CHAPTER – 4

ROTOR FLUX BASED MRAS ROTOR RESISTANCE ESTIMATOR USING ARTIFICIAL NEURAL NETWORKS

Contents of this chapter has been published in

1. “Performance Enhancement of Vector Controlled Drive with Neural Learning Algorithm based MRAS Rotor Resistance Estimator ” by M.Nandhini Gayathri, S.Himavathi, R.Sankaran in International Review on Modelling and Simulations (Volume.4.No.4, August 2011 - SCOPUS Indexed)

4.1 Introduction

Several techniques of identification of the rotor resistance for use in the RFOC drive have been discussed in Chapter 2. ANNs have the attributes of estimating parameters of non-linear systems with good accuracy and fast response. This chapter presents such an ANN based method of estimation of the rotor resistance for RFOC induction motor drive system. The back propagation neural network technique is used for the real time adaptive estimation of $R_r$. The error between the desired state variable of the induction motor and the actual state variable of a neural model is back propagated to adjust the weights of the neural model, so that the actual state variable tracks the desired value.

In this Chapter the task of tracking the variations in rotor resistance ($R_r$) of a Squirrel-Cage Induction Motor using Artificial Neural Network approach is formulated. A back propagation neural learning algorithm is developed leading to $R_r$ estimation in an iterative manner. The above work is further extended to tackle the problem of maintaining correct field orientation to realize a RFOC induction motor drive system. Since the majority of industrial drives use Squirrel-Cage induction motors, the investigations are confined to Squirrel–Cage Induction Motors.
4.2 Neural Network Architecture and Learning Algorithm

The performance of the Neural Network model depends on the Neural Architecture and Learning Algorithm. The popular neural architectures are feed forward Architecture and Cascaded Architecture. The learning algorithms are used to obtain the optimum parameters (weights and biases) of the network by minimizing the performance index defined in terms of mean square error function. The neural learning algorithms are mainly grouped into two types: first order approach algorithms (based on steep descent method) and second order approach algorithms. The most commonly used first order approach algorithms are Back Propagation algorithm with Momentum (BPM) and Variable Learning Rate Back Propagation algorithm (VLR). The Levenberg- Marquardt (LM) algorithm is the most commonly used second order approach algorithm [35], [36].

Multilayer feed forward neural networks are regarded as universal approximations and have the capability to acquire nonlinear input-output relationships of a system by learning via the back-propagation algorithm as proposed by Funahashi [35] and by Hornik et al [36]. A simple two-layer feed forward neural network trained by the back-propagation technique is employed in the rotor resistance identification. The modified technique using ANN proposed in this chapter is capable of being implemented on-line so that the resistance updates are available instantaneously and there is no convergence issues related to the learning algorithm [37].

Accordingly an alternate method of estimation of $R_r$ is attempted using Artificial Neural Network. The two-layered neural network based on a back-propagation with momentum learning algorithm is used to estimate the rotor resistance. Two models of the induction machine namely the voltage model and the current model is formulated, with
the former serving as a reference entity and the later as the adaptive entity. Simultaneous solution of the two state models for the rotor flux yields an instantaneous error between the desired and actual state variables which can be used for back propagation as shown in Fig.4.1, to adjust the weights of the neural model, so that the output of this model coincides with the actual output. When the training is completed, the weights of the neural network correspond to the parameters in the actual motor. Neural Networks have the ability to learn, so have become attractive tool for process control.

![Block diagram of Rotor Resistance identification using Neural Networks](image)

Fig.4.1 Block diagram of Rotor Resistance identification using Neural Networks
4.3 Neural Learning Algorithm Based MRAS Rotor Resistance Estimation for RFOC using Rotor Flux (NLRF-MRAS)

The basic structure of an adaptive scheme described by Fig.4.1 is extended for rotor resistance estimation of an induction motor as illustrated in Fig.4.2. Two independent observers are used to estimate the rotor flux vectors of the induction motor. Equation (4.1) is based on stator voltages and currents, which is referred as the voltage model of the induction motor. Equation (4.2) is based on stator currents and rotor speed, which is referred as the current model of the induction motor [4], [38]. Back propagation with momentum is used as a learning algorithm for adaptation.

