International Journal of Current Engineering and Technology
ISSN 2277 - 4106

Available at http://inpressco.com/category/ijcet

Research Article

Noninvasive External Faults Detection of Induction Motor using Feedforward Neural Network

Kalpesh J. Chudasama[*a] and Vipul Shah[b]

[a]Department of Electrical Engineering, ADIT, Research Scholar, Gujarat Technological University, Gujarat, India.
[b]Department of I & C, D.D.University, Gujarat, India.

Accepted 10 March 2013, Available online 1 June 2013, **Vol.3, No.2(June 2013)**

Abstract

Induction motors are come across various abnormalities and faults categorized as external and internal faults. This paper discusses the performance of the Feedforward Neural Network (FFNN) as non invasive fault detection technique for the external faults/abnormalities in induction motor. Alongwith normal condition various external faults like Overload (O/L), Overvoltage (O/V), Undervoltage (U/V), Stalling, opening of any stator phase, and voltage unbalances are simulated on Squirrel cage induction motor and FFNN train and tested using Levenberg-Maquardt (LM) Backpropagation algorithm using Matlab/Simulink. The various external faults are produced with wide range of operating three phase voltages which may exist in field for the find suitability of Artificial Neural Network (ANN) as fault detector. The main objective is to detect such external fault conditions correctly and also to find best ANN configuration. Number of ANN configurations are trained, parallel validated and tested for test (unseen) data with different combination like Early Stopping (ES) or Bayesian Regularization (BR) as generalization technique, Tansig or Purelin as activation performance function in the output layer and different processing technique of data. Good result of each configuration is shown and best among them found. 100% stastical parameters such as total classification accuracy, sensitivity and specificity for both test (unseen) and input data obtained for best Neural Network (NN) configuration. The best NN configuration found with BR as generalization, Purelin as activation function in output layer and data preporcessed with principle component analysis (PCA) after mapping each row of data mapped to have zero mean and unity Standard Deviation (SD). All kind of voltage unbalance like 1phase (ø) O/V, 2ø O/V, 1ø U/V, 2ø U/V, 3ø O/V, 3ø U/V and phase displacement is considered in voltage unbalance case and opening of any phase condition considered in stator open phase. Stalling at start detected within first few cycles which can eliminate the need of waiting for safe stalling time like in conventional protection.

Keywords: *Induction Motor, Neural Network (NN), Early stopping, Feedforward Neural Network, Backpropagation, Bayesian Regularization, Standard deviation, Principle component analysis , root means square(RMS)*

1. Introduction

Induction motors cater major industrial load and consumes a major part of overall electrical consumption. Exposure to wide variety of environments and conditions, improper operations, improper or insufficient protective system and manufacturing defects can make it subject to incipient faults or gradual deterioration and can lead to motor failure if not detected. Sometime a small HP motor failure can also create hours of plant stoppage in continuous processing industries. Correct noninvasive fault detection and classification in electrical machines is the major need of industry. Induction motors appears to different kind of faults or abnormalities which can majorly divided in two parts external and internal faults. Undervoltage, overvoltage, phase failure, unbalance supply, mechanical overload, locked rotor, earth fault in motor supply cable can consider as external faults and stator interturn failure, bearing, rotor faults, eccentricity consider as internal faults (Chudasama& Shah 2012).

There are invasive and noninvasive methods for fault detection. The noninvasive methods are more preferable because they are based on easily accessible and inexpensive measurements to diagnose the machine conditions without disintegrating the machine (Aroui et al., 2008, Anderson, Chow et al., 1993). These schemes are suitable for on-line monitoring and fault detection purposes (Chow et. Al,1993). ANN is proposed for fault identification and other power system applications (Kolla&Varatharsa, 2000, Tallam et al., 2003)& is emerging technologies promising for future widespread industrial usage (Chow et al., 1993). An overview of feedforward netsand the backpropagationtraining algorithm and a generalmethodology for the design of

