International Journal of Engineering, Science and Mathematics
Vol. 6 Issue 2, June 2017,
ISSN: 2320-0294 Impact Factor: 6.765
Journal Homepage: http://www.ijesm.co.in, Email: ijesmj@gmail.com

IMPROVING SPEED ACURACY OF A DC SERVOMOTOR USING MODEL RFERENCE ADAPTIVE CONTROL (MRAC) TECHNIQUE

Okafor, Patrick. U.
Department of Electrical and Electronic Engineering Department,
Faculty of Engineering, Enugu State University of Science and Technology (ESUT),
Enugu State, Nigeria.

Alor, Michael O.
Department of Electrical and Electronic Engineering Department,
Faculty of Engineering, Enugu State University of Science and Technology (ESUT),
Enugu State, Nigeria.

Eneh, Princewill Chigozie
Department of Electrical and Electronic Engineering Department,
Faculty of Engineering, Enugu State University of Science and Technology (ESUT),
Enugu State, Nigeria.

Abstract
The Artificial Neural Network Inverse Model (AIM) of the DC motor was designed as the plant in the Neural Model Reference Adaptive Control (MRAC) network. The concept is for the motor output to follow the reference input. An input to the Artificial Neural Network (ANN) controller is made from the DC motor with a corresponding desired or target response set at the output. Feed forward Back propagation training algorithm that propagates error was used in training the network. Error is computed from the difference between the desired response and the system output using the Least Mean Square Error (LMSE). The idea is to bring down the tracking error to zero or minimal acceptable value. Simulation was done at different sampling time and results show that by reducing the sampling time, the speed profile accuracy can be improved.

Keywords: Angular Displacement, Angular Velocity, Artificial Neural Network (ANN), DC Motor, Model Reference Adaptive Control (MRAC)

1.0 Introduction

Direct Current (DC) motors are one of the most widely used prime movers in industry today. Some years back, the most used servomotors for controlling purposes were Alternating Current(AC) motors. But in reality, AC motors are very difficult to control more especially in position control due to their nonlinear characteristics, which makes analytical task difficult (Kuo and Golnaraghi, 2002).DC motors are more expensive, because of their brushes and commutators, and variable-flux dc motors are suitable only for certain types of control

applications. Before the full introduction of permanent-magnet technology, the torque - per-unit volume or weight of a dc motor with a permanent-magnet (PM) field was far from desirable.

Today, with the development of the rare-earth magnet, it is possible to achieve very high torque-to-volume PM dc motors at reasonable cost. Also, the advances made in brush-and-commutator technology have made these wearable parts practically maintenance-free(Baldor Motors And Drives, 2009). The advancements made in power electronics have made brushless DC motors quite popular in high-performance control systems. Alsolow-time-constant properties have opened new applications for dc motors in computer peripheral equipment such as tape drives, printers, disk drives, and word processors, as well as in the automation and machine-tool industries.

One of the drawbacks of conventional tracking controllers for electric drives is that they are unable to capture the unknown load characteristics over a widely ranging operating point. Apparently, this makes tuning of controller parameters very difficult. There are many ways to overcome these difficulties but, generally there are four basic way that are common to adaptive controller; (1) Model reference adaptive control (MRAC), (2) Self tuning, (3) Dual control and (4) Gain scheduling. Usually load torque is a nonlinear function of a combination of variables such as speed and position of the rotor. Therefore, identifying the overall nonlinear system through a linearized model around a widely varying or changing operating point, under fast switching frequencies, can introduce errors which can lead to unstable or inaccurate performance of the system (Astron and Wittenmark, 1989).

2.0 Theory

Basically, the DC motor is a torque transducer that converts electric energy into mechanical energy. The torque developed on the motor shaft is directly proportional to the field flux and the armature current. Looking at fig.1, assume a current-carrying conductor is established in a magnetic field with fluxϕ, and the conductor is located at a distance r from the center of rotation (Zhao and Yu, 2011).