Voltage Model Equations:

\[
\begin{bmatrix}
\frac{d\lambda_{dr}}{dt} \\
\frac{d\lambda_{qr}}{dt}
\end{bmatrix} = \frac{L_r}{L_m} \begin{bmatrix} v_{ds} \\ v_{qs} \end{bmatrix} - \begin{bmatrix} R_s + s\sigma L_s & 0 \\ 0 & R_s + s\sigma L_s \end{bmatrix} \begin{bmatrix} i_{ds} \\ i_{qs} \end{bmatrix} \]

... (4.1)

Current Model Equations:

\[
\begin{bmatrix}
\frac{d\lambda_{dr}}{dt} \\
\frac{d\lambda_{qr}}{dt}
\end{bmatrix} = \begin{bmatrix} -1/T_r & -\omega_r \\ \omega_r & -1/T_r \end{bmatrix} \begin{bmatrix} \lambda_{dr} \\ \lambda_{qr} \end{bmatrix} + \frac{L_m}{T_r} \begin{bmatrix} i_{ds} \\ i_{qs} \end{bmatrix} \]

... (4.2)

From equation (4.2) the derivative terms are replaced by finite differences as follows:

\[
\frac{\lambda_{dr}(k) - \lambda_{dr}(k-1)}{T} = -\frac{1}{T_r} \lambda_{dr}(k-1) - \omega_r \lambda_{qr}(k-1) + \frac{L_m}{T_r} i_{ds}(k-1) \]

... (4.2a)

\[
\frac{\lambda_{qr}(k) - \lambda_{qr}(k-1)}{T} = \omega_r \lambda_{dr}(k-1) - \frac{1}{T_r} \lambda_{qr}(k-1) + \frac{L_m}{T_r} i_{qs}(k-1) \]

... (4.2b)
Accordingly the above difference equations in the form of a sample data system are obtained as:

\[ \lambda_{dr} (k) = W_1 \cdot \lambda_{dr} (k-1) - W_2 \cdot \lambda_{qr} (k-1) + W_3 \cdot i_{ds} (k-1) \quad \ldots (4.3) \]

\[ \lambda_{qr} (k) = W_1 \cdot \lambda_{qr} (k-1) + W_2 \cdot \lambda_{dr} (k-1) + W_3 \cdot i_{qs} (k-1) \quad \ldots (4.4) \]

Where, the symbols \( W_1, W_2 \) and \( W_3 \) are introduced and defined as:

\[
\begin{align*}
W_1 &= 1 - \frac{T}{T_r} \\
W_2 &= \omega_r \cdot T \\
W_3 &= \frac{T \cdot L_m}{T_r}
\end{align*}
\ldots (4.4 a)
\]

The neural network model represented by (4.3) & (4.4) is shown in Fig.4.3, where \( W_1, W_2 \) and \( W_3 \) represent the weights of the network. If the network shown in Fig.4.3 is used to estimate \( R_r \), \( W_2 \) is already known and \( W_1 \) and \( W_3 \) need to be updated.
The weights of the network, $W_1$ and $W_3$ are found from training, so as to minimize the error function $E$,

$$E = \frac{1}{2} \begin{bmatrix} \varepsilon_d(k) & \varepsilon_q(k) \end{bmatrix}$$ … (4.5)

The weight adjustment for $W_1$ using generalized delta rule is given by,

$$\Delta W_1(k) = \eta \left[ \varepsilon_d(k) * \lambda_{dr}^{vm}(k-1) + \varepsilon_q(k) * \lambda_{qr}^{vm}(k-1) \right]$$ … (4.6)

The weight $W_1$ has to be updated using the current weight and the required correction as in (4.6).

Where,

- $\eta$ is the Learning Rate

- $\varepsilon_d(k) = \lambda_{dr}^{vm}(k) - \lambda_{dr}^{nm}(k)$

- $\varepsilon_q(k) = \lambda_{qr}^{vm}(k) - \lambda_{qr}^{nm}(k)$
To accelerate the convergence of the error back propagation learning algorithm, the current weight adjustment is supplemented with a fraction of the most recent weight adjustment, as in equation (4.7).

\[
W_i(k) = W_i(k-1) + \Delta W_i(k) + \alpha \Delta W_i(k-1)
\] ... (4.7)

Where, \(\alpha\) is a user – selected positive Momentum Constant

Similarly the weight change for \(W_3\) can be calculated as follows:

\[
\Delta W_3(k) = \eta \left[ \varepsilon_d(k) I_d(k-1) + \varepsilon_q(k) I_q(k-1) \right]
\] ... (4.8)