*Corresponding author: Kalpesh J. Chudasama

feedforwardartificial neural networks to perform motor fault detection is discussed in (Chow et al., 1993). Feedfoward layered network structure and backpropagation algorithm used in ANN technique applied for identification of external faults of induction motor by (Kolla&Vartharsa, 2000, Tag Eldin et al., 2007). Three phase RMS current and voltages from the induction motor simulation are used in approach. Speed is the parameter can be utilized for discriminating starting stalling condition against normal starting condition. Simulated measurements are used to train ANN for External faults classification. Applicability of ANN for external fault detection and protection is discussed in (Tag Eldin et al., 2007). Results are discussed using waveform for respective condition. Siddeque(2004) used simulated faults and radial basis function Neural Network for mainly detects and classifies different voltage unbalance external fault condition. Instantaneous values of faults are taken and shown network classified the faults correctly. Calculation of simple statistical parameters such as the overall RMS value of a signal can give useful information. For instance, the RMS value of the vibration velocity is a convenient measure of the overall vibration severity (Sin et al., 2003, Riley et al., 2003). In the same way, the RMS value of the stator current provides a rough indication of the motor loading. Protection scheme for incipient faults using Microcontroller established in (Sudha&Anbalagan, 2009) using dynamic modeling of induction machine.

Limit values are entered for incipient faults and when unexpected situation occurs it trip contactor. Saied (2012) used ANN for detecting unbalanced faults for traction motor irrespective of the load and fault percentage.

This paper major contribution is external fault detection of induction motor including major kind of voltage unbalance using ANN for varied operating condition and finding the best generalized ANN configuration with minimum hidden node so it can capable to detect any unseen external fault condition with 100% classification accuracy, sensitivity and specificity. Various faults are simulated on symmetrical induction machine and RMS vaules of voltages and currents alongwithinstantenous angular speed are used to train FFNN. To find best generalized FFNN configuration we have taken ES and BR for generalization, purelin and tansig as output layer activation function and for data processing mapping of each data row with zero mean and unity SD with and without PCA. We have tested several NN configurations with different combination of generalization, output layer processing function and data processing method. We have found best configuration after comparing each good configuration obtained from different combination. The FFNN network validated in parallel with batch training and tested with unseen data.

The ANN trains with wide range of voltage (±10%) in case of normal condition and beyond it for low voltage and high voltage. The ANN trains to detect unbalance for more than 1% as voltage unbalance condition and lower as normal condition. One phase, two phase and three phase overvoltage and undervoltage unbalance considered along with phase angle displacement in one and two phase considered in voltage unbalance condition. It can also detect open phase condition occurred in any phase quickly. The advantage of ANN technique is it can identify starting stalling (locked rotor) condition very fast without waiting for safe stalling time unlike conventional relays. We have used LM backpropagation algorithm to train single hidden layer neural networks.

2. ANN Suitability for Fault Detection

The noninvasive parameter estimation requires knowledge of detail mathematical model and parameters are usually difficult to obtain (Chow et al., 1993). With proper monitoring and fault detection scheme incipient fault can be detect with noninvasive procedure using ANN and no mathematical model is needed(Goode & Chow, 1995). FFNN and various backpropagation algorithms have been proposed for on-line neural network learning algorithm for time varying inputs (Xingzuo et al., 2002).

3. ANN Training Algorithms

The core idea in back propagation learning of multi-layer networks for non-liner activation function is that errors for the units of the hidden layer are determined by back propagating of the output layer units.

3.1 Gradient Decent Backpropagation Algorithms

Standard backpropagation is a gradient descent algorithm, as is the Widrow-Hoff learning rule, in which the network weights are moved along the negative of the gradient of the performance function. It works on

$$x_{k+1} = x_k - \alpha g_k \quad (1)$$

Where x_k is vector of current weights is, g_k is the current gradient and α is the learning rate (Rumelheart et al.,1986). In batch mode training all the inputs are applied to the network and the gradients calculated at each training example are added together to determine the change in the weights and biases before the weights are updated in the direction of the negative gradient of mean square error. Standard Gradient Decent (GD) algorithm is too slow for practical problems and can stick to local minima. There are a number of variations of standard GD algorithm. Several high-performance algorithms that can converge faster than the standard algorithm like GD with momentum, variable learning rate and resilient backpropagation categorized developed form analysis of the performance of standard steepest descent algorithm. Most adaption techniques perform weight update by modification of learning rate according to some function of error.

