

# SIMULATION TOOLS FOR ELECTRICAL MACHINES MODELLING: TEACHING AND RESEARCH

T.C.NWODO

*Dept. of Electrical/Electronics Engineering  
Federal University of Technology, Owerri  
Imo State  
Email: [tcnwodo@yahoo.com](mailto:tcnwodo@yahoo.com)*

O.I.OKORO

*Dept. of Electrical Engineering  
University of Nigeria, Nsukka  
Enugu State.  
Email: [oiokoro@hotmail.com](mailto:oiokoro@hotmail.com)*

## ABSTRACT

*Simulation tools are used both for research and teaching to allow a good comprehension of the systems under study before practical implementations. This paper illustrates the way MATLAB is used to model non-linearities in synchronous machine. The machine is modeled in rotor reference frame with currents as state variables. The solution provide an easy way of determining the dynamic behaviour of the machine at run-up conditions. The MATLAB script is presented in Appendix 1 to encourage the reader use the script to verify the results presented. It is also expected that lecturers in Electrical machines who are involved in engineering undergraduate education will find the script very useful in terms of demonstrating the effect of non-linearities in Asynchronous machine dynamics.*

**Keywords:** Asynchronous machine; MATLAB scripts; engineering education; skin-effect; saturation effect; dynamic behaviour

## 1.0 Introduction

The gross under funding of Nigerian Universities has adversely affected research efforts in our nation's engineering faculties. Research facilities are in short supply and in some cases non-existent [1]. Where the former is the case, the amount of research in those Universities is better imagined than described. The equipment are not only obsolete but also in limited supply to the extent that the student-equipment ratio is usually alarming. In such situation, there is an urgent need to look at alternative method of research which will not only conserve the existing facilities but also reduce the enormous effort wasted in design and testing of results. System modelling and simulation of the actual physical system offer a better alternative. Simulation can be very useful in many scientific and engineering research that proceed as follows [2]:

- Observing the physical system
- Formulating a hypothesis or

mathematical model to explain the observation

- Predicting the behaviour of the system from solutions or properties of the mathematical model.
- Testing the validity of the hypothesis or mathematical model.

Depending on the nature of the actual physical system and the purpose of the simulation, the definitions of modelling and simulation will vary. Generally, simulation is a technique that entails setting up a model of a real situation and performing experiments on the model. The procedure for developing a model often take the following forms:

- Identifying the purpose of the model as to be able to impose on it any simplifying assumptions.
- Determining the means

(Instrumentation requirements) of obtaining parameters for the model.

- Defining the available computational facilities.

It is important to note that simplicity is the hallmark of most good models. The results of Okoro [3,4] amply attest to that. Whereas oversimplification and omissions may lead to unacceptable loss of accuracy as can be seen in [4]; where a great error existed in the conventional model as compare to when saturation and skin effects were included in the modelling of induction machine. However, a compromise between an oversimplified model and a detailed model must be reached as the later can be cumbersome to use both in computer simulation and hardware implementation [5,6]. In this paper, the need for MATLAB as a pedagogical tool in research and teaching is highlighted. A practical application which studies the effect of non-linearities in asynchronous machine is used to validate the enormous potentials of MATLAB. Lastly, relevant conclusion is made and way forward proposed in engineering research in Nigerian Universities.

### 2.0 Conventional Machine Model

The conventional model for asynchronous machine assumes a linear air-gap, balanced stator voltages with saturation and skin effects neglected. This method of analysis is well known and can be found in almost any standard textbook in electrical machine dynamics [2,7,8]. Thus, the electrical model of the squirrel-cage asynchronous machine in rotor reference frame is given by [7,8] as:

$$\begin{bmatrix} V_{qs} \\ V_{ds} \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} (R_s + L_s p) & -\omega_r L_s & L_m p & \omega_r L_m \\ -\omega_r L_s & (R_s + L_s p) & -\omega_r L_m & L_m p \\ L_m p & 0 & (R_r + L_r p) & 0 \\ 0 & L_m p & 0 & (R_r + L_r p) \end{bmatrix} \begin{bmatrix} i_{qs} \\ i_{ds} \\ i_{qr} \\ i_{dr} \end{bmatrix} \quad (1)$$

For the purpose of digital simulation, equation (1) is represented in state variable form with currents as state variables [9].

$$p[i] = -[L]^{-1}([R] + \omega_r[G])[i] + [L]^{-1}[V] \quad (2)$$

Where,

$$[V] = [V_{qs} V_{ds} 0 0] \quad (3)$$

$$[R] = \begin{bmatrix} R_s & 0 & 0 & 0 \\ 0 & R_s & 0 & 0 \\ 0 & 0 & R_r & 0 \\ 0 & 0 & 0 & R_r \end{bmatrix} \quad (4)$$

$$[L] = \begin{bmatrix} L_s & 0 & L_m & 0 \\ 0 & L_s & 0 & L_m \\ L_m & 0 & L_r & 0 \\ 0 & L_m & 0 & L_r \end{bmatrix} \quad (5)$$

$$[G] = \begin{bmatrix} 0 & L_s & 0 & L_m \\ -L_s & 0 & -L_m & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (6)$$

$$[i] = [i_{qs} i_{ds} i_{qr} i_{dr}] \quad (7)$$

The electromagnetic torque,  $T_e$  is given by [7] as:

$$T_e = \frac{3}{2} P L_m (i_{qs} i_{dr} - i_{ds} i_{qr}) \quad (8)$$

The stator voltages are balanced and represented as:

$$V_{as} = \sqrt{2V} \cos \omega_b t \quad (9)$$

$$V_{bs} = \sqrt{2V} \cos \left( \omega_b t - \frac{2\pi}{3} \right) \quad (10)$$

$$V_{cs} = \sqrt{2V} \cos \left( \omega_b t + \frac{2\pi}{3} \right) \quad (11)$$

