MULTI-BASIS WAVENET-BASED SPEED ESTIMATION IN DIRECT TORQUE CONTROLLED ASYNCHRONOUS MOTOR

Adel A. OBED
Electrical Department, College of Engineering, Basrah University.

ABSTRACT:

This paper presents a proposed method for speed estimation of asynchronous motor in Direct Torque Control (DTC) system, based on a new architecture of multi-basis wavenet model. Such multi-basis model utilizes multi-set daughter wavelets. Firstly, the structure and training algorithm of the proposed method is discussed. The descent gradient method is used to fulfill both system structure and parameters initialization. Secondly, the proposed speed estimator and the DTC asynchronous motor are combined based on stator current signal and the motor speed is then estimated online with the operation of the system. Finally, the effectiveness of this method is proved by simulation carried out using Matlab/Simulink library and compared with the actual results obtained from the dynamic equations of the motor. The simulation results are obtained over the entire speed of starting, load conditions and motor braking. These results show that the proposed method is effective for speed estimation in DTC drives.

KEYWORDS: Wavenet, Multi-basis wavenet, Direct torque control, Speed estimation, Asynchronous motor.

تقدير السرعة في نظام التحكم المباشر لعزم المحرك الغير متزامن باعتماد الشبكات العصبية المويجية متعددة الأساسات

الخلاصة :

في هذا البحث تم اقتراح طريقة لتقدير سرعة المحرك الغير متزامن في نظام التحكم المباشر للعزم باعتماد الشبكات العصبية المويجية المتعددة الأساسات بتوظيف عدة مجاميع من بنات المويجات. تمت في البداية مناقشة تركيب وخطوات تدريب النظام المقترح باستخدام خوارزمية الانحدار الهابط لتحقيق بنية النظام وحساب المعاملات. وتم ثانيا تركيب وربط نظام تقدير السرعة المقترح مع نظام التحكم المباشر للعزم عبر تيار الجزء الثابت للمحرك حيث تم تقدير السرعة بشكل مباشر أثناء عمل النظام. وأخيرا تم التحقق من كفاءة الطريقة المقترحة لتقدير السرعة من خلال نتائج المحاكاة التي انجزت باستخدام مكتبة Simulink في Matlab وقورنت نتائج السرعة المقدرة مع السرعة الحقيقية المحسوبة من معادلات الحركة للمحرك. أستحصلت نتائج المحاكاة لعدة مديات للسرعة عند البدء وحالات التحميل وعند حالة إيقاف المحرك. وقد أظهرت النتائج فعالية الطريقة المقترحة قي تقدير السرعة في أنظمة التحكم المباشر للعزم.

INTRODUCTION

The speed estimation in DTC asynchronous motor systems have developed during the last few years and become the main development direction of the AC speed adjustment [C. J. Chen, et al., 2007]. Generally to establish the speed loop feedback in DTC system, information of rotor speed is essential. Installation sensor for detecting rotor speed will increase hardware cost and add some volume of system as well as the installation and maintenance bring many difficulties beside the mechanical error which affect the detection precision and performance of DTC control.

Several methods that eliminate the speed sensor have been developed. The observer-based [J. Maes, et al., 2000, H. Kubota, et al., 1993, G. Yang, et al., 1993, C. Lascu, et al., 2004, and Y. R. Kim, et al.,, 1992] and the Model Reference Adaptive System (MRAS) [C. Schauder, 1992, C. Lascu, et al., 2000. and T. BANA, et al., 2002] seem the most used methods for sensorless speed DTC system. However, the estimated speed is affected by machine parameters. Moreover, these methods need to detect all the terminal voltages and currents of the motor. A good technique for speed sensorless operation based on Neural Network (NN) has been presented over the last years [M. P. Kazmiorkowski, et al., 1997, and L. Brahim, et al., 1993]. This method is based only on signals detected from motor terminals which eliminate the effect of motor parameters sensitivity on the estimated speed. However, the estimated speed could not be perfectible due to some problems such as trapping in to local minima and slow convergence.

The combination of NN with wavelet transform (wavenet), behaves good localization property in both time and frequency space and multi-scale property. It is used for the analysis of non-stationary signals and learning of the nonlinear functions [S. Mallat, 1999]. This technique is also proposed to estimate the speed in DTC system [C. Zhi, et al., 2003 and M. A Alwan, et al., 2008].

In this paper, the theory of wavenet is presented and a method of speed estimation based on multi-basis wavenet is proposed. The proposed method includes multi-set daughter wavelets from different mother functions in the hidden layer which is believed useful to represent functions containing different signal cutting, ripples and rapid signal changes. The simulink model of DTC asynchronous motor system is implemented and a speed estimator based on the above proposed method is combined with the system based on one line of stator current signal. Simulation results include developed torque, stator current beside the rotor speed which is compared with the actual motor speed. Note that the theory of DTC control has not been discussed here, however it can be found in details in [G. S. Buja, et al., 2004].

WAVENET

Wavenet can be considered a particular case of the feed forward basis function neural network model. In ordinary network, several types of basis functions, such as radial basis functions, splines and polynomial functions of synapse neurons are used instead of sigmodial function. The connection weights are taken to represent the corresponding coefficients. The output layer performs the sum of the output of all synapse neurons. Since wavelets have been shown their excellent performance in non stationary signal analysis and nonlinear function modeling, the neural network using wavelet basis function, wavenet, provides higher availability of rate of convergence for the approximation than an ordinary feed forward neural network [S. Mallat, 1999].