3.2 Levenberg-Marquardt Algorithm

As in (Rowies, 1999)Levenberg proposed algorithm is the blend of GD and Newton technique and the weight update rule is

$$x_{k+1} = x_k - (H + \lambda I)^{-1} g \quad (2)$$

Where g is the average error gradient, I is the identity matrix, λ is weight damping factor, and H is approximation to the hessian (matrix of mixed partials) which is obtained by averaging outer products of the first order derivative (gradient). Steepest Decent type method used until the approach a minimum and gradual switch over to the quadratic approximation. The algorithm adjusts λ according to wheather error increasing or decreasing. If error increases as a result the update then retracts the step means reset the weights to previous values and increase λ by set increasing factor and try for an update again. The step accepted if error decreases as result of update. Marquardt improved this method by replacing the identity matrix (I) by diagonal of the Hessian with a view insight to use benefit of hessian when λ is high, by scaling each component of the gradient according to the curvature, results in larger movement along the directions where the gradient is smaller so the classic error valley does not occur. The resulting update rule is

$$x_{k+1} = x_k - (H + \lambda diag(H))^{-1} g \quad (3)$$

For moderately sized models (few hundred parameters) LM is much faster than gradient descent (Ranganathan, 2004).

4. Generalization

NN learns during learning to approximate behaviour adaptively from training data while generalization is the ability to predict well beyond the training data. Overfitting in complex model such as NN can occurs when a model begins to memorize training data rather than learning to generalize from trend (Peter, 2000). We have used early stopping and bayseian regularization to improve generalization of NN and to make it robust.

4.1 Early Stopping

The usual approach for evaluating the generalization performance of an ANN is to divide the available data into three subsets training, validation and testing (Hessami et al., 2004). The second subset is the validation set. In early stopping error computed for a validation set at the same time that the network is being trained with the training set. ES is performed to avoid the case when the MSE might decrease in the training set but increases in the validation set. This happens when the network starts memorizing the training patterns. Thus network must be instructed to stop the training when the above situation occurs. When the validation error increases for a specified number of iterations the training is stopped, and the weights and biases at the minimum of the validation error are returned. Test subset is independent to check network generalization (Matlab, 2010)

4.2 Bayseian Regularization

The Regularization involves modifying the performance function, which is normally chosen to be the sum of squares of the network errors on the training set and given as

$$F_e = \frac{1}{n}\sum_{i=1}^{n}(t_i - a_i)^2 \quad (4)$$

Bayseian regularization modifies the performance function F by adding an additional term, which consists of the mean sum of squares of the network weights F_w

$$F = \beta F_e + \alpha F_w \quad (5)$$

Where α and β are Bayesian hyperparameters and Fe and Fw are sum of squared errors and sum of squared weights. By using Bayesian regularization costly cross validation can be avoided and need of testing can be eliminate or reduce. The Bayesian regularization proposed by (Mackay, 1992) automatically set optimum values for objective function parameters. The LM algorithm based on bayseian regularization produce a smooth network at the expense of sum square error (SSE) of network.

5. Induction Motor External Faults and simulation

A three phase Induction Motor simulation have been tested and three phase RMS line voltages, RMS line currents and angular speed dataare obtained for the external faults to get the input vector set to train ANN for Induction Motor external fault diagnosis which can generate trip/alarm. The fault simulation is prepared in MATLAB/SIMULINK using 3 phase, 4KW, 5.4 HP, 1430 rpm, 50Hz with star connected winding squirrel cage induction motor. Data obtained using ode 23tb stiff solver. The advantage of ANN is it can be modified in future if require and also FFNN have capability of on line learning. To create real field condition the data obtained with nominal and also at varied different voltages.

5.1 Normal condition

Induction motor normal operation data are obtained with rated load torque and also with some other normal variant loading (60-105% of normal load torque) condition and different normal balanced voltage (±10% of normal voltage) with which motor mostly operates in industry.

5.2 Overload

As the Mechanical load on Induction Motor increases the motor begins to draw high current and speed decrease. After certain amount of load heat generation rate is higher than heat dissipation rate than the insulation threatened. We have taken above 105% to 150% of normal load as overload condition. Alarm/trip can be initiated when motor overloaded.

5.3 Overvoltage

Induction Motor designed to withstand overvoltage upto +10% as general voltage design motor manufacture specification. When voltage increases beyond it motor

overheat because of increase in core losses and it should be stopped.

5.4 Undervoltage

As the voltage across motor reduces motor speed drops and current increases. Low voltage may prevent a motor from starting properly due to the reduced torque at low voltage. The current changes drastically as voltage reduces below 75 to 80% of rated voltage. We have considered 90% of rated voltage as undervoltage cut off. It can be also categorized in two parts below 90% alarm and below 75% instantaneous tripping.

