CHAPTER 4

FRT OF GRID CONNECTED DFIG EMPLOYING PROPOSED FUZZY SPIKING NEURAL NETWORK CONTROLLER

4.1 INTRODUCTION

In this work, a controller is designed employing fuzzy reasoning into spiking neural network for performing control action of grid connected doubly fed induction generator. As presented in previous chapter, SNN that belong to the third generation neural networks are applicable for problems that can be solved employing the first two generations. In addition to this, it is noted over the few years that these SNN possess the capability to act as a controller with relatively minimal number of spikes and also SNN is a temporal based computation. Based on these advantages of SNN and the growing interest in the temporal computation promote the applicability of SNNs in numerous applications like controller design, classification and image processing. The previous chapter presented a controller design with SNN IZ model wherein their weights are tuned using proposed hybrid particle swarm optimization and group search optimizer.

In this research work, fuzzy processing is carried out for the input that is to be presented for SNN and each input spikes represent a fuzzy input membership function. Fuzzy reasoning based on its membership function definition and rule base results in reduction of the computations in SNN which
already utilizes multiple synapses for each connection between a pre-synaptic and postsynaptic neuron. The developed FSNN model in this thesis is applied to perform controller action for grid connected DFIGs to control the reactive power and as well to eliminate the presence of auxiliary hardware components. Simulation results computed prove the efficiency of the proposed controller in terms of the fitness function evaluated and other performance metrics than that of the earlier proposed controller modules existing in the literature.

4.2 NEED FOR PROPOSED FUZZY SPIKING NEURAL NETWORK MODEL WITH LITERATURE BACKGROUND

For the past few decades, considering the fact that individually intelligent control techniques are well known for their advantages and henceforth are observed to be suitable for specific applications. Related to this the hybrid systems combining fuzzy logic and neural network models are widely employed for control and classification applications.

SNN approach is noted for its effectiveness in mapping between input and output but at times finds difficult in determining the causal relationship between the inputs and outputs. The input parameters that play a major role on the convergence of the particular output are not differentiated from the others and at this juncture, SNNs fail to provide the internal dynamics of the system. On the other hand, fuzzy systems are well known in stating how they reach their decisions but an expert knowledge is required in this case to develop and formulate the rules and membership functions which are used to make the decisions. When there exists lack of information in the application, then the fuzzy systems requires adjustment of membership functions to obtain the proper output. Hence this research focuses on
developing a hybrid approach to act most effectively with the controller action carried out by the spiking neural network and as well to minimize the computational cost and time yielding better performance. The fundamental background for this research work based on the available literatures is presented in this section.

Wang, T et al. 2015, have proposed a fault diagnosis method based on Fuzzy Reasoning Spiking Neural P Systems (FRSNP system) for power transmission networks. To test the validity and feasibility of FRSNP, seven cases of a local subsystem in an electrical power system are used.

Wang, T et al. 2014, discusses the application of fuzzy reasoning spiking neural P systems with trapezoidal fuzzy numbers (tFRSN P systems) to fault diagnosis of power systems. Some case studies show the effectiveness of the presented method. Brief comparisons between the presented method and several main fault diagnosis approaches from the perspectives of knowledge representation and inference process were presented. Oniz & Kaynak 2015, have developed a sliding mode theory based supervised training algorithm that implements fuzzy reasoning on a SNN. The results of the real-time experiments indicate that stable online tuning and fast learning speed are the prominent characteristics of the proposed algorithm.

Obo & Kubota 2014, proposed a structured learning in FSNN to enable optimization of the membership functions in the learning process. The effectiveness of the proposed method is demonstrated through comparative experiments. Ramírez-Mendoza 2014, presented the study of the response of the connection of adaptive fuzzy spiking neurons with self-synapse in each single neuron. The simulations results for a single neuron with self-synapse and three neurons connected in series with self-synapse each one, are presented and performed in Simulink of Matlab. Xiong et al. 2013,
implemented Fuzzy Reasoning Spiking Neural P systems (FRSN P systems) for fault diagnosis of power systems. Three different power systems are used to demonstrate the feasibility and effectiveness of the proposed fault diagnosis approach. In addition, it is independent of the scale of power system and can be used as a reliable tool for fault diagnosis of power systems.