These stator voltages are related to the d-q frame of reference by equation (12),

$$\begin{bmatrix} V_{qs} \\ V_{ds} \end{bmatrix} = [C_1] \begin{bmatrix} V_{as} \\ V_{bs} \\ V_{cs} \end{bmatrix} \quad (12)$$

Where,

$$C_1 = \frac{2}{3} \begin{bmatrix} \cos \theta, \cos \left( \theta_r - \frac{2\pi}{3} \right), \cos \left( \theta_r - \frac{4\pi}{3} \right) \\ \sin \theta, \sin \left( \theta_r - \frac{2\pi}{3} \right), \sin \left( \theta_r - \frac{4\pi}{3} \right) \end{bmatrix}$$

### 3.0 Saturation Effect Model.

The values of the inductances used in the development of the dynamic equations for the conventional machine model were assumed to be constant. By so doing, the model fails to take into consideration the saturation effects of both the magnetizing inductance and the leakage inductances. In this paper, it is assumed that only the main inductance varies with the magnetizing current. Therefore, the saturation curves of the test machine have to be determined. This curve which shows the dependency of the magnetizing inductance with the magnetizing current determined as

reported by [4] is shown in Figure 1. In order to find an analytical expression for the saturation characteristic curve of Figure 1, a curve-fitting method which employs the algorithm of Marquardt [10] is employed. In figure 1, the estimated function becomes,

$$L_m = 0.064i_m^4 - 0.94i_m^3 + 2.4i_m^2 - 1.4i_m + 230[mH] \quad (14)$$

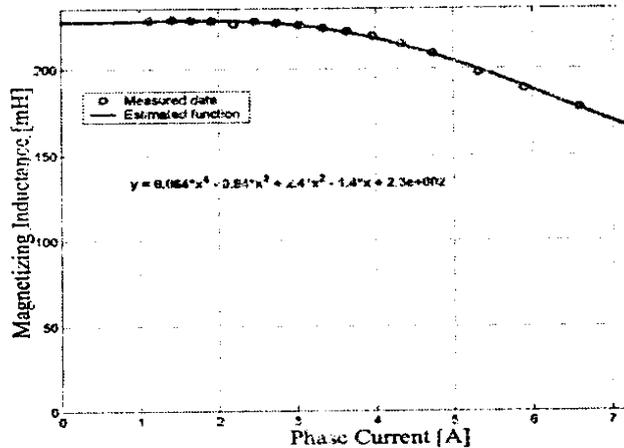


Figure 1: Graph of magnetizing inductance against magnetizing.

The magnetizing current,  $i_m$  is defined as,

$$i_m = \sqrt{i_{dt}^2 + i_{qt}^2} \quad (15)$$

And

$$i_{dt} = i_{ds} + i_{dr} \quad (16)$$

$$i_{qt} = i_{qs} + i_{qr} \quad (17)$$

By storing the analytical expression in the computer, the value of the magnetizing inductance in both the induction machine conventional model can be updated at each integration step

### 1.0 Skin-Effect Model.

The test machine has a rectangular- shaped rotor bar. In order to account for the influence of skin-effect, the lumped parameter network model is applied with the rotor divided into three sections as shown in Figure 2.

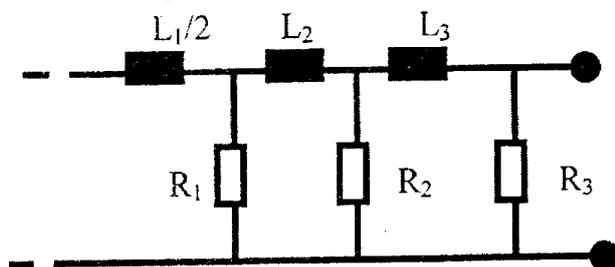


Figure 2: Configuration for 3-Section rotor bar.

In order to optimise the model impedance so as to give a good correction to the actual rotor bar impedance, an error function MATLAB programme is developed. The optimised values of the model impedance at 4KHz frequency shown in Table 1 are therefore used to represent accurately the actual machine in dynamic condition with skin-effect phenomenon included. The elaborate analysis on the bar division and model optimisation could be found in Reference [4]

Table 1: computed model impedance at 4KHz [4]

Resistance	[mΩ]	Inductance	[μH]
R <sub>1</sub>	0.669	L <sub>1</sub>	3.056e-2
R <sub>2</sub>	0.328	L <sub>2</sub>	4.647e-2
R <sub>3</sub>	0.165	L <sub>3</sub>	0.0948

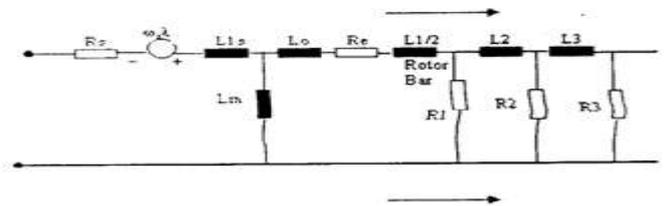


Figure 3: Skin-effect model equivalent circuit,

Decomposing Figure 3 into d- and q-axis with rotor values referred to the stator, the stator and rotor loop equations become [4]:

- **Stator Equations**

$$V_{ds} = R_s i_{ds} + L_s \frac{di_{ds}}{dt} - L_s \omega_r i_{qs} - L_m \omega_r i_{q1} + L_m \frac{di_{d1}}{dt} \quad (18)$$

$$V_{qs} = R_s i_{qs} + L_s \frac{di_{qs}}{dt} + L_s \omega_r i_{ds} + L_m \frac{di_{q1}}{dt} + L_m \omega_r i_{d1} \quad (19)$$

- **Rotor Equation**

$$V_{D1} = 0 = (R_r + R1r) i_{D1} - R1r i_{D2} + L1r \frac{di_{D1}}{dt} + L_m \frac{di_{ds}}{dt} + L_m \frac{di_{d1}}{dt} \quad (20)$$

$$V_{Q1} = 0 = (R_r + R1r) i_{Q1} - R1r i_{Q2} + L1r \frac{di_{Q1}}{dt} + L_m \frac{di_{qs}}{dt} + L_m \frac{di_{q1}}{dt} \quad (21)$$

$$V_{D2} = 0 = R1r i_{D2} - R1r i_{D1} + R2r i_{D3} + L2r \frac{di_{D2}}{dt} \quad (22)$$

$$V_{Q2} = 0 = R1r i_{Q2} - R1r i_{Q1} + R2r i_{Q3} - R2r i_{Q3} + L2r \frac{di_{Q2}}{dt} \quad (23)$$

$$V_{D3} = 0 = R2r i_{D3} - R3r i_{D3} + R2r i_{D2} + L3r \frac{di_{D3}}{dt} \quad (24)$$

$$V_{Q3} = 0 = R2r i_{Q3} - R3r i_{Q3} + R2r i_{Q2} + L3r \frac{di_{Q3}}{dt} \quad (25)$$

In state variable form, the variables in equation (2) are defined thus:

$$[R]= \begin{bmatrix} R_s & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & R_s & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & R_{11} & 0 & -R_{1r} & 0 & 0 & 0 \\ 0 & 0 & 0 & R_{11} & 0 & -R_{1r} & 0 & 0 \\ 0 & 0 & -R_{1r} & 0 & R_{22} & 0 & -R_{2r} & 0 \\ 0 & 0 & 0 & -R_{1r} & 0 & R_{22} & 0 & -R_{2r} \\ 0 & 0 & 0 & 0 & -R_{2r} & 0 & R_{33} & 0 \\ 0 & 0 & 0 & 0 & 0 & -R_{2r} & 0 & R_{33} \end{bmatrix} \quad (26)$$

Where ,

$$R_{11} = R_{er} + R_{1r}; R_{22} = R_{1r} + R_{2r}; R_{33} = R_{2r} + R_{3r} \quad (27)$$

$$[G]= \begin{bmatrix} 0 & -L_s & 0 & -L_m & 0 & 0 & 0 & 0 \\ L_s & 0 & L_m & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (28)$$

$$[L]= \begin{bmatrix} L_s & 0 & L_m & 0 & 0 & 0 & 0 & 0 \\ 0 & L_s & 0 & L_m & 0 & 0 & 0 & 0 \\ L_m & 0 & L_{mr} & 0 & 0 & 0 & 0 & 0 \\ 0 & L_m & 0 & L_{mr} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & L_{2r} & L_{2r} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & L_{3r} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & L_{3r} \end{bmatrix} \quad (29)$$

Where,

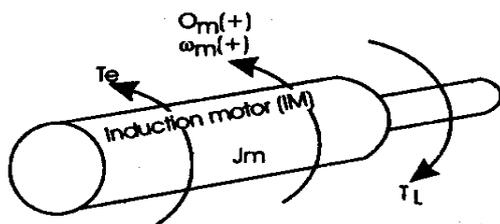
$$L_{mr} = L_{1r} + L_m \quad (30)$$

The rotor parameters in Figure 3 have been referred to the stator using appropriate transformation ratio as defined in [4].

### 5.0 Mechanical Model.

The mechanical model of an asynchronous motor comprises of the equations of motion of the motor and driven load as shown in Figure 4 and is usually represented as a second-order differential equation [11]

$$J_m p^2 \theta_m = T_e - F \omega_r - T_L \quad (31)$$



**Figure 4: Motor mechanical model.**

The mechanical data of the test machine obtained from the manufacturer indicates that the combined rotor and load viscous friction, F

is approximately zero, thus equation (31) becomes,

$$P \theta_m = \omega_m \quad (32)$$

$$J_m (p \omega_m) = (T_e - T_L) \quad (33)$$

But,

$$\omega_r = \omega_m P \quad (34)$$

$$\theta_r = \theta_m P \quad (35)$$

Where  $\omega_m, \theta_m, \theta_r, \omega_r, J_m$ , and  $T_L$  angular velocity of the rotor, rotor angular position, electrical rotor angular position, electrical angular velocity, combined rotor and load inertia coefficient, and applied load torque respectively.