5.5 Stalling at start

If there is a heavy load on the motor when it is started, the motor cannot start and rotor gets blocked (BhuvaneshOza et. Al, 2010) and in this condition also the motor is said to be stall. Data are drawn up from the simulation to detect locked rotor at end of second cycle. As the stalling current which is equal to starting current speed is the only be a distinguish parameter.

5.6 Stator Open Phase

Single phasing is the worst case of voltage unbalance and can be happened because of open circuit fault or in case of one pole open of circuit Breaker or contactor. Data for open phase in each phase is retrieved using circuit breaker one pole open after steady state condition reached within two cycles after stator open phase occurs.

5.7 Voltage Unbalance

Presence of small voltage Unbalance result in large current unbalance by a factor of six times and negative sequence phase components cause increased rotor loss, overheating , reduction in output torque and efficiency. Neema standard suggest no derating required up to 1% unbalance, from 1 to 5% motor derating require and above 5% operation is not recommended (Cummings et al., 1985). Standard motor are capable of operating under condition of supply voltage unbalance of 1% for long period. We have considered voltage unbalance more than 1% for alarm/trip in case of voltage unbalance condition. All types of voltage unbalance like single phase and two phase undervoltage and overvoltage unbalance, three phase undervoltage and overvoltage unbalance and one phase, two phase angle displacement considered in the case. Percentage line unbalance considered based on NEMA definition

% Line Unbalance Voltage Ratio = (Maximum Voltage from average line voltage magnitude / Average Voltage) x 100% (6)

The magnitude of the NEMA unbalanced voltage in percentage and negative sequence voltage in percent is almost equivalent for all practical purpose (Cummings et al., 1985).

6. ANN Architecture and Training

FFNN is used for this fault diagnostic study. The architecture of NN specifies the neuron connection. The data obtained from the induction motor fault simulation of three phase voltages, line currents and speed are used as concurrent input training vector to train the neural network. The input data matrix should be preprocessed for the efficient and better form of NN training. The goal of normalization is to ensure that the statistical distribution of values for each net input and output is roughly uniform.
We have scaled network inputs and targets so that minimum and maximum value of each row mapped to have zero mean and unity standard deviation (SD) using

$$y = (X - Xmean) * \frac{ystd}{xstd} + Ymean \quad (7)$$

We have also further preporcessed the input for some configuration using PCA as feature extraction, which orthogonializes the components of input vector and order the resulting orthogonal components such that largest variation comes first and eliminates the least contributed variation components from dataset. We have used Matlab inbuilt functions for above discussed scaling processing techniques and NN toolbox for the ANN training. Each input is weighted with randomly initialized weights and bias. The sum of weighted inputs and the bias forms the input to transfer function. Tansig transfer function is utilized in hidden layer. The output a^i of ith neuron of hidden layer can be given as

$$a^i = \text{tansig}\,(I.W * p + b) \quad (8)$$

Wherein I.W weight vector of different element of input vector p to i the hidden layer neuron and b^i is weight assigned to neuron. Same as the equation can be assigned for output layer. We have also used tansig and purelin as activation function for output layer. We have used 60% data for ANN training, 20% for validation and 20% for testing purpose out of total 194 input data. After training ANN configurations are tested with 51 test (unseen) data. The detail is mention in Table1. Table 2 shows some of the unseen test data used for testing. Table 3 shows the target assignment for ANN.

Table 1 Train and Test Data used for training and testing

Sr. No.	Condition	Train Data	Test Data
1	Normal (N)	31	6
2	Overload (O/L)	19	6
3	Overvoltage(O/V)	30	6
4	Undervoltage(U/V)	20	5
5	Stalling at start (S)	20	5
6	Stator Open Phase (SOP)	24 (8 Nos. for each phase)	15 (5 Nos. for each phase)
7	Voltage Unbalance Condition (VU/B)	49	8