Where,

\[
\varepsilon_d(k) = \lambda_{dr}(k) - \lambda_{dr}^{nm}(k)
\]

\[
\varepsilon_q(k) = \lambda_{qr}(k) - \lambda_{qr}^{nm}(k)
\]

As for \(W_3\), in order to accelerate the convergence of the error back propagation learning algorithm, the current weight adjustments are supplemented with a fraction of the most recent weight adjustment, as in equation (4.9).

\[
W_3(k) = W_3(k-1) + \Delta W_3(k) + \alpha \Delta W_3(k-1)
\] ... (4.9)

The value of \(R_r\) is related to the weights \(W_1\) and \(W_3\) in equation (4.4a) by substituting \(T_r = L_r/R_r\). The rotor resistance \(R_r\) can now be calculated from either \(W_3\) from (4.9) or \(W_1\) from (4.7), as follows:

\[
R_r = \left( \frac{L_r * W_3}{L_m * T} \right)
\] ... (4.10)

\[
R_r = \frac{L_r}{T} (1 - W_1)
\] ... (4.11)
4.4 Estimation Results with Squirrel – Cage Induction Motor

The main focus of this thesis is on investigating the rotor resistance identification for squirrel-cage induction motors, which are widely used in the industry. In order to verify the effectiveness and feasibility of estimating rotor resistance using Neural Learning Algorithm based MRAS using rotor flux described in Section 4.3, a simulation model has been developed in MATLAB/SIMULINK platform and the simulation model of the above is given in Fig.4.4. A PWM switching frequency of 5 kHz and DC link Voltage of 650V was used for simulations.

Fig.4.4 MATLAB/SIMULINK Schematic of NLRF-MRAS Based Rotor Resistance Estimator

The Space Vector PWM inverter fed induction motor drive is subjected to various changes in rotor resistance and the tracking capability of the method is tested.
Simulations have been done for various changes in $R_r$ for the operating condition of 415V/50Hz and rated load of 7.5Nm and the performance of rotor resistance estimator has been analysed for 100% Step, Ramp and Trapezoidal changes in $R_r$ and also compared with Rotor Flux based MRAS Rotor Resistance Estimator using PI Controller described in Chapter 3. The simulation for a 100% step change of $R_r$ at 1 sec from an initial value of $6.085\,\Omega$ to an updated value of $12.17\,\Omega$ was carried out for ascertaining the convergence capabilities of the NLRF-MRAS algorithm. The result of this simulation is shown in Fig.4.5. It is recognized that during normal run of a motor under load, a rotor resistance undergoes a gradual change over a period of time. In order to simulate this phenomenon, 100% Ramp change in $R_r$ is given at 0.5 sec is shown in Fig.4.6. The rotor resistance is rising in ramp manner gradually from 0.5sec to 1.7 sec and reaches the value $12.17\,\Omega$ from $6.085\,\Omega$.

![Graph showing actual and estimated rotor resistance for 100% step change in $R_r$.](image)

**Fig.4.5** Actual and Estimated Rotor Resistance for 100% **Step** change in $R_r$
Fig. 4.6 Actual and Estimated Rotor Resistance for 100% **Ramp** change in $R_r$

![Fig. 4.6 Actual and Estimated Rotor Resistance for 100% Ramp change in $R_r$](image)

Fig. 4.7 Actual and Estimated Rotor Resistance for 100% **Trapezoidal** change in $R_r$

![Fig. 4.7 Actual and Estimated Rotor Resistance for 100% Trapezoidal change in $R_r$](image)

Table 4.1 Estimator Error and Settling Time for Various Changes in $R_r$ of NLRF-MRAS