The wavenet can be constructed by means of replacing the nonlinear sigmodial function with nonlinear wavelet basis function. Figure 1 illustrates the wavenet structure and Fig. 2 shows some typical wavelet functions.

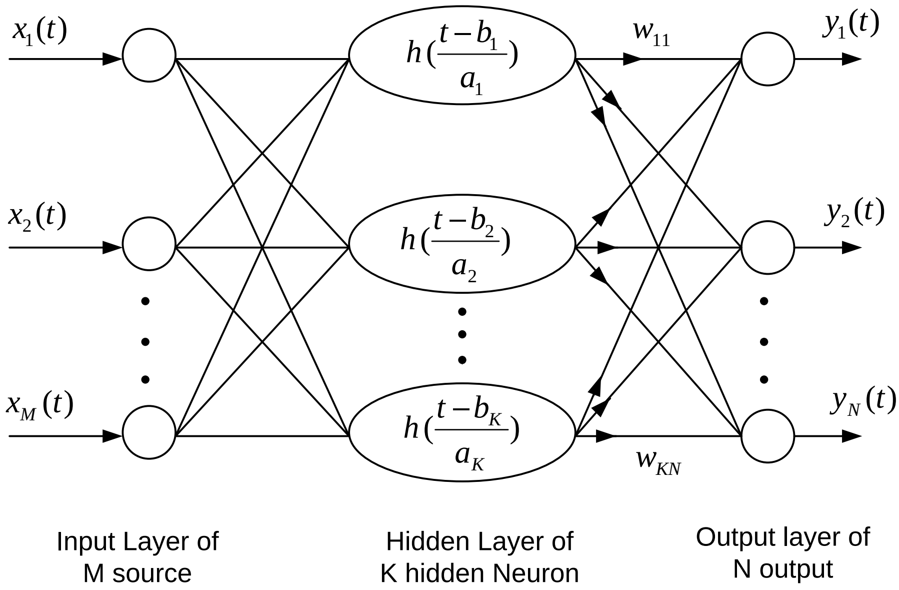

Fig.(1) The structure of wavenet

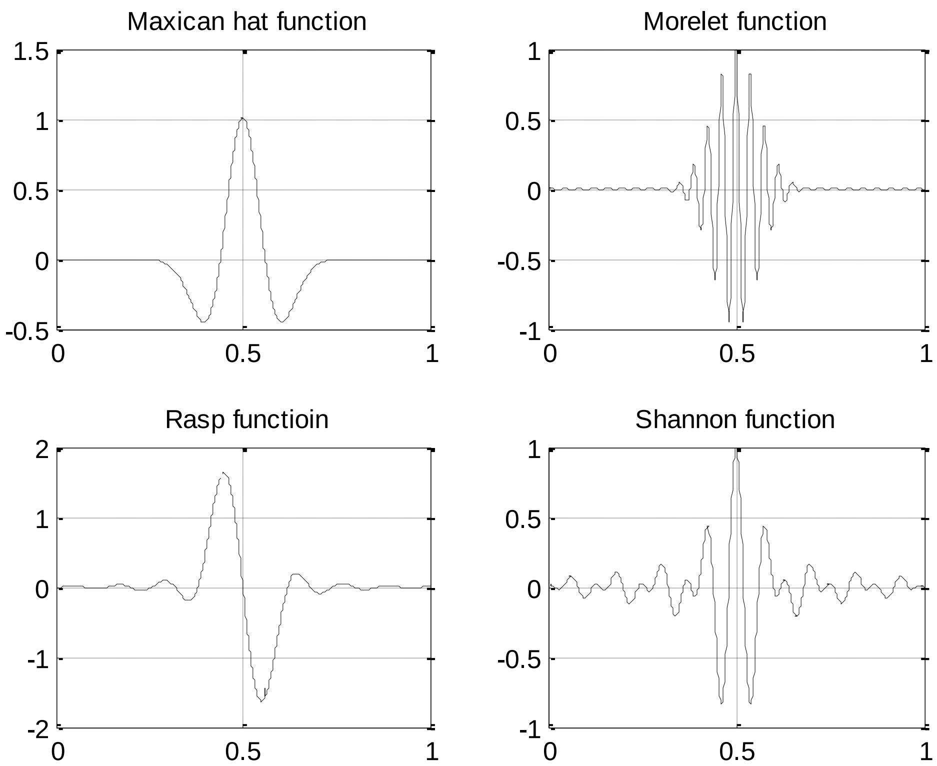

Fig. 2 Some typical wavelet functions

The network exhibits a multi-input to multi-output nonlinear system, realizing mapping from $R^m \Rightarrow R^n$. The approximated output signal of the network can be expressed as follows

$$y_i = f\left[\sum_{k=1}^{K} w_{ki} \sum_{m=1}^{M} x_m(t)\ h_m((t-b_k)/a_k)\right] \tag{1}$$

where $x_m (m = 1,2,...,M)$ is the input for the m-th training vector $X(t)$, $y_i (i = 1,2,..., N)$ is the output for the i-th training vector $Y(t)$, M is the node numbers of the input layer, K is the node numbers of hidden layer, w_{ki} is the weight between the k-th node of the hidden layer and the i-th node of the output layer, $h(t)$ is the mother wavelet, a, b are the dilation and translation and f is the linear function.