Wang & Peng 2013, have proposed a new class of spiking neural P systems called Weighted Fuzzy Spiking Neural P Systems (WFSN P systems). The proposed WFSN P systems is compared with other knowledge representation methods, such as fuzzy production rule, conceptual graph, and Petri nets, to demonstrate the features and advantages of the proposed techniques. Peng et al. 2013, have extended Spiking Neural P systems by introducing some new ingredients (such as three types of neurons, fuzzy logic and new firing mechanism) and proposed the Fuzzy Reasoning Spiking Neural P systems (FRSN P systems). Wang & Peng 2013, has proposed a class of Fuzzy Spiking Neural P systems by introducing some new features, called Adaptive Fuzzy Spiking Neural P systems (AFSN P systems).

Xie et al. 2013, have proposed a method of self-learning fuzzy spiking neural network to solve contradiction between application problems example scale and network scale. Xie, Z-J & Xie 2012, have proposed Fuzzy spiking neural network for Fault Diagnosis of Gas blower sets. Glackin et al. 2011, presented a supervised training algorithm that implements fuzzy reasoning on a spiking neural network. Tang, D & Kubot 2010, has analyzed the performance of the human localization by a SNN in informational structured space based on sensor networks.

Obo et al. 2010, developed SNN for the state estimation of human behaviours on the bed with fuzzy membership functions to extract sensitive changes of sensory inputs. Ionescu & Dragoș 2008, investigated
some applications of spiking neural P systems regarding their capability to solve some classical computer science problems. In this respect versatility of such systems is studied to simulate a well-known parallel computational model, namely the Boolean circuits. Glackin et al. 2008, presented a supervised training algorithm that implements fuzzy reasoning on a spiking neural network.

Sasaki & Kubota 2006, proposed the behavioural learning based on a SNN to realize high adaptability of a mobile robot. The effectiveness of the proposed method is verified through experimental results on behavioural learning for collision avoidance and target tracing behaviour. Kubota & Sasaki 2005, modelled a FSNN for behaviour learning of a mobile robot. The simulation results show the robot can update the network structure and learn the weights of FSNN according to the Spatio-Temporal context of the facing environment.

Considering the importance and applicability of the FSNN for various applications, this work proposes the new fuzzy reasoning based SNN to handle the LVRT rate of the grid connected wind turbines effectively. The modelling of doubly fed induction generator wind turbine system holds the same as that presented in section 2.3 of chapter 2. The conventional DFIG control system considered for comparison with that of the proposed controller is also the one described in section 2.3.2 of chapter 2.

4.3 PROPOSED FUZZY SPIKING NEURAL NETWORK CONTROLLER MODEL

Numerous researchers have worked in the area of fuzzy logic approaches and have applied the fuzzy logic systems for different applications which can be studied from the available literatures (Abe & Lan 1995, Alcala et
This section proposes a FSNN classifier with suitable parameter tuning mechanism for developing effective control action over the FRT of grid connected DFIG.

### 4.3.1 Design of Proposed FSNN controller model

With the growing scenario of incorporating intelligent computing techniques for various applications with their advantages and disadvantages brought out, they are made available for specific applications, and as well the hybrid systems that combine the features of fuzzy logic and neural networks are employed for controller design problems. In this research thesis, a fuzzy spiking neuronal model is modelled and employed for carrying out over voltage fault ride through of wind turbines.

The proposed hybrid model is developed so that it combines fuzzy systems with SNN. Here a NN with spiking neurons is employed for adapting the parameters of the triangular membership functions. These SNN spikes possess firing times and the firing times of the input spiking neurons facilitate the generation of the desired output of a network. For the grid connected DFIG problem considered, the input attributes specifically are converted into the inputs of the SNN using fuzzy membership functions. This is similar to the concept of population encoding, wherein an input variable gets distributed over a multiple input neurons employing multiple receptive fields. During this process, the neurons possessing the highest values fire at the earliest and the neurons possessing smaller values take longer time to fire.

The basic problem concerned with that of the SNN are these algorithms require higher computational time because of the increasingly higher number of parameters for adaptation. These parametric variations are
due to the architecture model wherein multiple synapses are employed for each and every interconnection between a pre synaptic and post synaptic neuron. In the proposed FSNN controller, the output layer employs linear activation function for reducing the computational complexity of the NN.