<table>
<thead>
<tr>
<th>Change in $R_r$ (%)</th>
<th>Actual $R_r$ (ohms)</th>
<th>Estimated $R_r$ (ohms)</th>
<th>Settling Time (sec)</th>
<th>Error (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>6.694</td>
<td>6.682</td>
<td>0.03</td>
<td>0.179</td>
</tr>
<tr>
<td>20</td>
<td>7.302</td>
<td>7.294</td>
<td>0.03</td>
<td>0.110</td>
</tr>
<tr>
<td>30</td>
<td>7.910</td>
<td>7.904</td>
<td>0.03</td>
<td>0.076</td>
</tr>
<tr>
<td>40</td>
<td>8.519</td>
<td>8.511</td>
<td>0.04</td>
<td>0.094</td>
</tr>
<tr>
<td>50</td>
<td>9.127</td>
<td>9.117</td>
<td>0.04</td>
<td>0.110</td>
</tr>
<tr>
<td>60</td>
<td>9.736</td>
<td>9.727</td>
<td>0.04</td>
<td>0.092</td>
</tr>
<tr>
<td>70</td>
<td>10.34</td>
<td>10.33</td>
<td>0.05</td>
<td>0.096</td>
</tr>
<tr>
<td>80</td>
<td>10.95</td>
<td>10.94</td>
<td>0.05</td>
<td>0.091</td>
</tr>
<tr>
<td>90</td>
<td>11.56</td>
<td>11.55</td>
<td>0.05</td>
<td>0.087</td>
</tr>
<tr>
<td>100</td>
<td>12.17</td>
<td>12.16</td>
<td>0.05</td>
<td>0.082</td>
</tr>
</tbody>
</table>
The third case above corresponds to the loading profile of an induction motor which is starting from an initial $R_r$ value, undergoes a heat run for a certain interval, reaching a steady temperature rise followed by a cooling period due to reduction of load and switch off. The above pattern is represented as a trapezoidal model for imposing $R_r$ change in a compressed time frame. Fig.4.7 shows the NLRF-MRAS output for 100% Trapezoidal change in $R_r$. The rotor resistance is rising in ramp manner from 0.4sec to 0.8 sec and reaches the value $12.17\,\Omega$ from $6.085\,\Omega$. From 1.4sec to 1.8sec rotor resistance is decreasing gradually in ramp fashion and reaches again $6.085\,\Omega$ from $12.17\,\Omega$. Here, the figures of merit for evaluating the estimator are percentage error in estimation and settling time. Table.4.1 shows the percentage error in estimation between the actual and estimated rotor resistance and settling time for various changes in rotor resistance using NLRF-MRAS and it has been compared with the conventional MRAS.

From the results obtained, it is observed that the maximum percentage error in estimation for NLRF-MRAS is 0.18% and maximum settling time is found to be 0.05sec. From the Table.4.2, it is observed that the percentage error in estimation and settling time of NLRF-MRAS is found to be much less compared to conventional MRAS rotor resistance estimator. It is concluded that the Neural learning based adaptation is suitable for Rotor Flux based MRAS.

### Table.4.2 Comparison Table

<table>
<thead>
<tr>
<th>Methods using Rotor Flux</th>
<th>Max.Percentage Error</th>
<th>Settling Time(sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conventional MRAS</td>
<td>3</td>
<td>0.13</td>
</tr>
<tr>
<td>Neural Learning Algorithm Based MRAS(NLRF-MRAS)</td>
<td>0.18</td>
<td>0.05</td>
</tr>
</tbody>
</table>
A schematic diagram showing the Space Vector PWM inverter, RFOC controller, rotor resistance estimator and the Squirrel Cage induction motor together with the loading arrangement is shown in Fig.4.8. The stator voltages and currents are measured to estimate the rotor flux linkages using the voltage model as shown in this figure. The inputs to the Rotor Resistance Estimator are the stator currents, rotor flux linkages and the rotor speed. The estimated rotor resistance $R_r$ will then be used in the RFOC controllers for the flux model. The parameters of the squirrel–cage induction motor used are given in Appendix C.

4.5 Performance Evaluation of Vector Controlled Induction Motor Drive with Neural Learning Algorithm based MRAS Rotor Resistance Estimator using Rotor Flux (NLRF-MRAS)

After establishing the validity of the Rotor Resistance Estimation described in Section 4.3 with the Squirrel-Cage Induction Motor, investigations were carried out for Vector Controlled Induction Motor Drive in MATLAB/SIMULINK platform and the simulation model of the above is given in Fig.4.9. The Vector Controlled drive consists of Space Vector PWM inverter, RFOC controller, Rotor Resistance Estimator and the Squirrel Cage induction motor together with the loading arrangement is shown in Fig.4.8. The sampling time used for the speed controller is $200\mu \text{seconds}$, and it is $20\mu \text{seconds}$ for the current controller. The sampling time of the rotor resistance estimator was $100\mu \text{seconds}$.