PROPOSED WAVENET SPEED ESTIMATOR

The structure of the proposed P-based wavenet speed estimator is shown in Fig. 3 and the set up of its training and estimation for online speed changes in DTC asynchronous motor system is shown in Fig. 4.

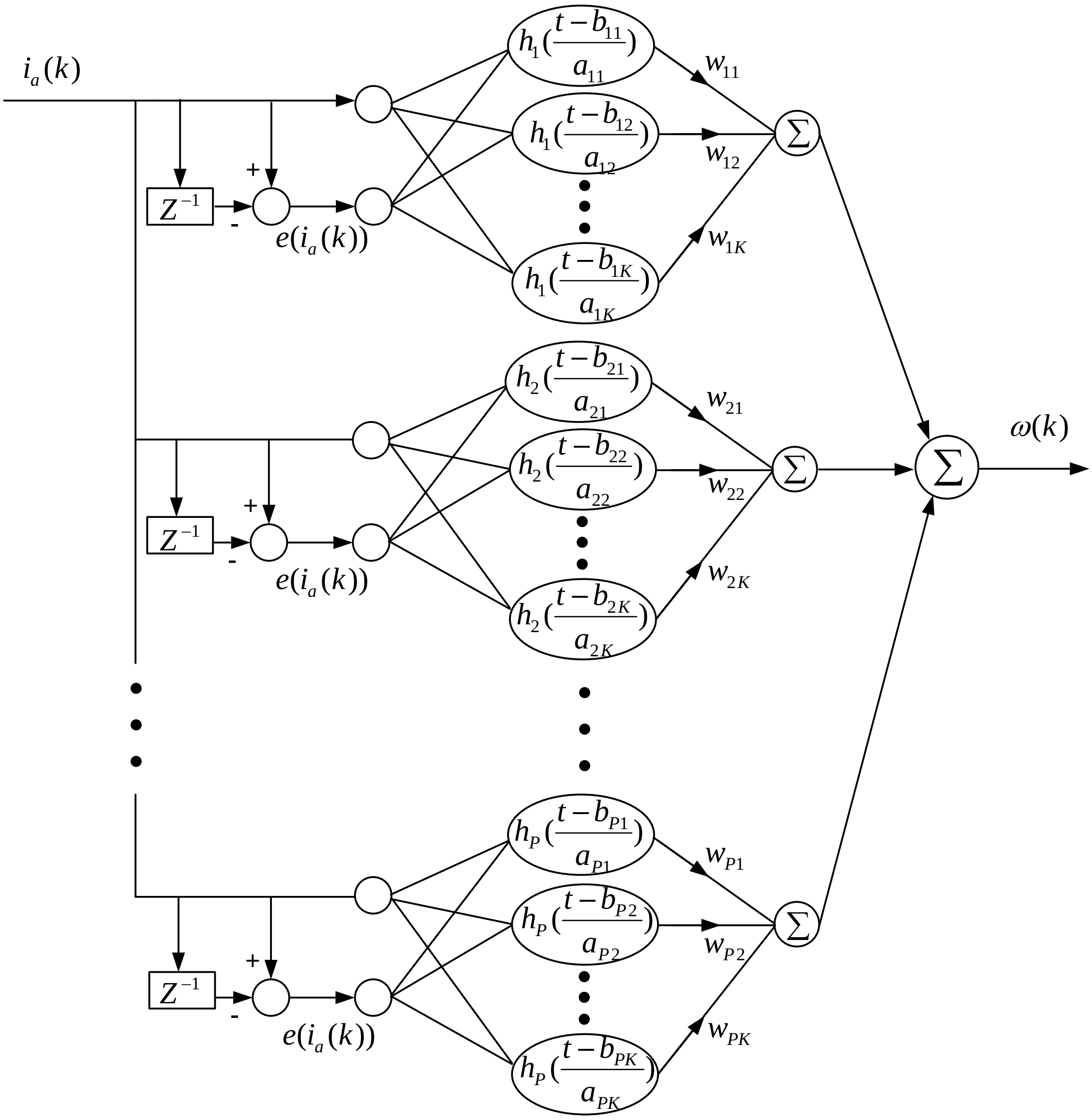

Fig.3 The proposed 2-input/ single output P-based (each of K daughters) wavenet speed estimator.