### 6.0 MATLAB solution and results.

In order to study the effect of non-linearities on the dynamic behaviour of asynchronous motor, equation (2), together with equation (14), equation (8) and equation (33)-which gives the mechanical behaviour of the motor are solved by developing MATLAB m-files [12] as shown in Appendix 1 which incorporate an in-built numerical algorithm, ODE-45-a programme that uses Runge-Kutta numerical method. The simulations have been carried out using the motor data obtained from the no-load, blocked-rotor, D.C. measurement, and retardation tests of the motor under study as well as the computed rotor bar impedance of Table 1. The machine parameter is shown in Table 2

**Table2: test machine parameter.**

Output Power	7.5KW
Rated voltage	340V
Winding connection	Delta
Number of poles	4
Rated speed	1400rpm
Rated frequency	50Hz
Stator reactance	2.252195Ω
Rotor leakage reactance	1.95145Ω
Rotor resistance	0.976292Ω
Rotor leakage reactance	2.9945Ω
Magnetizing reactance	55.3431Ω
Mechanical shaft torque	51.26N.m
Estimated rotor inertia moment	0.117394kgm <sup>2</sup>
Rate current	19.2
Moment of inertia of the D. C motor	0.1096Kgm <sup>2</sup>
Shaft stiffness constant	14320Nm/rad

Programming the above mentioned differential equations in MATLAB is straightforward for the reader familiar with it. However, for the reader not familiar with MATLAB, references [12] and [13] should be consulted. The developed m- files are categorized broadly into two: One is the function file (nwodokoro 1. m) and the other the calling programme (nwodokoro 2.m). The function file defines the differential equations for the dynamic studies while the calling programme uses the in-built fourth-order Runge-Kutta method to solve the equations with the initial currents and speed values. Details of the programme are presented in Appendix 1 with adequate comments (%) to clarify processes.

The programme is highly interactive and permits the user with the options of choosing either conventional model, conventional model with saturation effect, skin-effect model or end of the programme. The programme accepts as its inputs the machine parameters and the rotor circuit parameters. By typing nwodokoro2 at the MATLAB command window, the programme is executed while the response curves for the dynamic studies as shown in figure 5, figure 6 and figure 7 are displayed on the figure window

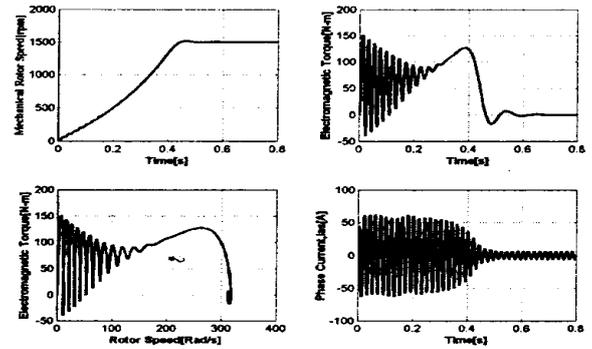


Figure 5: Conventional model

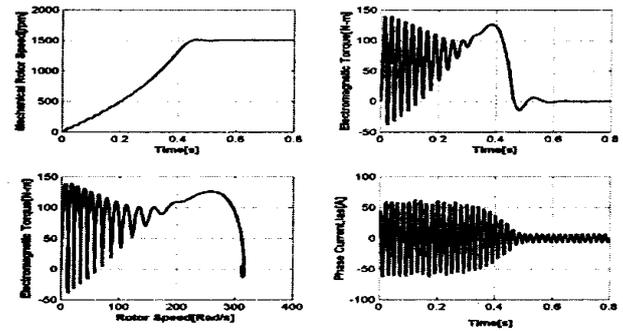
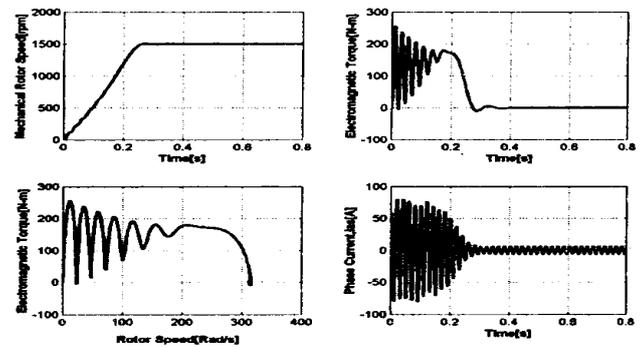


Figure 6: Conventional and saturation effect model.



### 1.0 Conclusion

The paper has shown that it is possible to use MATLAB to model and simulate the dynamic behaviours of asynchronous machine. Through the use of interactive MATLAB file, students/researchers now have a tool to play with as they visualize the effects of non-linearities in the dynamic performance of asynchronous motor. The figures presented in section 6 shows that the time function of the mechanical rotor speed rises faster to synchronous speed in skin-effect model than the conventional model. These simulation results clearly show the errors that are involved when the conventional model is used to predict the dynamic behaviours of asynchronous machine without inclusion of

saturation and/or skin-effect.

The authors therefore, make a case for the use of MATLAB as a teaching and research tool in engineering education since it is easy to learn, versatile in application, enhanced library of functions and interactive and editable environment.