The performance of the Vector Controlled Induction Motor drive is analyzed for the following operating conditions:
Operating Condition 1:

- Reference Speed = 100 rad/sec
- Reference Rotor Flux = 0.9Wb
- Load Torque = 7.5Nm
- 100% Step change in Rotor Resistance is given at 1 sec.

The performance of the Vector Controlled Induction Motor drive system was analyzed by applying an abrupt (100% step) change in $R_r$ of motor, from $6.085\Omega$ to $12.17\Omega$ at 1 second. Fig.4.10 (a) shows the actual and reference d-axis Rotor Flux when the rotor resistance used in the RFOC controller was kept unaltered. From the result it is observed that without Rotor Resistance Estimator, the actual rotor flux deviates from the reference rotor flux for 100% step change in rotor resistance at 1sec. Subsequently simulation was repeated after enabling the rotor resistance estimator block, so that the rotor resistance in the controller was updated with the estimated rotor resistance. The modeling results obtained for this case is shown in Fig.4.10 (b). From the result it is observed that with NLRF-MRAS based Rotor Resistance Estimator, the actual rotor flux is tracking the reference rotor flux. The rotor flux increase was only due to the fact that a linear magnetic circuit was assumed without magnetic saturation.
Fig. 4.8 Block Diagram of the RFOC Squirrel–Cage Induction Motor Drive with NLRF – MRAS Based on-line Rotor Resistance tracking
Fig. 4.9 MATLAB/SIMULINK schematic of the RFOC Squirrel-Cage Induction Motor Drive with NLRF-MRAS based on-line Rotor Resistance tracking
Fig. 4.10 Actual and Reference d-axis Rotor Flux (a) Without NLRF-MRAS (b) With NLRF-MRAS

Fig. 4.11 (a) shows the q-axis Rotor Flux when the rotor resistance used in the RFOC controller was kept unaltered. From the result it is observed that without Rotor Resistance Estimator, the q-axis rotor flux is not zero for 100% step change in rotor resistance at 1sec indicating the absence of field orientation. Subsequently simulation was repeated after enabling the rotor resistance estimator block, so that the rotor resistance in the controller was updated with the estimated rotor resistance. The modeling results obtained for this case is shown in Fig.4.11 (b). From the result it is observed that with NLRF-MRAS based Rotor Resistance Estimator, the q-axis rotor flux is zero indicating field orientation even after 100% Step change in Rotor Resistance at 1 sec.
Fig.4.11 q-axis Rotor Flux (a) Without NLRF-MRAS (b) With NLRF-MRAS

Fig.4.12 (a) shows the actual and reference speed when the rotor resistance used in the RFOC controller was kept unaltered. From the result it is observed that without NLRF-MRAS based Rotor Resistance Estimator, the actual speed deviates from the reference speed and takes a long time to track the reference speed for 100% step change in rotor resistance at 1sec. Subsequently simulation was repeated after enabling the rotor resistance estimator block, so that the rotor resistance in the controller was updated with the estimated rotor resistance. The modeling results obtained for this case is shown in Fig.4.12 (b). From the result it is observed that with NLRF-MRAS based Rotor
Resistance Estimator the actual speed is tracking the reference rotor speed within a short period.

Fig.4.13 (a) shows the electromagnetic torque developed when the rotor resistance used in the RFOC controller was kept unaltered. From the result it is observed that without NLRF-MRAS based Rotor Resistance Estimator, the controller is slightly failed to control the torque for 100% step change in rotor resistance at 1sec. Subsequently simulation was repeated after enabling the rotor resistance estimator block, so that the rotor resistance in the controller was updated with the estimated rotor resistance. The simulation results obtained for this case is shown in Fig.4.13 (b). From the result it is observed that with NLRF-MRAS based Rotor Resistance Estimator, the instantaneous torque control is achieved.
Fig. 4.12 Actual and Reference Speed (a) Without NLRF-MRAS (b) With NLRF-MRAS

Fig. 4.13 Electromagnetic Torque (a) Without NLRF-MRAS (b) With NLRF-MRAS
Fig.4.14 q-axis Stator Current (a) Without NLRF-MRAS (b) With NLRF-MRAS

Fig.4.14 (a) shows the q-axis stator current when the rotor resistance used in the RFOC controller was kept unaltered. From the result it is observed that without NLRF-MRAS based Rotor Resistance Estimator, the q-axis stator current deviates from the reference value for 100% step change in rotor resistance at 1 sec. Subsequently simulation was repeated after enabling the rotor resistance estimator block, so that the rotor resistance in the controller was updated with the estimated rotor resistance. The simulation results obtained for this case is shown in Fig.4.14 (b). From the result it is observed that with NLRF-MRAS based Rotor Resistance Estimator the q-axis stator
current is tracking the rated value. The current \( i_{qs} \) was also found to follow a similar profile of the motor torque.