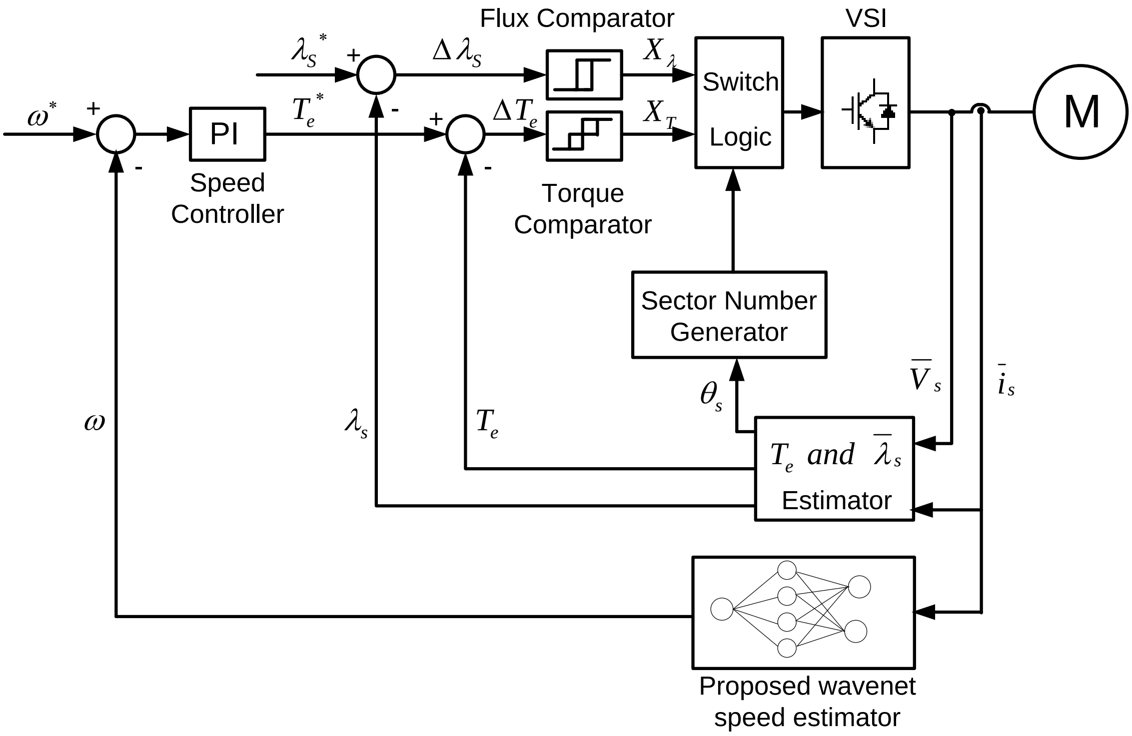

Fig.4 The set up for online training and estimation for speed in DTC asynchronous motor system.

The wavenet speed estimator exhibits two inputs and single output. The input nodes are the stator current $i_a(k)$ and the error in the stator current $e(i_a(k))$. They are defined as follows
$i_a(k)$ = instantaneous value of stator current.

$$e(i_a(k)) = i_a(k) - i_a(k-1) \tag{2}$$

A single signal from one line current is used here which uses a single current sensor. This is useful for simplicity and economy [M. A. Alwan, et al., 2008]. The hidden layer is of P different sets of daughter wavelets. The output of the wavenet is the motor speed which can be given as follows [C. Zhi, et al., 2003]

$$y = \omega(k) = f\left[\sum_{p=1}^{P}\sum_{k=1}^{K} w_{pk}\sum_{m=1}^{M} X_m\, h_p((t-b_{pk})/a_{pk})\right] \tag{3}$$

In order to determine the adjustable weights $w_{kp}(k=1,2,...K,\ \ p=1,2,...P)$ and the adjustable parameters a_{pk} and b_{pk}, a least mean square (LMS) energy minimizing function can be applied:

$$E = \frac{1}{2}\sum_{l=1}^{q}\sum_{i=1}^{n}(e^l)^2 \tag{4}$$

where $e^l = F^l - y^l$, q and F^l are the number of training samples and the desired value of y^l. To minimize the energy error E, a method of steepest descent which requires the gradients $\frac{\partial E}{\partial w_{pk}}, \frac{\partial E}{\partial a_{pk}}$ *and* $\frac{\partial E}{\partial b_{pk}}$ is used for updating the incremental changes to each parameter $w_{pk},\ a_{pk},\ and\ b_{pk}$. The gradients of E are:

$$\frac{\partial E}{\partial w_{pk}} = -\sum_{l=1}^{q} \sum_{i=1}^{N} \sum_{m=1}^{M} e^l \ X_m{}^l \ h(\tau) \tag{5}$$

$$\frac{\partial E}{\partial b_{pk}} = -\sum_{l=1}^{q} \sum_{i=1}^{N} \sum_{m=1}^{M} e^l \ X_m{}^l * \ w_{pk} \frac{\partial h(\tau)}{\partial b_{pk}} \tag{6}$$

$$\frac{\partial E}{\partial a_{pk}} = -\sum_{l=1}^{q} \sum_{i=1}^{N} \sum_{m=1}^{M} e^l \ X_m{}^l * w_{pk}\, \tau \frac{\partial h(\tau)}{\partial b_{pk}} = \tau \frac{\partial E}{\partial b_{pk}} \tag{7}$$

where $\tau = \frac{t - b_{pk}}{a_{pk}}$. The updated weight w_{pk} and the parameters a_{pk} and b_{pk} are:

$$w_{pk}(n+1) = w_{pk}(n) - b_w \frac{\partial E}{\partial w_{pk}} + c_w \Delta w_{pk}(n) \tag{8}$$

$$a_k(n+1) = a_k(n) - b_a \frac{\partial E}{\partial a_k} + c_a \Delta a_k(n) \tag{9}$$

$$b_k(n+1) = b_k(n) - b_b \frac{\partial E}{\partial b_k} + c_b \Delta b_k(n) \tag{10}$$

where b_w, b_a, and b_b are steps size, c_w, c_a, and c_b are the forgetting factors which are variable factors and can greatly reduce the number of iterations for convergence.