## REFERENCES

- [1] O.I. Okoro, "Application of numerical softwares in electrical machines modeling", Proceedings of the Electrical Division of the Nigerian Society of Engineers conference, October 2004, pp.1-7.
- [2] C. Ong, *Dyanmic Simulation of Electric Machinery: Using MATLAB/SIMULINK*, Prentice-Hall Inc., New Jersey, 1997.
- [3] O.I. Okoro, "Steady-state simulation of induction machine with non-linear effects", Proceedings of the Electrical Division of the Nigerian Society of Engineers conference, October 2003, pp.94-99.
- [4] O.I. Okoro, *Dynamic and Thermal Modelling of Induction Machine with Non-Linear Effects*, Kassel University Press, Germany, 2002.
- [5] P.C. Krause and C.H. Thomas, Simulation of symmetrical induction machinery", *Transactions IEEE, PAS-84*, Vol 11, 1965, pp.1038-1053.
- [6] P. Barrade, "Simulation tools for power electronics: Teaching and Research", pp.35 46.
- [7] P.c. Krause, *Analysis of Electric Machinery and Drive Systems*, McGraw-Hill, New York, 1989.
- [8] R. Krishman, *Electric Motor Drives: Modelling Analysis and Control*, Prentice-Hall Inc., New Jersey, 2001.
- [9] O.I. Okoro, "MATLAB simulation of induction machine with saturable leakage and magnetizing inductances", *Botswana Journal of Technology*, Vol. 13, No. 2,2004, pp. 20-28.
- [10]D.W. Marquardt, "An Algorithm for least-square estimation of non-linear parameters" *J Soc. Ind. Appl. Math*, vol.11, No.2, June 1963,pp.431-441.
- [11] Peter Vas, *Electrical machines and drives-A space vector theory approach*, Clarendon Press, Oxford, 1992.
- [12] *MATLAB User's Guide*. The Mathworks, Inc, Natick, 1991.
- [13]A. Biran and M. Breiner, *MATLAB 5 for Engineers*, Addison-Wesley, New York, 1999.

[5] P.C. Krause and C.H. Thomas, Simulation of

```

Appendix 1: MATLAB Script for the dynamic studies.
%*****
% This programme is a function program. %The
name is nwodokoro1.m
%It solves the induction machine differential
%equations using the
%State variable equation Xdot=AX +BU
%The program is simulated with %nwodokoro2.m
%Convectional Model, Saturation Model and Skin-
effect Model are
%are included.
%*****
function Xdot=nwodokoro1(t,y,flag)
global flag
global iu to tf tspan x0 TL Lm Ls R Lr Llr Lls V1 V
wb G P J Te Rs Rr L Lind A1 iabc1 idq ias ibs ics
theta I IT f
global id iq t0 tf tspan x0 L0 TL Lm Lmd Lmq Rs
Rer Lls V L1r L2r L3r R1r R2r R3r wb P K Me Lm1
i_s
global iQ1 iD1 iQ2 iD2 iD3 iQ3 Psimd Psimq Iabc
VQD T_Park Tinv_Park V_qsds To
%*****
%1. CONVECTIONAL MODEL %ONLY(CMO)
%*****
if flag==1 %IN ROTOR REFERENCE
%FRAME(w=wr).
Xdot=zeros(6,1);
%i=[iqs ids iqr idr]; %CURRENTS AS %STATE
VARIABLE
%D-Q CURRENTS*****
I=[y(1) y(2) y(3) y(4)];
IT=[y(1) y(2) y(3) y(4)];
%*****
I1_s=y(1)+y(3);
I1_r=y(2)+y(4);