**Operating Condition 2:**

- Reference Speed = 100 rad/sec
- Reference Rotor Flux = 0.9Wb
- Load Torque = 7.5Nm
- 100% **Ramp change** in Rotor Resistance is given at 1 sec.

The rate of temperature rise for the Squirrel-Cage Induction Motor is also very slow. The change in rotor resistance due to temperature rise is also very slow. To investigate this situation, a simulation was also carried out introducing a 100% ramp change in the rotor resistance. The performance of the drive system was analyzed by changing the rotor resistance by 100%, where, the \( R_r \) was increased from 6.085Ω to 12.17Ω in a ramp fashion.
Fig. 4.15 Actual and Reference d-axis Rotor Flux (a) Without NLRF-MRAS (b) With NLRF-MRAS

Fig. 4.16 q-axis Rotor Flux (a) Without NLRF-MRAS (b) With NLRF-MRAS
Fig.4.15 (a) shows the actual and reference d-axis Rotor Flux when the rotor resistance used in the RFOC controller was kept unaltered. From the result it is observed that without Rotor Resistance Estimator, the actual rotor flux deviates from the reference rotor flux for 100% Ramp change in rotor resistance at 1sec. Subsequently simulation was repeated after enabling the rotor resistance estimator block, so that the rotor resistance in the controller was updated with the estimated rotor resistance. The modeling results obtained for this case is shown in Fig.4.15 (b). From the result it is observed that with NLRF-MRAS based Rotor Resistance Estimator, the actual rotor flux is tracking the reference rotor flux.

Fig.4.16 (a) shows the q-axis Rotor Flux when the rotor resistance used in the RFOC controller was kept unaltered. From the result it is observed that without Rotor Resistance Estimator, the q-axis rotor flux is not zero for 100% Ramp change in rotor resistance at 1sec indicating the absence of field orientation. Subsequently simulation was repeated after enabling the rotor resistance estimator block, so that the rotor resistance in the controller was updated with the estimated rotor resistance. The modeling results obtained for this case is shown in Fig.4.16 (b). From the result it is observed that with NLRF-MRAS based Rotor Resistance Estimator, the q-axis rotor flux is zero indicating field orientation following a 100% Ramp change in Rotor Resistance at 1 sec.
Fig. 4.17 Actual and Reference Speed (a) Without NLRF-MRAS (b) With NLRF-MRAS

Fig. 4.17 (a) shows the actual and reference speed when the rotor resistance used in the RFOC controller was kept unaltered. From the result it is observed that without NLRF-MRAS based Rotor Resistance Estimator, the actual speed deviates from the reference speed and takes a long time to track the reference speed after 100% Ramp change in rotor resistance at 1 sec. Subsequently simulation was repeated after enabling the rotor resistance estimator block, so that the rotor resistance in the controller was updated with the estimated rotor resistance. The modeling results obtained for this case is shown in Fig. 4.17 (b). From the result it is observed that with NLRF-MRAS based Rotor
Resistance Estimator the actual speed is tracking the reference rotor speed within a short period.

Fig. 4.18 Electromagnetic Torque (a) Without NLRF-MRAS (b) With NLRF-MRAS

Fig. 4.18 (a) shows the electromagnetic torque developed when the rotor resistance used in the RFOC controller was kept unaltered. From the result it is observed that without NLRF-MRAS based Rotor Resistance Estimator, the controller is slightly failed to control the torque for 100% Ramp change in rotor resistance at 1sec. Subsequently simulation was repeated after enabling the rotor resistance estimator block, so that the rotor resistance in the controller was updated with the estimated rotor resistance. The
simulation results obtained for this case is shown in Fig.4.18 (b). From the result it is observed that with NLRF-MRAS based Rotor Resistance Estimator, the instantaneous torque control is achieved.