SIMULATION RESULTS

Two sets of daughter wavelet functions $(P = 2)$ with seven neurons $(K = 7)$ in each set are used to represent the hidden layer of a 2-basis wavenet speed estimator. The two mother wavelets used in these sets are Mexican hat and Shannon functions. These functions with their derivative with respect to the translation b are given as follows

Mexican hat function

$$h(\tau) = \frac{2}{\sqrt{3}} \pi^{\frac{1}{4}} (1 - \tau^2)\, e^{-\frac{\tau^2}{2}} \tag{11}$$

$$\frac{\partial h(\tau)}{\partial b} = \frac{1}{a} (3\tau - \tau^3) \exp(-\frac{\tau^2}{2}) \tag{12}$$

and for Shannon function

$$h(\tau) = \frac{\sin 2\pi\tau - \sin \pi\tau}{\pi\tau} \tag{13}$$

$$\frac{\partial h(\tau)}{\partial b} = \frac{\pi}{a} \frac{(-\pi\tau \cos \pi\tau - 2\pi \cos 2\pi\tau + \sin \pi\tau + \sin 2\pi\tau)}{(\pi\tau)^2} \tag{14}$$

where $\tau = \dfrac{t - b_{pk}}{a_{pk}}$. The wavenet speed estimator was trained off line before combining it with the DTC system based on data obtained from the conventional operation of the DTC system. The training is based on eq. 2 to 10 which can programmed in M-file in Matlab library as given in the Appendix. The simulation was carried out to verify the function of the speed estimator where the motor used in this system is a 3-ph, 1250 hp, 4160V, 150A, 6 poles, and 60 Hz induction motor (asynchronous motor). The simulation parameters of the motor are described as follows: N_s =1200r.p.m., R_s =0.21 Ω, $R_r = 0.146\Omega$, $L_s = L_r = 2mH$, $L_m = 0.155H$ and $J = 22kg.m^2$.

The rotor speed of the motor which is used to complete the torque loop in the DTC system (Fig. 4) is taken either from the actual value calculated from the dynamic motor equations or from the proposed wavenet speed estimator.

Figure 5 shows the actual rotor speed for starting the DTC system with no load from standstill to a speed of 100 rad./sec. through 2 seconds, then the speed is increased suddenly to 200 rad./sec. through the next 2 seconds. Figure 6 shows the estimated rotor speed identified by the proposed wavenet speed estimator for the case above. It shows higher dynamical following performance. The starting current and the starting torque for the same previous case are show in Fig. 7 and 8 respectively.

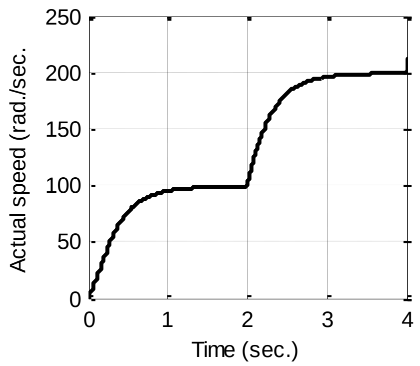

Fig. 5 Actual speed (electrical) during starting

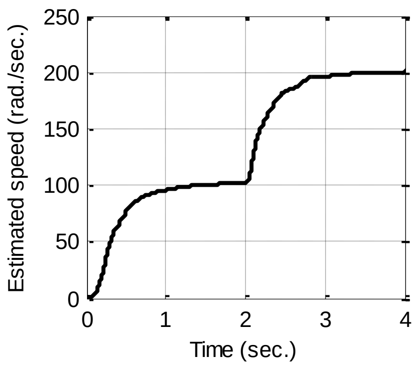

Fig. 6 Estimated speed (electrical) during starting

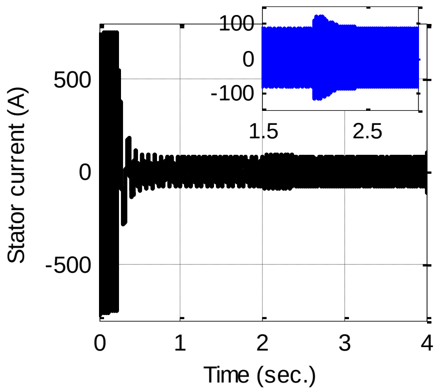

Fig. 7 Stator current during starting with magnified portion at speed step change instant.

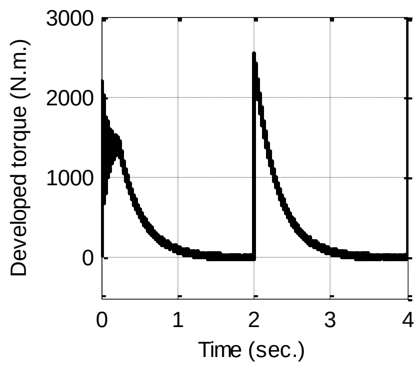

Fig. 8 Developed torque during starting.