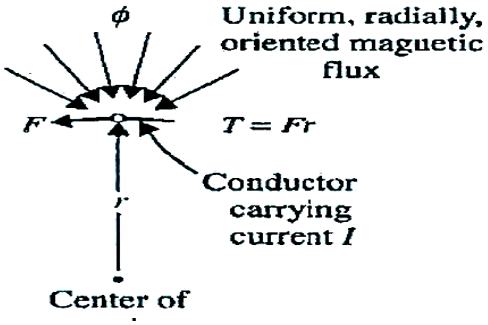

Fig.1: development of toque in the motor shaft.

The relationship among the developed torque, flux ϕ, and current i_a is given as:

$$T_m = k_m \phi i_a. \qquad (1)$$

Where;
T_m = the motor torque (in N-M)
ϕ, = the magnetic flux in (in webers)
i_a = the armature current (in amperes)
k_m = a proportional constant

In addition to the torque developed, when the conductor moves in the magnetic field, a voltage is generated across its terminals. This voltage is known as the *back emf***,** which is proportional to the shaft velocity, and tends to oppose the current flow. The relationship between the back emf and the shaft velocity is:

$$e_b = k_m \phi w_m. \qquad (2)$$

Where;
e_b = the back emf (in volts)
w_m = the shaft velocity of the motor (in rad/sec)
Equations 1 and 2 form the fundamentals of the dc-motor operation.

2.1 Artificial Neural Network Controller

An input to the ANN controller is made from the DC motor with a corresponding desired or target response set at the output. Feed forward Back propagation network that does not have feedback connections, but errors are propagated during training was used (Valluru, 1995). Least Mean Squared Error (LMSE) was used to calculate the error. Error is computed from the difference between the desired response and the system output. The Feed forward Back propagation ANN is shown in figure 3.

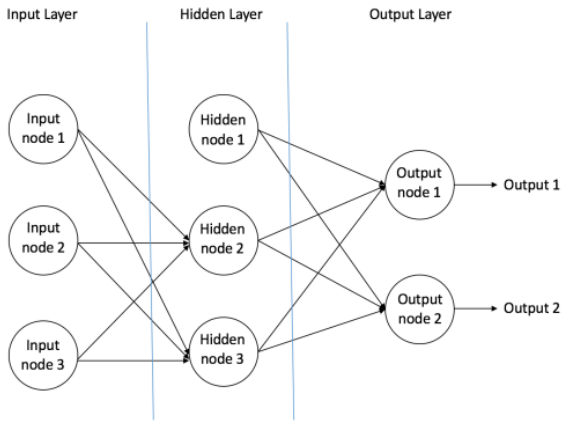

Figure 3: Feed forward Back propagation ANN

This error information is fed back to the system and adjusts the system parameters in a systematic fashion (the learning rule). The process is repeated until the performance is acceptable i.e. error is reduced to zero or minimal acceptable value. It is clear from this description that the performance rests heavily on the data. This operating approach should be contrasted with the

traditional engineering design, made of exhaustive subsystem specifications and intercommunication protocols.

2.2 The Model Reference Adaptive Control (MRAC)

MRAC is one of the neural network architectures for prediction and control, implemented in Neural Network Toolbox software. The model reference architecture requires that a separate neural network controller be trained offline, in addition to the neural network plant model. The controller training is computationally expensive, because it requires the use of dynamic backpropagation. The MRAC architecture uses two neural networks: a controller network and a plant model network. The plant model is identified first, and then the controller is trained so that the plant output follows the reference model output (Beale, Hagan and Demuth, 2015). The block diagram of an MRAC system is shown in figure 2.

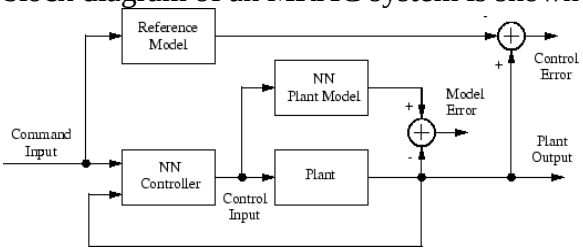

Figure 2: Block Diagram of MRAC System

3.0 Modeling the PM DC motor

DC motors are used extensively in control systems especially in industrial actuators so, it is paramount to establish mathematical model for analytical purposes for efficient control application of dc motors. The DC motor takes in single input in the form of an input voltage and generates a single output parameter in the form of output speed. It is a single-input, single-output system (SISO). Fig 4 shows the electrical representation of a DC motor.