```

```

iu=sqrt(I1_s^2+I1_r^2);
%STATOR CURRENTS*****
Tinv_Park=[cos(y(6))      sin(y(6))
           cos(y(6)-2*pi/3) sin(y(6)-2*pi/3)
           cos(y(6)-4*pi/3) sin(y(6)-4*pi/3)];
iqd=[y(1) y(2)];
theta=(y(6));
Iabc=Tinv_Park*iqd;
ias=Iabc(1); %Phase "a" current
%STATOR VOLTAGES*****
Vas=sqrt(2)*V*cos(wb*t);
Vbs=sqrt(2)*V*cos(wb*t-2*pi/3);
Vcs=sqrt(2)*V*cos(wb*t+2*pi/3);

%D-Q-VOLTAGE TRANSFORMATION*
%IN STATIONARY REFERENCE %FRAME
VQD=[Vas Vbs Vcs]'; % Stator voltages in
%Matrix Form
T_Park = (2/3)*[cos(y(6)) cos(y(6)-(2*pi/3))
               cos(y(6)-(4*pi/3))
               sin(y(6)) sin(y(6)-(2*pi/3)) sin(y(6)-(4*pi/3))];
%Park %transform(Non-Power %invariant form)
%D-Q-VOLTAGE TRANSFORMATION
%IN ROTOR REFERENCE FRAME
V_qsds=T_Park*VQD;
Vqs=V_qsds(1);
Vds=V_qsds(2);
%ELECTROMAGNETIC TORQUE
Te=1.5*P*Lm*(y(1)*y(4)-y(2)*y(3));
%Stator/Rotor resistances
R=diag([Rs,Rs,Rr,Rr]); R-matrix
Lind=[Ls 0 Lm 0;0 Ls 0 Lm;Lm 0 Lr 0;0 Lm 0
Lr]; %L-Matrix
G=[0 Ls 0 Lm;-Ls 0 -Lm 0;0 0 0 0;0 0 0 0]; %G-
Matrix
V1=[Vqs Vds 0 0]'; %V-Matrix
L=inv(Lind);
%STATE VARIABLE FORM
%Xdot=AX+BU*****
idot=-L*(R+y(5)*G)*I+L*V1;
%MECHANICAL ROTOR SPEED
Me=(y(5)*60)/(P*2*pi);
%SPEED EQUATION*****
Xdot(5)=(P/J)*(Te-TL);
%Xdot(5)=(P/J1)*(Te-y(8));
%ROTOR ANGLE EQUATION
Xdot(6)=y(5);
%Y=[ids iqs idr iqr wr theta_r]';
Y=[y(1) y(2) y(3) y(4) y(5) y(6)]'; %The %state
variables
%Complete function equations
Xdot=[idot; Xdot(5);Xdot(6)];
end
%*****
%2. SKIN-EFFECT MODEL %ONLY(SEMO)
*
%*****
if flag==2 % (IN ROTOR REFERENCE
%FRAME WITH CURRENTS AS STATE
%VARIABLES)
Xdot=zeros(10,1);
%i=[ids iqs id1 iQ1 id2 iQ2 id3 iQ3 wr %thetar];
%D-Q CURRENTS*****
I=[y(1) y(2) y(3) y(4) y(5) y(6) y(7) y(8)]';
%STATOR CURRENTS*****
Iabc1=[cos(y(10))      sin(y(10))
       cos(y(10)-2*pi/3) sin(y(10)-2*pi/3)
       cos(y(10)+2*pi/3) sin(y(10)+2*pi/3)];
idq=[y(1) y(2)]';
theta=(y(10));
ias=y(2)*cos(theta)+y(1)*sin(theta);
ibs=y(2)*cos(theta-2*pi/3)+y(1)*sin(theta-2*pi/3);
ics=y(2)*cos(theta+2*pi/3)+y(1)*sin(theta+2*pi/3);
%STATOR VOLTAGES*****
Vas=sqrt(2)*V*cos(wb*t);
Vbs=sqrt(2)*V*cos(wb*t-2*pi/3);
Vcs=sqrt(2)*V*cos(wb*t+2*pi/3);
%*****D-Q-VOLTAGE %TRANSFORMATION
IN STATIONARY %REFERENCE FRAME
Vqs1=2/3*(Vas-0.5*Vbs-0.5*Vcs);
Vds1=(-Vbs+Vcs)/sqrt(3);
%D-Q-VOLTAGE TRANSFORMATION
%IN ROTOR REFERENCE FRAME
Vqs=Vqs1*cos(y(10))-Vds1*sin(y(10));
Vds=Vqs1*sin(y(10))+Vds1*cos(y(10));
%Magnetizing Flux
Psimd=Lm*(y(1)+y(3));
Psimq=Lm*(y(2)+y(4));
%ELECTROMAGNETIC TORQUE
Te=1.5*P*(Psimd*y(2)-Psimq*y(1));
L=[%Insert Equation 29];
R=[%Insert Equation 26];
G=[%Insert equation 28];
L_inv=inv(L);
V1=[Vds Vqs 0 0 0 0 0]'; %V-Matrix
%State variable form
IDx=-L_inv*(R+y(9)*G)*I+L_inv*V1;
%SPEED EQUATION*****
Xdot(9)=(P/J)*(Te-TL);
%*****
% THE ROTATIONAL SPEED
%EQUATION(Theta_r)
Xdot(10)=y(9);
%*****
Y=[y(1) y(2) y(3) y(4) y(5) y(6) y(7) y(8) y(9)
y(10)]'; %The state variables
%MECHANICAL ROTOR SPEED****
Me=(y(9)*60)/(P*2*pi);
Xdot=[IDx; Xdot(9);Xdot(10)];
end
if flag==3 %IN ROTOR REFERENCE
FRAME(w=wr).
%*****
%3. CONVENTIONAL MODEL PLUS
%SATURATION(CMPS)*
%*****