ROBUSTNESS

In order to test the robustness of the proposed method, the effect of system braking and sudden change for full load torque have studied on the performance of the torque control and speed changes.

To illustrate the performance of braking control, the operating mode have simulated without load and the motor runs at rated speed of 377.8 rad./sec., the braking starts at the instant of 3 sec. and the system is kept at standstill after the braking. Figure 9 and 10 show the actual and estimated speed during braking while Fig. 11 and 12 show the stator current and developed torque for the same mentioned case. The estimated speed follows the actual with an error less than 1% which reflect the ability of the proposed speed estimator.

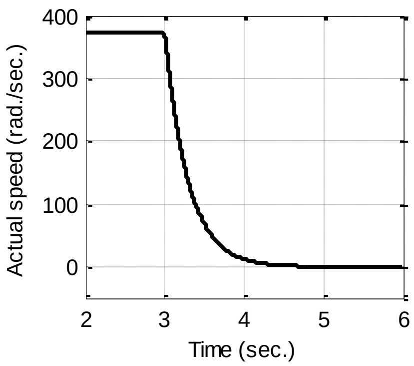

Fig. 9 Actual speed (electrical) during braking

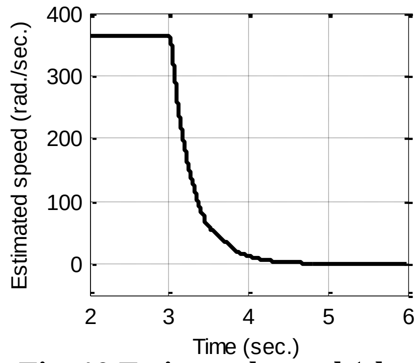

Fig. 10 Estimated speed (electrical) during braking

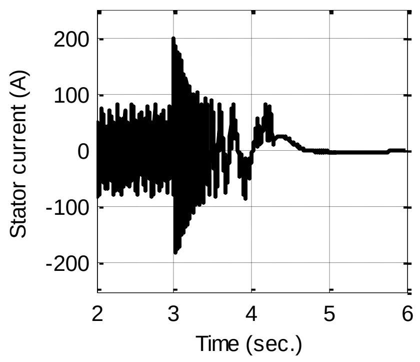

Fig. 11 Stator current during braking.

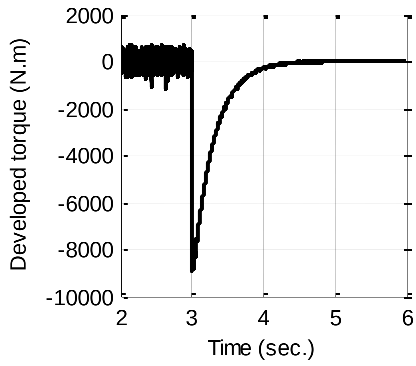

Fig. 12 Developed torque during braking.

Figures 13-15 show the tests of robustness realized with sudden change in load torque from no load to full load of the motor used, 7490 N.m, at the instant t=4sec. and its elimination at t=6 sec. This process is repeated two times. In this case, the estimated speed is used as a feed back signal to complete the torque loop of the DTC system. The developed torque curve in Fig. 15 reflects the direct control of torque. The developed torque follows its demand quickly with a time less than 0.05 sec.

For the robustness of control, a braking case, the estimated speed follows its actual and for system loading when the speed is used as feed back signal, the DTC system gives an adequate accuracy of system performance.

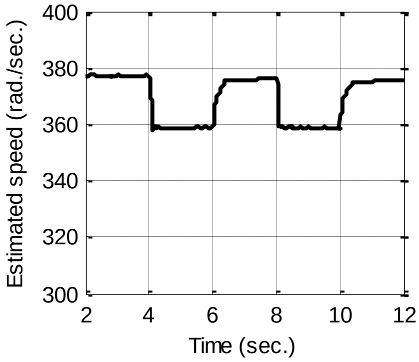

Fig. 13 Estimated speed (electrical) during load change.

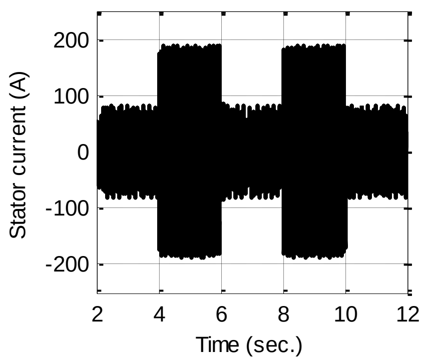

Fig. 14 Stator current during load change

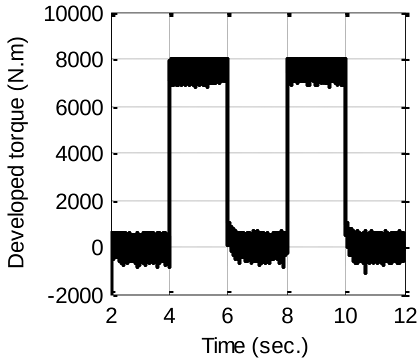

Fig. 15 Developed torque during load change

CONCLUSIONS

The paper presents a new multi-basis wavenet speed estimator based on multi-set daughter wavelet functions in DTC asynchronous motor system. The multi-basis wavenet is useful because of high nonlinearity in the stator current waveform. The wavenet speed estimator is trained offline based on stator current information obtained from the conventional operation of the DTC system. The advantage of using multi-basis wavenet is the adequate speed estimated for different operating conditions. The simulation results reveals a good system performance.

