE Trans. Power Electron..

[17] GRABOWSKI, P. Z.—KAZMIERKOWSKI, M. P.—BOSE, B. K.—BLAABJERG, F.: A Simple Direct Torque Neuro Fuzzy Control of PWM Inverter Fed Induction Motor Drive, IEEE Trans. Ind. Electron. **47** No. 4 (Aug 2000), 863–870.

Received 17 October 2005
Revised 22 May 2006

Abdallah Miloudi was born in Saida, Algeria in October 1958. He received the BSc in 1981 in Mechanical Engineering, MSc in 1983 in Control Engineering all from Bradford University, England, and his PhD in 2006 from USTO, in Algeria. His main research interests are in the field of analysis, and intelligent control of electrical drives, and DSP programming. He is now a senior lecturer in the department of Electrical Engineering in Saida University, Algeria.

Eid Al-radadi was born in Madinah, Saudi Arabia in 1967. He received the BEng degree from Detroit Mercy University, USA in 1994, MSc and PhD degrees from Wayne State University, USA in 1998, 2002 respectively al in Electrical Engineering. He is now a senior lecturer and Dean of Madinah College of Technology.

Azeddine Draou was born in Maghnia, Algeria in 1955. He received the BEng degree from Sheffield University, UK in 1980, the MSc degree from Aston University in Birmingham, UK in 1981, and the PhD degree from Tokyo Institute of Technology, Japan in 1994 all in Electrical Engineering. His main area of research includes power electronics, static VAR compensation, multilevel inverters, and intelligent control of AC drives, UPFC and FACTS devices. He has published over 130 papers in technical journals and conference proceedings. Dr. Draou is a senior member of IEEE/PES, IES, IAS societies. He is now a senior lecturer and head of research and development centre at Madinah College of Technology.