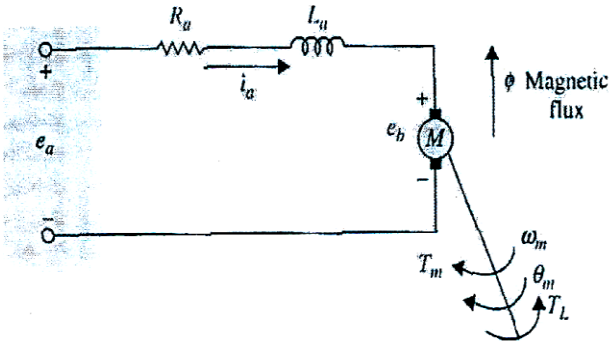

Fig 4: Electrical Model of DC Motor

The armature is modeled as a circuit with resistance R_a connected in series with an inductance L_a , and a voltage source e_b representing the Back Electromotive Force (EMF) in the armature when the rotor rotates. Looking at the diagram of fig 4, it can be seen that the control of the dc motor is applied at the armature terminals in the form of applied voltage e_a (t). It can be deduced that the

torque developed in the motor is proportional to the air-gap flux and the armature current. The equations that describes the DC servomotor behavior is as stated (Kuo and Golnaraghi (2002) thus,

$T_m(t) = K_m(t)\,\phi\, i_a(t)$ (3)
Since ϕ is constant, Equation 3 is in form
$T_m(t) = K_i i_a(t)$
$K_i i_a(t) = i_m \omega_m + b\omega_m + T_L$(4)
Where:
$T_m(t)$ =motor torque $i_a(t)$ = armature current $T_L(t)$ = load torque
ϕ = magnetic flux in the air gapk_m = proportionality constantK_i = torque constant in N-m/A.
$\omega_m(t)$=rotor angular velocityi_m = equivalent moment of inertia reflected at the motor shaft

Putting the control input voltage e_a (t) into consideration, the cause and effect equations for the motor circuit in same fig 4 are:

$$\frac{di_a(t)}{dt} = \frac{1}{L_a} e_a(t) - \frac{R_a}{L_a} i_a(t) - \frac{1}{L_a} e_b(t) \quad (5)$$

$$e_b(t) = k_b \frac{d\theta_m(t)}{dt} = K_b \omega_m(t) \quad (6)$$

$$\frac{d^2\theta_m(t)}{dt^2} = \frac{1}{J_m} T_m(t) - \frac{1}{J_m} T_L(t) - \frac{B_m}{J_m} \frac{d\theta_m(t)}{dt} \quad (7)$$

Where:
$l_a(t)$ = armature inductance
R_a = armature resistance
$e_a(t)$ = applied voltage
$e_b(t)$ = back emf
K_b = back emf constant
$\omega_m(t)$ = rotor angular velocityB_m = viscous-friction coefficient
θ_m (t) = rotor displacement
J_m =rotor inertia
From equations 3 through 6, the applied voltage $e_a(t)$ is considered as the cause and Equation 5 considers that $\frac{di_a(t)}{dt}$ the immediate effect due to the applied voltage. From Equation 3, armature current $i_a(t)$ causes the motor torque $T_m(t)$, while in Equation 6 the back emf$e_b(t)$ was defined. It can be seen also from Equation 7 that the motor torque produced causes the angular velocity $\omega_m(t)$ and displacement θ_m (t) of the rotor respectively. The state variables are of the system can be define as

- Armature current = $i_a(t)$

- Rotor angular velocity = $\omega_m(t)$
- Rotor angular displacement =$\theta_m(t)$