Table 2 Test input (unseen data) to neural networks

Condition	Vry	Vyb	Vbr	Ir	Iy	Ib	ω
N	405.1	405.4	405.6	7.74	7.72	7.73	150.9
N (U/V <10 %)	369.7	369.9	370	8.16	8.16	8.16	149.4
O/L	399.7	400	400	10.33	10.33	10.33	147.8
O/V	443.2	443.4	443.5	7.44	7.45	7.44	152.1
U/V	347.6	347.8	348	8.55	8.54	8.54	148.3
S	403.7	404.1	404.5	51.61	51.53	51.54	0
SOP(Rphase)	300.2	412.7	374.6	0	16.26	16.26	150.1
SOP (y phase)	370.6	265.7	400.4	20.1	0	20.1	150.8
VU/B (2ø U/V)	365.9	389.9	355.5	6.4	11.04	8.02	148.1
VU/B (3ø O/V)	423.6	438.5	433.8	6.37	7.8	8.52	151.6
VU/B (2ø angle disp.)	407.9	405.8	385.2	9.67	6.14	8.01	151.7

Table 3 Target assignment table for faults

N	O/L	O/V	U/V	S	SOP	VU/B
0	0	0	0	0	0	1
0	0	0	0	0	1	0
0	0	0	0	1	0	0
0	0	0	1	0	0	0
0	0	1	0	0	0	0
0	1	0	0	0	0	0
1	0	0	0	0	0	0

Table 4 Details of good NN configurations obtained with LM algorithm, particular generalization method (ES or BR), output layer processing function (tansig or Purelin) and data processing techniques

NN confguration No.	Hidden neuron	Algorithm/ Generalization/preprocessing	Performance goal obtained	Activation Function for hidden and o/p layer	% Total classification accuracy for input data	% Total classification accuracy for test data
NN1	14	LM/ ES/MAPSTD	0.16	Tansig Tansig	97.4	100
NN2	14	LM/ ES/MAPSTD	0.0889	Tansig Purelin	98	98
NN3	14	LM/BR/MAPSTD	156	Tansig Tansig	98.5	100
NN4	13	LM/BR/MAPSTD	7.23	Tansig Purelin	100	98
NN5	15	LM/BR/ +PCA	5.91	Tansig Purelin	100	100
NN6	12	LM/BR/+PCA	157	Tansig Tansig	97.5	100
NN7	14	LM/ ES/+PCA	0.0354	Tansig tansig	98.5	98
NN8	14	LM/ ES/+PCA	0.0109	Tansig Purelin	97.5	100

Table 5 Actual outputs generated by the best ANN configuration sr. no 5 of Table 4 for test input

	Actual output & condition identified										
	N	N	O/L	O/V	U/V	S	SOP	SOP	VU/B	VU/B	VU/B
VU/B	-0.002	-0.068	0.003	-0.024	0.035	-0.002	-0.011	-0.04	1.05	0.781	0.965
SOP	0.021	-0.027	0.005	0.005	0.007	0.002	1.008	1	-0.025	-0.089	0.03
S	-0.005	-0.014	-0.001	-0.027	0.003	0.996	0.003	-0.015	0.005	-0.043	-0.018
U/V	0.008	0.111	-0.025	0.010	0.9302	-0.028	0.054	0.017	-0.01	-0.053	-0.012
O/V	-0.003	-0.000	0.0	1.005	0.000	-0.000	0.0	-0.000	0.001	-0.016	-0.002
O/L	0.006	-0.016	1.007	-0.003	0.0142	0.0005	-0.001	-0.01	-0.02	0.0165	-0.011
N	0.965	0.983	0.010	0.010	0.019	-0.002	0.005	0.014	0.013	0.318	0.035

Table 6 Statically parameter sensitivity (Sen.) and specificity (Spe.) for input dataset

NN no.	NN1		NN2		NN3		NN4		NN5		NN6		NN7		NN8	
Data set	Sen. (%)	Spe. (%)	Sen. (%)	Spe. (%)	Sen. (%)	Spe. (%)	Sen. (%)	Sen. (%)	Spe. (%)	Spe. (%)	Sen. (%)	Spe. (%)	Sen. (%)	Spe. (%)	Sen. (%)	Spe. (%)
N	96.7	99.39	93.3	98.78	93.3	98.78	100	100	100	100	93.3	98.62	96.1	99.39	96.7	98.17
O/L	95	100	100	100	95	100	100	100	100	100	95	100	90	100	90	100
O/V	100	99.39	100	99.39	100	99.39	100	100	100	100	100	100	100	100	100	99.43
U/V	100	99.42	95	100	100	99.42	100	100	100	100	100	99.42	100	100	95	99.43
S	100	100	100	100	100	100	100	100	100	100	100	99.39	100	100	100	99.39
SOP	92	100	100	100	100	100	100	100	100	100	92	100	100	100	100	100
VU/B	91.83	98.63	98	99.31	98	100	100	100	100	100	95.9	98.17	92	98.62	93.9	98.17