```

```

Xdot=zeros(8,1);
%i=[iqs ids iqr idr];
%D-Q CURRENTS*****
I=[y(1) y(2) y(3) y(4)];
IT=[y(1) y(2) y(3) y(4)];
%*****
I1_s=y(1)+y(3);
I1_r=y(2)+y(4);
iu=sqrt(I1_s^2+I1_r^2);
Lm=(6.4e-2*iu^4-0.94*iu^3+2.4*iu^2-
1.4*iu+230)*1e-3;
Lr=Llr+Lm;
Ls=Lls+Lm;
%STATOR CURRENTS*****
Tinv_Park=[cos(y(6)) sin(y(6)) %Park
%Inverse transform
cos(y(6)-2*pi/3) sin(y(6)-2*pi/3)
cos(y(6)-4*pi/3) sin(y(6)-4*pi/3)];
iqd=[y(1) y(2)];
theta=(y(6));
Iabc=Tinv_Park*iqd;
ias=Iabc(1); %Phase "a" current
%STATOR VOLTAGES*****
Vas=sqrt(2)*V*cos(wb*t);
Vbs=sqrt(2)*V*cos(wb*t-2*pi/3);
Vcs=sqrt(2)*V*cos(wb*t+2*pi/3);
%D-Q-VOLTAGE TRANSFORMATION
%IN STATIONARY REFERENCE %FRAME
VQD=[Vas Vbs Vcs];
T_Park = (2/3)*[cos(y(6)) cos(y(6)-(2*pi/3))
cos(y(6)-(4*pi/3))
sin(y(6)) sin(y(6)-(2*pi/3)) sin(y(6)-(4*pi/3))];
%Park transform(Non-Power %invariant form)
%D-Q-VOLTAGE TRANSFORMATION
%IN ROTOR REFERENCE FRAME*
V_qsds=T_Park*VQD;
Vqs=V_qsds(1);
Vds=V_qsds(2);
%ELECTROMAGNETIC TORQUE
Te=1.5*P*Lm*(y(1)*y(4)-y(2)*y(3));
%Stator/Rotor resistances
R=diag([Rs,Rs,Rr,Rr]); %R-matrix
Lind=[Ls 0 Lm 0;0 Ls 0 Lm;Lm 0 Lr 0;
0 Lm 0 Lr]; %L-Matrix
G=[0 Ls 0 Lm;-Ls 0 -Lm 0;0 0 0 0;0 0 0 0];
%G-Matrix
V1=[Vqs Vds 0 0]'; %V-Matrix
L=inv(Lind);
%STATE VARIABLE FORM %Xdot=AX+BU
idot=-L*(R+y(5)*G)*I+L*V1;
%MECHANICAL ROTOR SPEED
Me=(y(5)*60)/(P*2*pi);
%SPEED EQUATION*
Xdot(5)=(P/J)*(Te-TL);
%ROTOR ANGLE EQUATION
Xdot(6)=y(5);
%Y=[ids iqs idr iqr wr theta_r];
Y=[y(1) y(2) y(3) y(4) y(5) y(6)];

%The state variables
%Complete function equations
Xdot=[idot; Xdot(5);Xdot(6)];
end
%*****
%The name of this Programme file is
%nwodokoro2.m
%This is a general purpose programme that %solves
the differential equations
%of induction machine in Rotor reference
%frames.The resistances, inductances and
%reactances are all in S.I. unit.A function
%Program,nwodokoro1.m is %used to call %this
parameter file.Machine models with %and without
%Skineffect are considered.
%*****
global flag
test=0;
while test==0
disp(' ')
disp(['1. Conventional Machine Model']);
disp(['2. Machine Model with Skineffect ']);
disp(['3. Conventional Machine Model with
saturation effect']);
disp(['4 End of the programme']);
disp(' ');
flag=input('Please indicate your Simulation
modes[1 or 2, etc]: ');
if flag<1
disp(' ');
disp(['Sorry the selection is between 1 and 4']);
disp(' ');
elseif flag>4
disp(' ');
disp(['Sorry the selection is between 1 and 4']);
elseif flag==4
disp('Thank you. You have come to the end of the
programme on Dynamic simulation of IM');
return;
else
test=1;
end
end
if flag==1
parameter1a; %Parameter in Table2.
end
if flag==2
parameter1b; %Parameter in Table 2 and Table 3.
end

if flag==3
parameter1c; %Parameter in table 2.
end
*****
[t,y]=ode45('nwodokoro1',tspan,x0);
if flag==1 %CONVENTIONAL %MACHINE
MODEL.
%D-Q-CURRENTS

```

```

%i=[iqs ids iqr idr];
I=[y(:,1) y(:,2) y(:,3) y(:,4)];
IT=[y(:,1) y(:,2) y(:,3) y(:,4)];
I1_s=y(:,1)+y(:,3);
I1_r=y(:,2)+y(:,4);
%ELECTROMAGNETIC TORQUE
Te=1.5.*P.*Lm.*(y(:,1).*y(:,4)-y(:,2).*y(:,3));
%STATOR VOLTAGES*****
Vas=sqrt(2)*V.*cos(wb.*t);
Vbs=sqrt(2)*V.*cos(wb.*t-2.*pi/3);
Vcs=sqrt(2)*V.*cos(wb.*t+2.*pi/3);
VQD=[Vas Vbs Vcs]'; % Stator voltages in
%Matrix Form
%D-Q-VOLTAGE TRANSFORMATION %IN
STATIONARY REFERENCE %FRAME
T_Park = (2/3).*[cos(y(:,6)) cos(y(:,6)-
(2.*pi/3)) cos(y(:,6)-(4.*pi/3))
sin(y(:,6)) sin(y(:,6)-(2.*pi/3)) sin(y(:,6)-
(4.*pi/3))]; %Park %transform(Non-Power
%invariant form)
theta=(y(:,6)); %ROTOR %ANGLE, THETA_R
%D-Q-VOLTAGE TRANSFORMATION
%IN ROTOR REFERENCE FRAME
V_qsds=T_Park*VQD;
Vqs=V_qsds(1,:);
Vds=V_qsds(2,:);
Vq=Vqs;
Vd=Vds;
%THE STATOR PHASE CURRENTS
%D-Q- TRANSFORMATION****
Tinv_Park=[cos(y(:,6)) sin(y(:,6))
cos(y(:,6)-(2.*pi/3)) sin(y(:,6)-(2.*pi/3))
cos(y(:,6)-(4.*pi/3)) sin(y(:,6)-(4.*pi/3))];
iqd=[y(:,1) y(:,2)];
theta=(y(:,6));
Iabc=Tinv_Park*iqd
ias=y(:,1).*cos(y(:,6))+y(:,2).*sin(y(:,6));
%Phase "a" current
%*MECHANICAL ROTOR SPEED**
Me=(y(:,5).*60)/(P.*2.*pi);
end
if flag==2 %MACHINE MODEL WITH
%SKIN-EFFECT(CURRENTS AS STATE
%VARIABLES)
I=[y(:,1) y(:,2) y(:,3) y(:,4) y(:,5) y(:,6) y(:,7)
y(:,8)];
Vas=sqrt(2).*V.*cos(wb.*t);
Vbs=sqrt(2).*V.*cos(wb.*t-2.*pi/3);
Vcs=sqrt(2).*V.*cos(wb.*t+2.*pi/3);
Vqs1=2/3.*(Vas-0.5.*Vbs-0.5.*Vcs);
Vds1=(-Vbs+Vcs)/sqrt(3);
Vq=Vqs1.*cos(y(:,10))-Vds1.*sin(y(:,10));
Vd=Vqs1.*sin(y(:,10))+Vds1.*cos(y(:,10));
id=y(:,1);
iq=y(:,2);
%THE ELECTROMAGNETIC TORQUE
%EQUATION
Te=1.5.*P.*(x(:,1).*iq-x(:,2).*id);