It is possible to eliminate all the non-state variables from Equation 3 through 7 by direct substitution then present the dc state equation in vector-matrix form as follows:

$$\begin{bmatrix} \frac{di_a}{dt}(t) \\ \frac{d\omega_m(t)}{dt} \\ \frac{d\theta_m(t)}{dt} \end{bmatrix} = \begin{bmatrix} -\frac{R_a}{L_a} & -\frac{K_b}{L_a} & 0 \\ \frac{K_i}{J_m} & -\frac{B_m}{J_m} & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} i_a(t) \\ \omega_m(t) \\ \theta_m(t) \end{bmatrix} + \begin{bmatrix} \frac{1}{L_a} \\ 0 \\ 0 \end{bmatrix} e_a(t) + \begin{bmatrix} 0 \\ -\frac{1}{J_m} \\ 0 \end{bmatrix} T_L(t). \quad (8)$$

Note that in the case of Equation 8 above, that $T_L(t)$ is handled as a second input to the state equations. The transfer function between the motor displacement and the input voltage is obtained as thus;

$$\frac{\Theta_m(s)}{E_a(s)} = \frac{k_i}{L_a J_m s^3 + (R_a J_m + B_m L_a)s^2 + (K_b K_i + R_a B_m)s} \quad (9)$$

Note that T_L has been set to zero in Equation 9. Fig 5 shows a block diagram of the dc motor system for speed control. From the diagram, one can see clearly how the transfer function is related to each block. It can be seen from Equation 9 that *s* can be factored out of the denominator and the significance of the transfer function $\frac{\Theta_m(s)}{E_a(s)}$ is that the dc motor is an integrating device between these two variables. $\Theta_m(s)$is the rotor angular displacement Laplace transfer function, $E_a(s)$ is the input voltage Laplace transfer function and $\Omega_m(s)$ is the transform of angular velocity respectively.

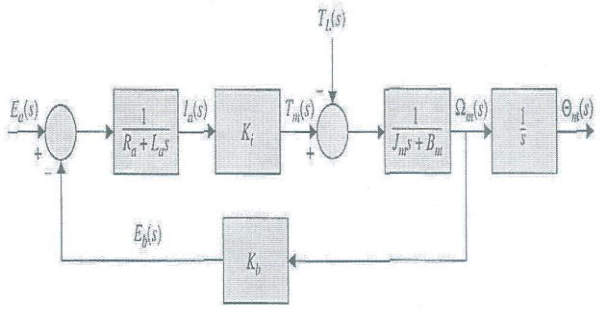

Fig 5: Block Diagram of DC Servomotor in terms of speed

From fig 5 also, it can be seen that the motor has a built-in feedback loop caused by the back emfE_b. The back-emf physical represents the feedback of a signal that is proportional to the negative of the speed of the motor. From equation 9, it can be noted that back emf constant K_b represents an added term to the resistance R_a and the viscous-friction coefficient B_m. Effectively, the back-emf effect is equivalent to an electric friction which tends to improve the stability of the motor and apparently the stability of the system.

4.0 DC Motor Equivalent Circuit In Discrete Form

Recall that ANN is the modeling tool. So simulation can be performed on the control of Dc motor using ANN model, there is need to construct an equivalent DC motor to a discrete time model. Effectively, the load torque is assumed as

$T_L = \mu\omega^2\, m(t)[sgn(\omega_m\,(t))]$ (10)

Where μ = a constant

It is obvious that from equation (10) that load torque always, opposes the direction of motion. Note that the choice of load torque here is arbitral because considering load torque as one of the functions of a DC motor; it is a common characteristic for most propeller driven loads. Then using a sampling time interval of ΔT in discrete form, the derivate of speed and current may be presented in a general way as follows:

$$\frac{d\omega_m}{dT} = \frac{\omega_m(k+1) - \omega_m(k)}{\Delta T} \quad (11)$$

$$\frac{di_a}{dt} = \frac{i_a(k) - i_a(k-1)}{\Delta T}. \quad (12)$$

Where k represents the time instant, then considering equations 11 and 12 above, the discrete equivalent of equations 3.2 and 3.3 will thus be

$$k_e i_a(k) = I_m \frac{[\omega_m(k+1) - \omega_m(k)]}{\Delta T} + b\omega_m(k) + T_L(k) \quad (13)$$