Table 7 Stastical parameter sensitivity (Sen.) and specificity (Spe.) for test (unseen) dataset

NN no.	NN1		NN2		NN3		NN4		NN5		NN6		NN7		NN8	
Data set	Sen. (%)	Spe. (%)	Sen. (%)	Spe. (%)	Sen. (%)	Spe. (%)	Sen. (%)	Sen. (%)	Spe. (%)	Spe. (%)	Sen. (%)	Spe. (%)	Sen. (%)	Spe. (%)	Sen. (%)	Spe. (%)
N	100	100	100	97.77	100	100	100	100	100	100	100	100	100	100	100	100
O/L	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
O/V	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
U/V	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
S	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
SOP	100	100	100	100	100	100	100	96.43	100	100	100	100	93.3	100	100	100
VU/B	100	100	87.5	100	100	100	87.5	100	100	100	100	100	100	97.72	100	100

7. Results and Discussions

The NN training objective is to adjust the weights so that NN can produce the desired output when it appears to unseen data and identify and classify fault condition. We have tested our problem not only to produce good detection results but to find the best NN configuration. We have trained different single hidden layer ANN configuration as shown in Table 4 having tansig or purelin as output layer activation function, ES or BR as generalization method and different data processing technique. All input and target vector data are preprocessed by mapping the row to zero mean and unity variance using Matlab mapstd function and data of NN no. 5 to 8 of table 4 are also further preprocessed with PCA. We have tested NN configurations atleast 25 or more iteration with initializing new weights every time. As if network has lesser hidden neuron it underfit the data and if more nodes then problem of overfitting increase so different ANN configuration tested with different hidden layer neurons gradually increased from lower side. All ES run are stop with validation stop and all BR run when maximum weight damping factor reached. As we increase hidden node after certain node even the performance error reduce the results of test simulation and input simulation deproved i.e. the network may overtrained. At lesser hidden node the network is not sufficiently simulate the

Table 8 Weights from input to hidden layer and bias {1} of hidden layer for NN5 configuration

Weight from inputs to hidden layer					bias {1}
-12.8441	20.50278	-15.5231	-3.11923	4.49661	21.47516
-18.7561	7.291488	-0.70412	0.353977	-0.47013	24.87071
0.687205	0.890272	5.426286	-8.8521	-2.76763	-2.32015
-0.59351	-0.61689	3.240572	13.79844	2.490056	-0.66864
-6.69662	4.196722	41.84173	14.72279	0.29078	0.478036
22.86743	-6.45144	-11.0188	3.009632	-3.71165	-20.87
-0.39003	-0.60865	3.487829	1.250031	3.010849	3.881527
0.155606	0.177922	-0.26915	-0.4196	-0.91266	-1.09495
-25.5709	-0.7389	4.654554	-1.52011	-17.6461	-17.9528
-5.65354	-1.76028	-7.76329	-5.84226	-3.01687	3.352388
-4.23849	-0.55612	0.234111	-2.33927	6.767685	-4.96711
0.119874	6.354606	-16.5281	-19.8093	-23.6356	-10.8796
-3.31287	22.76426	8.615496	-0.72317	1.425839	-39.4266
-1.34297	-0.86946	4.300653	18.72181	4.853103	-1.39389
4.857575	-7.19375	-4.68344	0.950667	11.97972	-1.46213

Table 9 weights from hidden to output layer and bias {2} for output layer for NN5 configuration

Weight from hidden to output layer							
-0.20059	-0.103952	1.072027	-5.120192	-0.52947	0.0946069	0.8759837	-2.23112
-0.0513	0.0079612	-0.367057	-0.752558	-0.032572	0.001565	-1.880298	-3.112651
-0.051002	-0.09903	0.0694288	0.3408475	-0.023409	-0.025516	-0.021006	3.316647
-0.445679	9.6251029	0.1097543	1.6091935	-0.194420	9.5270474	9.5681648	2.0385375
-0.000933	-0.004281	0.0110862	-0.028604	-0.013143	-0.001533	-0.0282	-0.071584
-0.039399	0.0014213	-0.031695	-0.06616	0.0981309	-0.012291	0.0755433	0.3027767
1.2987222	0.0050149	-1.208867	3.2029969	0.9966521	-0.049063	0.9049057	2.8233464