```

```

%ROTOR SPEED IN Rad/s
wr=y(:,9);
theta=y(:,10);
ia=iq.*cos(theta)+id.*sin(theta);
%MECHANICAL ROTOR SPEED IN rpm
Me=(y(:,9).*60)/(P.*2.*pi);
end
if flag==3%CONVENTIONAL MACHINE
%MODEL WITH SATURATION EFFECT
%D-Q-CURRENTS*****
%i=[iqs ids iqr idr];
I=[y(:,1) y(:,2) y(:,3) y(:,4)];
IT=[y(:,1) y(:,2) y(:,3) y(:,4)];
I1_s=y(:,1)+y(:,3);
I1_r=y(:,2)+y(:,4);
iu=sqrt(I1_s.^2+I1_r.^2);
Lm=(6.4e-2.*iu.^4-0.94.*iu.^3+2.4.*iu.^2-
1.4.*iu+230)*1e-3;
Lr=Llr+Lm;
Ls=Lls+Lm;
%ELECTROMAGNETIC TORQUE**
Te=1.5.*P.*Lm.*(y(:,1).*y(:,4)- y(:,2).*y(:,3));
%STATOR VOLTAGES*****
Vas=sqrt(2)*V.*cos(wb.*t);
Vbs=sqrt(2)*V.*cos(wb.*t-2.*pi/3);
Vcs=sqrt(2)*V.*cos(wb.*t+2.*pi/3);
VQD=[Vas Vbs Vcs]'; % Stator voltages in %Matrix
Form
%D-Q-VOLTAGE TRANSFORMATION
%IN STATIONARY REFERENCE %FRAME
T_Park = (2/3).*[cos(y(:,6)) cos(y(:,6)-(2.*pi/3))
cos(y(:,6)-(4.*pi/3))
sin(y(:,6)) sin(y(:,6)-(2.*pi/3)) sin(y(:,6)-
(4.*pi/3))]; %Park %transform(Non-Power
%invariant form)
theta=(y(:,6));
%D-Q-VOLTAGE TRANSFORMATION
%IN ROTOR REFERENCE FRAME
V_qsds=T_Park*VQD;
Vqs=V_qsds(1,:);
Vds=V_qsds(2,:);
Vq=Vqs;
Vd=Vds;
%THE STATOR PHASE CURRENTS.
%D-Q- TRANSFORMATION
Tinv_Park=[cos(y(:,6)) sin(y(:,6))
cos(y(:,6)-2.*pi/3) sin(y(:,6)-2.*pi/3)
cos(y(:,6)-4.*pi/3) sin(y(:,6)-4.*pi/3)];
iqd=[y(:,1) y(:,2)];
theta=(y(:,6));
Iabc=Tinv_Park*iqd;
ias=y(:,1).*cos(y(:,6))+y(:,2).*sin(y(:,6)); %Phase "a"
current
%MECHANICAL ROTOR SPEED
Me=(y(:,5).*60)/(P.*2.*pi);
end
if flag==1 %PLOTS WITHOUT %SKINEFFECT
figure(1);

```

```

subplot(2,2,1)
plot(Me,Te,'k')
grid on
xlabel('Speed [rpm]');
ylabel('Electromagnetic Torque [N-m]')
subplot(2,2,2)
plot(t,ias,'k')
grid on
xlabel('Time[s]');
ylabel('ias [A]')
subplot(2,2,3)
plot(t,Te,'k')
grid on
xlabel('Time [s]');
ylabel('Electromagnetic Torque [N-m]')
subplot(2,2,4)
plot(t,Me,'k')
grid on
xlabel('Time[s]');
ylabel('Mechanical rotor Speed[rpm]')
end
if flag==2 %PLOTS WITH SKIN-EFFECT
figure(1);
subplot(2,2,1)
plot(Me,Te,'k');
grid on
xlabel('Mechanical rotor Speed[rpm]');
ylabel('Electromagnetic Torque[N-m]');
subplot(2,2,2)
plot(t,ia,'k');
grid on
ylabel('ias[A]');
xlabel('Time[s]');
subplot(2,2,3)
plot(t,Te,'k')
grid on
xlabel('Time [s]');
ylabel('Torque [N-m]')
subplot(2,2,4)
plot(t,Me,'k')
grid on
xlabel('Time[s]');
ylabel('Mechanical rotor Speed[rpm]')
end
plot(t,Te,'k');
grid on
xlabel('Time[s]');
ylabel('Electromagnetic Torque[N-m]');
subplot(2,2,4)
plot(t,Me,'k');
grid on
xlabel('Time[s]');
ylabel('Mechanical rotor Speed[rpm]');
end
if flag==3 %PLOTS WITH %SATURATION
EFFECT
figure(1);
subplot(2,2,1)
plot(Me,Te,'k')
grid on
xlabel('Speed [rpm]');
ylabel('Electromagnetic Torque [N-m]')
subplot(2,2,2)
plot(t,ias,'k'); grid on
hold on
xlabel('Time[s]');
ylabel('ias[A]')
subplot(2,2,3)
plot(t,Te,'k')
grid on
xlabel('Time [s]');
ylabel('Torque [N-m]')
subplot(2,2,4)
plot(t,Me,'k')
grid on
xlabel('Time[s]');
ylabel('Mechanical rotor Speed[rpm]')
end

```