$$k_e\omega_m(k) = -R_a i_a(k) - L_a \frac{[i_a(k) - i_a(k-1)]}{\Delta T} + v_a(k). \quad (14)$$

$$i_a(k) = \frac{I_m}{(\Delta T)k_e}[\omega_m(k+1) - \omega_m(k)] + \frac{b\omega_m(k)}{k_e} + \frac{T_L(k)}{k_e}. \quad (15)$$

$$\frac{i_a(k) - i_a(k-1)}{\Delta T} = \frac{I_m}{k_e(\Delta T)^2}[\omega_m(k+1) - 2\omega_m(k) + \omega_m(k-1)] +$$

$$\frac{b}{k_e(\Delta T)}[\omega_m(k) - \omega_m(k-1)] + \frac{1}{k_e(\Delta T)}[T_L(k) - T_L(k-1)] \quad (16)$$

Next, use equations (15) and (16) above in equation (14) and rearrange then, the finite differential equation translates to:

$$\omega_m(k+1) = \alpha\omega_m(k) + \beta\omega_m(k-1) + \gamma sgn(\omega_m(k))\omega_m{}^2(k) +$$

$$\delta sgn(\omega_m(k-1))\omega_m{}^2(k-1) + \xi V_a(k) \quad (17)$$

The five coefficients: $\alpha, \beta, \gamma, \delta, \text{and } \xi$ of equation 17 have values as follows: $\alpha = \frac{-K_e{}^2(\Delta T)^2 + R_a I_m \Delta T - R_a(b\Delta)^2 + 2L_a I_m - L_a b\Delta T}{(L_a I_m + R_a I_m \Delta T)}$. (18)

$$\beta = \frac{L_a\, b\Delta T - L_a I_m}{(L_a I_m + R_a I_m \Delta T)} \quad (19)$$

$$\gamma = -\frac{\mu L_a \Delta T}{(L_a I_m + R_a I_m \Delta T)} \quad (20)$$

$$\delta = \frac{\mu L_a \Delta T}{(L_a I_m + R_a I_m \Delta T)} \quad (21)$$

$$\xi = \frac{\mu L_a (\Delta T)^2}{(L_a I_m + R_a I_m \Delta T)} \quad (22)$$

Now the next thing to do is to rewrite equation (17) to obtain the final discrete form of the DC motor.

$$v_a(k) = f[\omega_m(k+1), \omega_m(k), \omega_m(k-1) \quad (23)$$

In equation (3.21), $v_a(k)$ is a function of speed at successive time intervals k+1, k and k-1. Therefore, the following form the input speed of the ANN:

$\omega_m(k+1)$. Speed at first instance

$\omega_m(k)$. = speed at second instance

$\omega_m(k-1)$. = speed at third instance

Its obvious that precise control of speed does not mean accurate control of the position. So for accurate position to be achieved, position profile should be incorporated in the equation governing the precise control of the speed by simple integration. Then to improve the performance of the position control, a feedback module should be incorporated into the network. The position error is amplified through the feedback module and that effect used to modify the actual motor speed $\omega_m(k)$. Then, from equation (17),

$$v_a(k) = \frac{A-B-C-D-E}{F} \quad (24)$$

Where: A = $\omega_m(k+1)$, B = $\alpha\omega_m(k)$,
C $\beta\omega_m(k-1)$,
D = $\gamma sign(\omega_m(k))\omega_m{}^2(k)$,
E= $\delta sign(\omega_m(k-1))\omega_m{}^2(k-1)$,
F = ξ.

From reviews, it is known that the load affects the voltage through its inertial and non-inertial terms. The inertial term contributes to the linear variation of the voltage with speed while the non-inertial terms contribute to the non- linear variation of voltage. Viscous parameters are the cause the non-linear effects and these effects due to non-inertial terms are small compared to the effects due to the inertial terms. So if the effects due to the non-inertial terms are neglected in equation (24), from the work of Weerasooriya and El-Sharkawi, (1991), the equation becomes:

$$v_a(k) \propto [\omega_m(k+1) - \alpha\omega_m(k) - \beta\omega_m(k-1)] \quad (25)$$

Since $\omega_m(k+1), \omega_m(k), \omega_m(k-1)$ are speeds at successive interval,