Weigh from hidden to output layer							bias {2}
0.0089163	0.9897995	-1.328049	0.4326265	-0.463633	5.7692258	-0.440815	-2.616256
0.0160795	-1.345946	1.8061079	-0.187141	0.1574085	0.6125633	-0.087043	1.685967
-0.039060	-0.155359	-0.045127	-0.184768	0.0236251	-0.210255	-0.016774	2.5881928
0.0210392	0.1400972	-0.180075	0.0003698	0.0116028	-1.407999	-0.163520	-8.016360
-0.004025	-0.007028	-0.003428	-0.024272	1.4158566	0.0258252	-0.012986	0.95104
-1.645008	0.1187092	-0.131443	0.0537402	-0.071164	0.0287771	-1.551911	-0.442851
1.2881247	-0.435637	0.4795224	-0.337987	-1.029372	-4.200836	2.2582023	-0.796636

test and input data. It is observed good performance of NN configurations are obtained when hidden neurons are nearby double than input neurons while output neurons are same as input neurons. In case of BR the result are observed good and remain nearly same after certain good node configuration selected while in case with ES it deproved after further increase in hidden node.

Our aim is also to get well train and well generalized nn configuration. We have used total classification accuracy, sensitivity, specificity to determine the best performance of ANN as fault detector. Sensitivity is the ratio of number of true positive decisions to number of actually positive cases. Specificity is the number of true negative decisions to number of actually negative cases. Total classification accuracy is number of correct decisions/ total number of cases.The best NN configuration obtained is NN5 (table 4) with 100% total classification accuracy of input and test data and well classified all condition including normal with 100% sensitivity and specificity shown in table 6 and table 7. It also gives best overall training accuracy result. The network was stopped when performance, sum squared parameters are observed constant and maximum weight damping factor reached. Actual output obtained is shown in Table 5 for the Table 2 test (unseen) input for NN5 configuration of table 4.

Statistical values sensitivity and specificity of all NN configurations with classified set as normal and different faulty condition are shown in Table 6 and Table 7. It found that well trained and well generalized configuration is NN5. Weights from input to hidden layer , from hidden to output layer and biases of hidden and output layer are shown in table 8 and table 9 for the best NN configuration No 5 of table 4.

Conclusions

Induction motors are important and major industrial load in industries and demands more accurate, sensitive and specific fault detection scheme. Various external fault conditions are simulated for 4KW induction motor using MATLAB/SIMULINK & FFNN trained with LM backpropagation algorithm. Normal condition and External faults are simulated with varied operating voltage. Nos. of ANN configurations are tested starting form lower to higher nos. of hidden neurons in hidden layer with generalization technique Es or BR, output layer processing function tansig or purelin and data preprocessing technique like minimum and maximum value of each row mapped to have zero mean and unity SD without and with PCA. It is also observed that good results are obtained when hidden neuron are almost double than input neurons while output neurons are same as input. The best configuration evaluated among all good one is NN5 using stastical measures total classification accuracy of test and train input data, sensitivity, specificity and overall training accuracy. NN5 is obtained using LM algorithm with bayseian regularization, purelin as output layer activation function and input data are processed with unity standard deviation, zero mean and principle component analysis. Ann can detect all unseen external fault test data with 100 % classification accuracy, sensitivity and specificity. NN detect all kind of voltage unbalance as voltage unbalance, opening of any phase as SOP, locked rotor at start and distinguish unbalance less than 1% as normal condition and above as voltage unbalance case. It detects wide range of O/V, U/V and O/L successfully.

Conventional protective scheme have to wait upto safe stalling time while detecting stalling condition at start which may undue stress the induction motor mechanical parts and windings and also it is difficult to detect correctly different kind of unbalance voltage as unbalance voltage condition using the same. This study describes the writer's efforts to identify such condition effectively with best NN configuration and step forward to proven ANN as future enabling protective technology.

References

Aroui, T., Koubaa, Y., and Toumi, A., (2008), Magnetic Coupled Circuit Modeling of Induction machines oriented to Diagnostics. *Leonardo Journal of sciences*, 7 (13), 103-121.

Anderson P.M. (1999), Power System Protection. *IEEE Press Power Engineering*, NJ, Willey interscience, 782- 800.