REFERENCES

- C. J. Chen and T. C. Chen, "Speed Sensorless of an Induction Motor Using Self-Tuning Fuzzy Identification", 2nd International Conference on Information and Control ICICIC, 5-7 Sep. 2007.

- C. Lascu and A. M. Trzynadlowski, "A Sensorless Hybrid DTC Drive for High Volume Low-Cost Application", IEEE Trans. Ind. Electron., Vol. 51, No. 5, pp. 1048-1055, Oct. 2004.

- C. Lascu, and F. Blaabjerg, "A Modified Direct Torque Control for Induction Motor Sensorless Drive", IEEE Trans. Ind. Applicat., Vol.36, No. 1, Jan./Feb., 2000.

- C. Schauder, "Adaptive Speed Identification for Vector Control of Induction Motors without rotational transducers", IEEE Trans. Ind. Applicat., Vol. 28, No. 5, pp. 1054-1061, Sept./Oct. 1992.

-C. Zhi Cao, Mu_ping Lu, and Xin Wang, "Speed Estimation and Stimulation of DTC System Based on Wavelet Neural Network", Proc. On 2nd International on Machine Learning and Cybernetics, Xi'an, 2-5 Nov. 2003.

- G. S. Buja and M. P. Kazmierkowski, "Direct Torque Control of PWM Inverter-Fed AC Motors-A Survey", IEEE Trans. Ind. Electron., Vol. 51, No. 4, pp. 744-757, Aug. 2004.

- G. Yang and T.-H. Chin, "Adaptive-Speed Identification Scheme for a Vector-Controlled speed Sensorless Inverter Induction Drive." IEEE, Trans. Ind. Applicat, Vol. 29, No. 4, pp 820-825, July/Aug. 1993.

- H. Kubota, K. Matsuse, and T. Nakano, "DSP-Based Speed Adaptive Flux Observer of Induction Motor", IEEE Trans. Ind. Applicat., Vol. 29, No. 2, pp. 344-348, Mar./Apr. 1993.

- J. Maes and A. Melkebeek, "Speed Sensorless Direct Torque Control of Induction Motors Using Adaptive Flux Observer", IEEE Trans. Ind. Applicat., Vol., 36, No. 3, pp. 778-785, May/June 2000.

- L. Brahim and R. Kurosawa, "Identification of Induction Motor Speed Using Neural Networks", in PCC-Yokohama, pp. 689-694, 1993.

- M. A. Alwan, J. M. Abdul-Jabbar and A. A. Obed, " Speed Estimation of DTC Induction Motor Using Single Current Sensor Based on Wavenet Theory", Basrah Journal of Engineering Sciences, Vol. 8, No. 1, pp 27-38, 2008.

- M. P. Kazmiorkowski, D. L. Sobczuk and Filipek, "Sensorless Control of Induction Motor Using a Neural Network for Speed Estimation", Industrial Electronics, Vol. 3, PP. 1242-1246, July, 1997.

- S. Mallat, "A Wavelet Tour of Signal Processing", Academic Press, 1999.

- T. PANA, C. RUSU, "Speed and Rotor Flux Estimation in Speed Sensorless Control of Induction Motor", Ion BIVOL Fascicle III, 2002.

- Y. R. Kim, S.-K. Sul, and M.-H. Park, " Speed Sensorless Vector Control of an Induction Motor Using an Extended Kalman Filter", in conf. Rec. IEEE-IAS Annu. Meeting, pp. 594-599 1992.

LIST OF SYMBOLES AND ABBREVIATIONS

A.C: Alternating Current.
DTC: Direct Torque Control.
LMS: Least Mean Square.
MRAS: Model Reference Adaptive System.
a : Dilation
b : Translation.
c : Forgetting factor.
$e(i_a)$: error in stator current (A).
E : Energy error.
f : Linear function.
F : Desired vector of the output vector.
h : Mother wavelet.
i_a : Stator current (A).
J : Moment of inertia ($kg.m^2$).
K : Node numbers of hidden layer.
L : Inductance (H).
M : Node numbers of the input layer.
N : Node numbers of the output layer, Motor speed (rad./sec.).
P : Number of daughter wavelets.
q : Number of training samples.
R : Resistance Ω.
w : Weight.
x : Input vector.
y : Output vector.