$|\omega_m(k+1) - \omega_m(k)| \ll \omega_m(k+1)$ Then, it can be assumed that the equation

$$\omega_m(k+1) \approx \omega_m(k-1) = \omega_m(k) \quad (26)$$

$$v_a(k) \propto (1-\alpha-\beta)\omega_m(k). \quad (27)$$

Therefore,

$$(1-\alpha-\beta) = 1 - \frac{-K_e{}^2(\Delta T)^2 + R_a I_m \Delta T - R_a(b\Delta)^2 + 2L_a I_m - L_a b\Delta T}{(L_a I_m + R_a I_m \Delta T)} - \frac{L_a b\Delta T - L_a I_m}{(L_a I_m + R_a I_m \Delta T)} \quad (28)$$

After subtraction, equation (28) becomes;

$$\frac{K_e{}^2(\Delta T)^2 + R_a(b\Delta)^2}{(L_a I_m + R_a I_m \Delta T)}. \quad (29)$$

Note that the terms in the right side of equation (29) are positive, effectively it means that $(1-\alpha-\beta)$, is positive as well. Then, if that is the case, it means that $v_a(k)$ increases with increase in motor speed $\omega_m(k)$.

5.0 Controlling The Speed of The DC Motor

The purpose of the AIM controller is to track the speed of the PMDC motor by providing or rather supplying the input voltage signals needed for excitation. The input voltage would serve to follow the reference signal that needs to be controlled or tracked. To control the speed of the DC motor, the first step is to select a suitable stable reference model for the motor to follow. In this work, the reference model is chosen to behave as a second order system whose input is r(k) and whose output is ωd (k). For a continuous time system, the second order reference model can be selected as (Makableh, 2011):

$$\frac{\omega_d(s)}{r(s)} = \frac{\omega_n^2}{s^2 + 2\beta\omega_n s + \omega_n^2} \quad (30)$$

ω_n is the natural frequency. The natural frequency for the PMDC motor chosen for simulation is 60 rad/s. The damping coefficient β is taken as 0.6. For the digital control system, the equivalent transfer function for the model is given by:

$$\frac{\omega_d(z)}{r(z)} = \frac{K(z-z_I)}{(z-p_I)(z-p_I^*)} \quad (31)$$

For a unit step input, the conjugate poles p_I and p_I^* of Equation (31) can be derived as (Kuo, 1980);

$$p_I = \exp(-\beta\omega_n\Delta T)$$

$$\arg(p_I) = \sqrt{1-\beta^2}\,\omega_n\Delta T \quad (32)$$

In order to ensure effective data extraction, the sampling interval ΔT was selected such that the sampling frequency 2π/ΔT is at least twice the natural frequencyω_n. The zero z_1 can be

determined by selecting a suitable time to the peak overshoot. For the system under consideration, the sampling time is taken as 40 ms. The z_I is arbitrarily placed at zero and the K value is set to unity.

$$p_I = \exp(-0.6 * 60 * 40 * 0.001)0.24(33)$$

$$arg(p_I) = \sqrt{1-(0.6)^2} * 60 * 40 * 0.001 = 1.92(34)$$

Using (33) and (34) in Equation 31 and rearranging, the second order reference model in difference equation form becomes:

$$\omega_d(k+1) = 0.6\omega_d(k) + 0.2w_d(k-1) + r(k) \quad (35)$$

r(k) is the bounded input to the reference model. For any given desired sequence ω_d, the corresponding control sequence r(k) can be calculated using Equation (35). The MRAC Simulink Model of the speed trajectory control system is shown in fig 6.