Chow, M., Sharpe, R. and Hung. (1993) J., On the Application and Design of Artificial Neural Networks for motor fault detection – Part I. *IEEE Transactions on Industrial Electronics*, 40, (2), 181-188.

Chow, M., Sharpe, R. and Hung, J. (1993), On the Application and Design of Artificial Neural Networks for motor fault detection – Part II. *IEEE Transactions on Industrial Electronics*,40(2),189-196.

Chudasama K.J., Shah V. (2012), Induction motor noninvaisve fault diagnostic techniques: A Review. *International journal of Engineering Research and Technology*, 1 (5)

Cummings, P., Dunki-Jacobs, J. and Kerr. (1985) R. Protection of Induction Motors against Unbalanced Voltage Operation', *IEEE Transactions on Industry Applications*, IA-21 (4), 778-792.

Goode, P.V., Chow M.Y. (1995), Using a Neural/Fuzzy system to extract Heuristic knowledge of incipient faults in I.M: part I-Methodology, *IEEE Transactions on industrial Electronics*, 42 (2), 131-138.

Hessami, M., Francois, A., Viau, A. (2004) Selection of Artificial Neural Network Model for the post – calibration of wheather Radar Rainfall estimation, Journal of Data Science, 107-124

Kola, S., Varatharasa L. (2000) Identifying 3 phase I.M faults using neural networks. *ISA Transactions*, 2000 (39), Elsevier, Science Direct, 433-439.

Mackay, D.J.C. (1992) Bayseian interpolation, *Neural computation*, 4(3), 415-447.

Matlab 7.10 Mathworks Inc. (2010).

Oza, B.A, Nair, N., Metha R. and Makwana, V., (2010), Power system protection and switchgear.TataMc-Graw Hill, 306-311.

Peter Zhang, G. (2000) Neural Networks for classification: A Survey. *IEEE Transactions on Systems, Man and Cybernetics – Part c: Applications and Reviews, 30 (4), 451-461.*

Riley, C., Lin B., Habetler, T. &Kilman G. (1999) Stator current harmonics and their causal vibrations: A Preliminary

investigation of sensorles vibration monitoring applications, *IEEE Transactions on Industrial applications, 35(1), 94-99*.

Ranganathan, A. (2004) The Levenberg-Marquardt Algorithm . *Technical Report*. http://www. ananth.in / docs/ lmtut.pdf, Honda Research Institute.

Rowies S. (1999) Levenberg _Marqardt Optimization', http:// WWW. Cs. Nuu. edu/~rowies/note/lm. Pdf.

Rumelhart, D. E., Hinton, G. E. & Williams, R. J. (1986). Learning internal representations by error propagation, in D. E. Rumelhart& J. L. McClelland (eds), Parallel Distributed Processing: Explorations in the microstructure of cognition, MIT Press, chapter 8, pp. 318-361.

Sin M.L., Soong W.L. &Ertugrul. (2003) On –line condition moitoring and fault diagnosis – A survey. *Australian Universities Power Engineering Conference*, Brisbane, Australia.

Sudha, M., Anbalagan, P. (2009) A protection Scheme for three phase I.M form Incipient faults using Embedded Controller. *Asian Journal of Scientific Research*, 2(1).

Siddeque, A., Yadav. G. & Singh, B. (2004) Identification of 3 phase induction motor faults using ANN. *IEEE symposium*, USA, 30-34.

SaeidMoosavi, S., Djerdir, A., Ait-Amirat, Y., &Khaburi D.A., (2012) fault detection in 3 phase traction motor using Artificial Neural Networks. Transportation Electrification conference and Exposure (ITEC) IEEE, 1-6.

Tag Eldin E.M., E.M.Emara H.R., Aboul-Zahab E.M. &Refaat S.S., (2007) Monitoring and Diagnosis of External faults in Three Phase Induction Motor using Artificial Neural Network. *IEEE Power Engineering Society General Meeting*.

Tallam, R., Habetler, T., and Harley, R. (2003) Experimental testing of a neural-network-based turn-fault detection scheme for induction machines under accelerated insulation failure conditions, *4th IEEE International Symposium on Diagnosis of Electrical Machines Power Elect. and Drives*, *SDEMPED*, 58-62.

Xinghuo, Y., OnderEfe, M, and Kaynak, O., (2002) A General Backpropagation Algorithm for feedforward Neural Networks Learning', *IEEE Transactions on Neural Networks.*, 13 (1).