![Graph](image)

**Fig.4.19** q-axis Stator Current (a) Without NLRF-MRAS (b) With NLRF-MRAS

Fig.4.19 (a) shows the q-axis stator current when the rotor resistance used in the RFOC controller was kept unaltered. From the result it is observed that without NLRF-MRAS based Rotor Resistance Estimator, the q-axis stator current deviates from the reference value for 100% Ramp change in rotor resistance at 1sec. Subsequently simulation was repeated after enabling the rotor resistance estimator block, so that the
rotor resistance in the controller was updated with the estimated rotor resistance. The simulation results obtained for this case is shown in Fig.4.19 (b). From the result it is observed that with NLRF-MRAS based Rotor Resistance Estimator the q-axis stator current is tracking the rated value.

**Operating Condition 3:**

- Reference Speed = 100 rad/sec
- Reference Rotor Flux = 0.9Wb
- Load Torque = 7.5Nm
- 100% Trapezoidal change in Rotor Resistance is given at 1 sec.

Finally, the algorithm is tested for a cyclic change in $R_r$ in the form of a trapezoidal function. Fig.4.20 (a) shows the actual and reference d-axis Rotor Flux when the rotor resistance used in the RFOC controller was kept unaltered. From the result it is observed that without Rotor Resistance Estimator, the actual rotor flux deviates from the reference rotor flux for 100% Trapezoidal change in rotor resistance at 1 sec. Subsequently simulation was repeated after enabling the rotor resistance estimator block, so that the rotor resistance in the controller was updated with the estimated rotor resistance. The simulation results obtained for this case is shown in Fig.4.20 (b). From the result it is observed that with NLRF-MRAS based Rotor Resistance Estimator, the actual rotor flux is tracking the reference rotor flux.

Fig.4.21 (a) shows the q-axis Rotor Flux when the rotor resistance used in the RFOC controller was kept unaltered. From the result it is observed that without Rotor Resistance Estimator, the q-axis rotor flux is not zero for 100% Trapezoidal change in rotor
resistance at 1 sec indicating the absence of field orientation. Subsequently simulation was repeated after enabling the rotor resistance estimator block, so that the rotor resistance in the controller was updated with the estimated rotor resistance. The simulation results obtained for this case is shown in Fig.4.21 (b). From the result it is observed that with NLRF-MRAS based Rotor Resistance Estimator, the q-axis rotor flux is zero indicating field orientation for 100% Trapezoidal change in Rotor Resistance at 1 sec.

![Graph](image1)

![Graph](image2)

**Fig.4.20** Actual and Reference d-axis Rotor Flux (a) Without NLRF-MRAS (b) With NLRF-MRAS
Fig.4.21  q-axis Rotor Flux (a) Without NLRF-MRAS (b) With NLRF-MRAS

Fig.4.22 (a) shows the actual and reference speed when the rotor resistance used in the RFOC controller was kept unaltered. From the result it is observed that without NLRF-MRAS based Rotor Resistance Estimator, the actual speed deviates from the reference speed and takes a long time to track the reference speed after 100% Trapezoidal change in rotor resistance at 1sec. Subsequently simulation was repeated after enabling the rotor resistance estimator block, so that the rotor resistance in the controller was updated with the estimated rotor resistance. The simulation results obtained for this case is shown in Fig.4.22 (b). From the result it is observed that with NLRF-MRAS based
Rotor Resistance Estimator the actual speed is tracking the reference rotor speed within a short period.

Fig.4.22  Actual and Reference Speed (a) Without NLRF-MRAS (b) With NLRF-MRAS

Fig.4.23 (a) shows the electromagnetic torque developed when the rotor resistance used in the RFOC controller was kept unaltered. From the result it is observed that without NLRF-MRAS based Rotor Resistance Estimator, the controller is slightly failed to control the torque for 100% Trapezoidal change in rotor resistance at 1sec. Subsequently simulation was repeated after enabling the rotor resistance estimator block, so that the rotor resistance in the controller was updated with the estimated rotor
resistance. The modeling results obtained for this case is shown in Fig.4.23 (b). From the result it is observed that with NLRF-MRAS based Rotor Resistance Estimator, the instantaneous torque control is achieved.

Fig.4.23 Electromagnetic Torque (a) Without NLRF-MRAS (b) With NLRF-MRAS

Fig.4.24 (a) shows the q-axis stator current when the rotor resistance used in the RFOC controller was kept unaltered. From the result it is observed that without NLRF-MRAS based Rotor Resistance Estimator, the q-axis stator current deviates from the reference value for 100% Trapezoidal change in rotor resistance at 1sec. Subsequently simulation was repeated after enabling the rotor resistance estimator block, so that the rotor resistance in the controller was updated with the estimated rotor resistance. The
modeling results obtained for this case is shown in Fig. 4.24 (b). From the result it is observed that with NLRF-MRAS based Rotor Resistance Estimator the q-axis stator current is tracking the reference value.