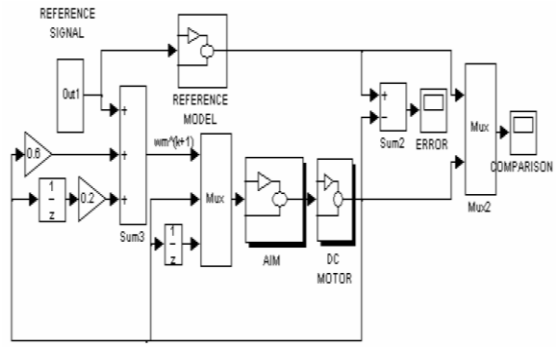

Fig 6: MRAC Simulink Model with Speed Input

The tracking performance was tested for an arbitrary trajectory T_1 as given by

$$\omega_d(K) = 10*sin\ (1.5707k\Delta\ T) + 16*sin\ (0.8976\ k\Delta T)\ \text{k= 1, 2, 3…} \quad (36)$$

The reference trajectory was chosen so that the speed varies within the operating range of the DC motor. Tracing the lines of the actual and the reference signals of figure 7, it can be seen that the DC motor follows the actual profile closely. This conforms with the idea of MRAC having the output of the system being made to following the reference signal.

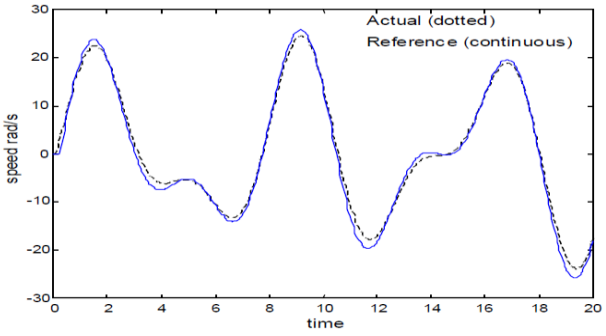

Fig 7: Simulated Speed Control Performance T_I at 0.04S Sampling Time.

The speed error profile is shown in fig 8. Observation from the illustration shows that the speed error varies within –2.5 to +1.4 rad/s. Remember also that the accuracy of tracking depends upon the sampling time.

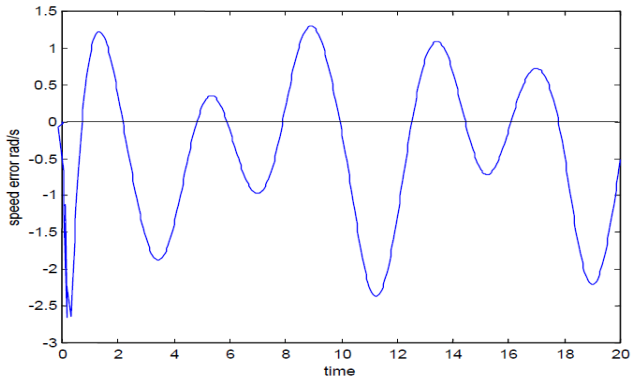

Fig 8: Simulated Speed Control Error T_I at 0.04s Sampling Time.

To continue with the control performance test simulation, the values of α,β,γ, δ and ξ in equations 18 to 22 are recalculated using the new sampling time. The data values for retraining the controller are obtained using equation 24. Figure 9, displays the error plot for the same sampling time of 0.001sec.

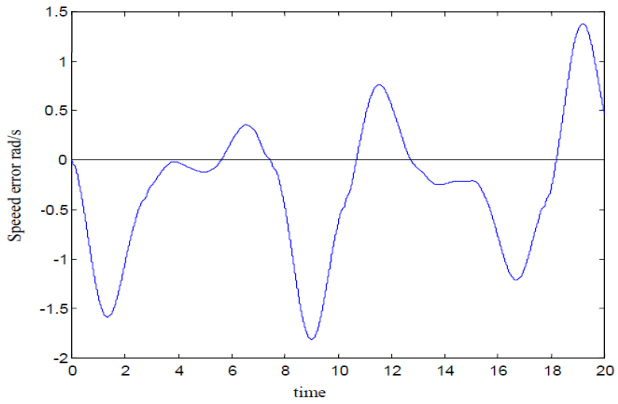

Fig 9: Simulated Speed Control Error T_I at 0.001sec Sampling Time

The error profiles of both simulated speed control of T_I at 0.04sec and 0.001sec respectively when compared, shows that the speed error variation is reduced to within ±1.49 and ±1.51 rad/s, which is within 5% of the actual speed profile. It could be found that by reducing the sampling time the speed profile accuracy can also be improved. Thus, in reality or in physical control of the DC motor, it means making use of a more sophisticated DC motor with higher sampling frequency.