APPENDIX

```
%Matlab program for learning the proposed wavenet speed estimator based on eq.2 to10.
%Mexican Hat and Shannon mother wavelet functions are used in learning
w1=[.1,.1,.1,.1,.1,.1,.1];a1=[.1,.1,.1,.1,.1,.1,.1];b1=[-.1,0,.1,.2,.5,.8,1.2];
w2=[.1,.1,.1,.1,.1,.1,.1];a2=[.1,.1,.1,.1,.1,.1,.1];b2=[-.1,.1,.3,.5,.9,1,1.3];
DW1=[0,0,0,0,0,0,0]; DB1=[0,0,0,0,0,0,0]; DA1=[0,0,0,0,0,0,0];
DW2=[0,0,0,0,0,0,0]; DB2=[0,0,0,0,0,0,0]; DA2=[0,0,0,0,0,0,0];
t=[0:1700];% time sample 1701samples
x11=[];% input current, can be loaded from file,1701 samples
F1=[];% target speed, can be loaded from file, 1701 samples
x=(x11-min(x11))/(max(x11)-min(min(x11));%  normalized input current
F=(F1-min(F1))/(max(F1)-min(F1));%normalized target
aw1=.993;aa1=.993;ab1=.993;bw1=.0001;ba1=.0001;bb1=.0001;
aw2=.1;aa2=.1;ab2=.1;bw2=.01;ba2=.01;bb2=.01;
for I=[1:1000]%iteration counter
    E=0;
    for n=[1:1701];%sampled number
        y(n)=0;
        for k=[1:7];%weighs counter
            y1=w1(k)*(x(n))*(2/sqrt(3))*(pi^(-1/4))*(1-(((x(n)-b1(k))/a1(k))^2))*(exp(-(((x(n)-
b1(k))/a1(k))^2)/2));
            y2=w2(k)*(x(n))*(sin(2*pi*((x(n)-b2(k))/a2(k)))-sin((x(n)-b2(k))/a2(k)))/(pi*(x(n)-
b2(k))/a2(k));
            y(n)=y(n)+y1+y2;
        end; %end k
        E1=0.5*((F(n)-y(n))^2); E=E+E1;
    end %end n
     if E< abs(.01)
        disp('end of work')
        break;
     else
         for k=[1:7
          PEPW1(k)=0;PEPB1(k)=0;PEPA1(k)=0;PEPW2(k)=0;PEPB2(k)=0;PEPA2(k)=0;
           for n=[1:1701];
             t1=(x(n)-b1(k))/a1(k);
             PEPW11(k)=-(F(n)-y(n))*(x(n))*(2/sqrt(3))*(pi^(-1/4))*(1-(((x(n)-
b1(k))/a1(k))^2))*(exp(-(((x(n)-b1(k))/a1(k))^2)/2));
             PEPB11(k)=-(F(n)-y(n))*x(n)*w1(k)*(2/sqrt(3))*(pi^(-1/4))*(1/a1(k))*(3*t1-
t1^3)*exp(-(t1^2)/2);
             PEPA11(k)=PEPB1(k)*t1; PEPW1(k)=PEPW1(k)+PEPW11(k);
             PEPB1(k)=PEPB1(k)+PEPB11(k); PEPA1(k)=PEPA1(k)+PEPA11(k);
             t2=(x(n)-b2(k))/a2(k);
             PEPW22(k)=-(F(n)-y(n))*((x(n)-b2(k))/a2(k))/((((x(n)-b2(k))/a2(k))^2+1)^2);
             PEPB22(k)=-(F(n)-y(n))*x(n)*w2(k)*(1/a2(k))*((pi*t2*cos(pi*t2)-
2*pi*cos(2*pi*t2)+sin(pi*t2)+sin(2*pi*t2))/((pi*t2^2)));
             PEPA22(k)=PEPB2(k)*t2; PEPW2(k)=PEPW2(k)+PEPW22(k);
             PEPB2(k)=PEPB2(k)+PEPB22(k); PEPA2(k)=PEPA2(k)+PEPA22(k);
           end % end n
           storew1(k)=w1(k);
           w1(k)=w1(k)-bw1*PEPW1(k)+aw1*DW1(k);
```

```
        DW1(k)=w1(k)-storew1(k);
        storeb1(k)=b1(k);
        b1(k)=b1(k)-bb1*PEPB1(k)+ab1*DB1(k);
        DB1(k)=b1(k)-storeb1(k);
        storea1(k)=a1(k);
        a1(k)=a1(k)-ba1*PEPA1(k)+aa1*DA1(k);
        DA1(k)=a1(k)-storea1(k);
        storew2(k)=w2(k);
        w2(k)=w2(k)-bw2*PEPW2(k)+aw2*DW2(k);
        DW2(k)=w2(k)-storew2(k);
        storeb2(k)=b2(k);
        b2(k)=b2(k)-bb2*PEPB2(k)+ab2*DB2(k);
        DB2(k)=b2(k)-storeb2(k);
        storea2(k)=a2(k);
        a2(k)=a2(k)-ba2*PEPA2(k)+aa2*DA2(k);
        DA2(k)=a2(k)-storea2(k);
      end %end k
    end;%end if
end %end I
%Plot the training results
subplot(2,2,1)
plot(t,y,'k'); grid on
xlabel('time(sec)')
ylabel('output speed')
title('Fig.1.identified speed')
subplot(2,2,2)
plot(t,F,t,y); grid on
xlabel('time(sec)')
ylabel('target identified speed')
title('Fig.2.target and identified speed')
subplot(2,2,3)
plot(t,x11,'r'); grid on
xlabel('time(sec)')
ylabel('input1')
title('Fig.3.input')
subplot(2,2,4)
plot(t,E,'g')
grid on
xlabel('time(sec)')
ylabel('Error')
title('Fig.4.Error')
```