Fig. 4.24  q-axis Stator Current (a) Without NLRF-MRAS (b) With NLRF-MRAS

Operating Condition 4:

The variation in stator resistance affects the flux based MRAS as flux estimation is dependent on $R_s$. This is demonstrated using the following operating condition.

- Reference Speed = 100 rad/sec
- Reference Rotor Flux = 0.9Wb
- Load Torque = 7.5Nm
- 100% **Ramp change** in **Rotor Resistance** is given at 1 sec.
- 100% **Step Change** in **Stator Resistance** is given at 2.5 sec.

(a)

(b)
Fig.4.25 (a) Actual and Estimated Rotor Resistance (b) Actual and Reference d-axis Rotor Flux (c) q-axis Rotor Flux (d) Actual and Reference Speed (e) Electromagnetic Torque (f) q-axis Stator Current

Fig.4.25 (a) shows the NLRF-MRAS based rotor resistance estimation output for 100% Ramp change in $R_r$ and 100% Step change in $R_s$. The Rotor Resistance is rising in ramp manner gradually from 1 sec to 2.2sec and reaches the value from 12.17Ω from 6.085Ω. With NLRF-MRAS based $R_r$ estimator, the estimated rotor resistance deviates from the actual rotor resistance value when a step change in stator resistance is detected at 2.5sec.

Fig.4.25 (b) shows the d-axis rotor flux, after enabling the rotor resistance estimator block, so that the rotor resistance in the RFOC controller was updated with the estimated rotor resistance. From the result it is observed that, the actual d-axis rotor flux deviates from the reference rotor flux when a step change in $R_s$ is given at 2.5sec.

Fig.4.25 (c) shows the q-axis rotor flux, after enabling the rotor resistance estimator block, so that the rotor resistance in the RFOC controller was updated with the estimated
rotor resistance. From the result it is observed that, the q-axis rotor flux is not zero indicating that field orientation is lost when a step change in $R_s$ is given at 2.5sec.

Fig.4.25 (d) shows the actual and reference speed, after enabling the rotor resistance estimator block, so that the rotor resistance in the RFOC controller was updated with the estimated rotor resistance. From the result it is observed that, the actual speed is getting deviated from the reference speed when a step change in $R_s$ is given at 2.5sec.

Fig.4.25 (e) shows the Electromagnetic torque developed, after enabling the rotor resistance estimator block, so that the rotor resistance in the RFOC controller was updated with the estimated rotor resistance. From the result it is observed that, controller is slightly failed to control the torque when a step change in $R_s$ is given at 2.5sec.

Fig.4.25 (f) shows q-axis stator current, after enabling the rotor resistance estimator block, so that the rotor resistance in the RFOC controller was updated with the estimated rotor resistance. From the result it is observed that the q-axis stator current deviates from the reference value when a step change in $R_s$ is given at 2.5sec, confirming loss of field orientation.
4.6 Conclusion

The Neural Learning Algorithm is proposed and implemented for adaptation in RF-MRAS based Rotor Resistance Estimator. Among the different neural learning algorithms the most suitable algorithm for on-line estimation is back-propagation with momentum. The NLRF-MRAS based Rotor Resistance Estimator was designed, simulated and analysed. From the simulation results obtained, it is observed that the percentage error in estimation of NLRF-MRAS is 0.18% and settling time is found to be 0.05sec. The percentage error in estimation and settling time of NLRF-MRAS is found to be less compared with Conventional MRAS Rotor Resistance Estimator described in Chapter 3. The Neural Learning Algorithm based MRAS Rotor Resistance Estimator using Rotor Flux is found to be more accurate and the corresponding performance enhancement achieved in Vector Controlled drive is presented.

As the proposed MRAS is flux based, the governing equations depend on stator resistance which also varies with operating conditions. Hence a separate estimator is required for stator resistance estimation which in turn increases the complexity of the control algorithm for the drive.

To make Rotor Resistance Estimation independent of Stator Resistance a Reactive Power based MRAS Rotor Resistance Estimator is proposed and investigated in the next Chapter.