Apparently, reducing the sampling frequency will in turn put a constraint on the size of the controller that can be used. AIM requires finite computation for evaluating its output. A smaller

sampling time implies a smaller structure of AIM has to be used. So, to achieve more accuracy, it will be a good practice to accommodate both situations by compromising between the sampling time and the size of the AIM. The AIM training can also be improved to improve the accuracy of performance of the speed trajectory system. Figure 10 and Figure 11 gives the speed and the speed error profile respectively for a different trajectory T_{II} given by equation 37

$$\omega_d(K) = 10\cos(k\Delta T) + 15\sin(0.5\,k\Delta T) k=1,2,3..... \quad (37)$$

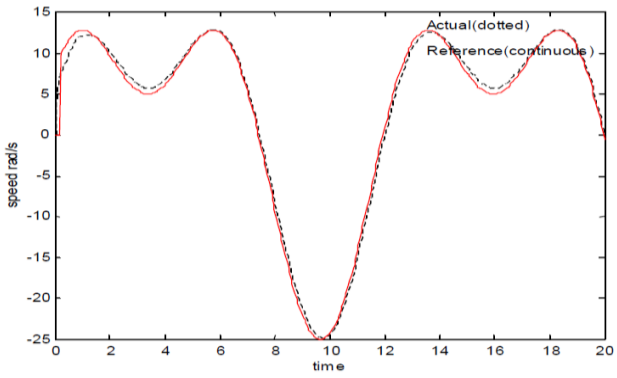

Fig 10: Simulation of Speed Control Performance T_{II}

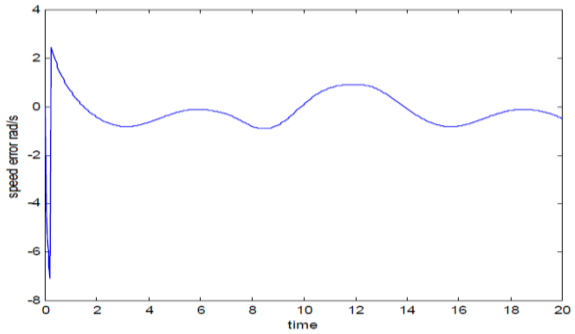

Fig 11: Simulated Speed Control Error T_{II}

Observation from illustration of figure 11 shows that the trajectory tracking system depicts good tracking accuracy within ±1 rad/s. Other similar tests were conducted for a number of profiles at simulation time but, these two profiles were shown for the purpose of illustration.

Conclusion

The model was developed using the motor speed as input to the controller, which is the artificial inverse model (AIM). Simulation was performed for two different arbitrary trajectories using different Sampling times. Results show the speed error variation is reduced to within ±1.49 and ±1.51 rad/s, which is within 5% of the actual speed profile. It was found that by reducing the sampling time, the speed profile accuracy isimproved.

References

Astrom K.J. and Wittenmark B., (1989). Adaptive Control. Addison-Wesley.

BALDOR MOTORS AND DRIVES, 2009).Servo Control Facts.*A Hand Book Explaining the Basics of Motion*. Available online at: www. Baldormotorsanddrives.com. Last Accessed 27th -05-2014.

Beale M.H, Hagan M.T. and Demuth H.B. (2015) Neural Network Toolbox User'

Kuo B. C and Golnaraghi F. (2002). *Automatic Control Systems*, Eight Edition. Pulisher: John Wiley & Sons, Inc. ISBN-13 978-0470-04896-2,

Valluru B.R. (1995). *C++ Neural Networks and Fuzzy Logic*. MTBooks, IDG Books Worldwide, Inc. ISBN:n1558515526.

WeerasooriyaS. and. El-SharkawiM.A. (1995) "Identification and control of a DC motor using Back-propagation Neural Networks", IEEE Transactions on Energy Conversion, Vol.6, No.4, pp. 663-669, December 1991.

Zhao J. and Yu Y. (2011) *Brushless DC Motor Fundamentals Application Note.* MPS AN047