Figure 4.1 shows the three layered proposed FSNN controller model designed to handle the specified application. The fuzzy input triangular membership values that corresponds to a respective attribute is given as the inputs to the input neurons of SNN model. For the considered ‘n’ number of input attributes, ‘m’ numbers of membership functions are assigned for each of the specified ‘n’ inputs. The output layer of the proposed controller is made up of the respective number of output class entities as given for the specified data. In the proposed neuronal architecture model, there exists input layer, spike hidden layer and the output layer. The inputs given to the SNN are the fuzzy membership values. The membership values are defined with respect to the datasets considered and the number of attributes under consideration. In this proposed FSNN controller module, triangular membership functions are employed to compute the firing times of the neurons in the input layer.
4.3.2 Proposed FSNN controller learning algorithm

In this research work spiking neurons act as the temporal computational units for SNN model resulting in precise firing of neurons as applicable for the control problems. The fuzzy tuned spike neurons are employed to retrieve information and as well act for binary data bit coding. The proposed algorithm developed to control the reactive power for the considered grid connected doubly fed induction generator in this research is as follows:

Step 1: Initiate the learning process of the proposed algorithm.
Step 2: Present the data samples to the network with respect to the given attributes.
Step 3: For each of the input attributes, assign the linguistic variables.
Step 4: For each of the input parameters, assign the membership functions.

For any triangle with sides A, B and C,

For an approximate isosceles triangle, for the given conditions $A \geq B \geq C \geq 0$ & $A + B + C = 180^\circ$ the membership function is given as,

$$\mu_I(A, B, C) = 1 - \frac{1}{60^\circ} \min(A \cdot B \cdot B \cdot C)$$  \hspace{1cm} (4.1)

For an appropriate right triangle, for the same conditions, the membership function is given as,

$$\mu_R(A, B, C) = 1 - \frac{1}{90^\circ} \left(A \cdot 90^\circ\right)$$  \hspace{1cm} (4.2)

For the other occurring triangles the membership function can be given as the complement of the logical union of the two already defined membership functions.

$$\mu_O(A, B, C) = \overline{I \cup R}$$  \hspace{1cm} (or) \hspace{1cm} (4.3)

Employing Demorgans law, membership function is computed as,

$$\mu_O(A, B, C) = I \cap \overline{R} = \min \left\{ I \cdot \mu_I(A, B, C), \frac{I \cdot \mu_I(A, B, C)}{I \cdot \mu_R(A, B, C)} \right\}$$  \hspace{1cm} (4.4)

Step 5: After then, the membership functions are given to the input neurons of the input layer.

Step 6: Initialize the weights randomly between 0 and 1. Set the value of threshold.

Step 7: Each of the input synapse receives input signal during the feed – forward phase and the input signal gets transmitted to the considered next hidden layer neurons. The hidden layer units get processed and the outputs from these units are passed on to output units.

Step 8: The output layer of the neuronal units delivers spike functions as response for the given input signals. The firing time of a spike neuron will then be computed and is given as ($t_s$). The desired time of the spike flow is given by ($t_d$).
Step 9: Error back-propagation algorithm is implemented between output and hidden and hidden and input layers. Compare time ‘t\(_s\)’ with ‘t\(_d\)’. The mean square error equation is given by,

\[ E = \frac{1}{2} \sum_{j \in y} (t_s - t_d)^2 \]  \hspace{1cm} (4.5)

Step 10: Error \( \delta_j \) between hidden and output is computed and is given by,

\[ \delta_j = \frac{\delta E}{\delta x_j(t_s)} = \frac{(t_d - t_s)}{\sum_{i \in r} \sum_l w_{il} \frac{\delta y_{jl}(t_s)}{\partial t_s}} \]  \hspace{1cm} (4.6)

\[ \Delta w_{ijk} = -\alpha \frac{\delta E}{\partial w_{ijk}} = -\alpha y_{ik}(t_a) \delta_j \]  \hspace{1cm} (4.7)

Where, the pulse edge is given by ‘E’, Dirac function is \( \delta(x) \) and \( \alpha \) represents the learning rate with ‘y’ being the output.

Step 11: Evaluate ‘\( \delta_i \)’ with actual (t\(_{af}\)) and desired firing (t\(_d\)) time of each \( \delta_j \) calculated and that of each of the hidden neurons.

\[ \delta_i = -\frac{\delta t_{af}}{\delta x_i(t_{af})} \sum_{j \in r} \delta_j \frac{\delta x_j(t_{af})}{\partial t_{af}} \]  \hspace{1cm} (4.8)

\[ \Delta w_{hik} = -\eta y_{hk}(t_{af}) \delta_j \]  \hspace{1cm} (4.9)

Step 12: Perform updation of weights and the new weight will be the summation of old weights and change in weights.

\[ w_{ijk}(new) = w_{ijk}(old) + \Delta w_{ijk} \]  \hspace{1cm} (4.10)

Step 13: For each of the neurons in the hidden layer, the change in weights are computed and the weight updation is performed as follows:

\[ w_{hik}(new) = w_{hik}(old) + \Delta w_{hk} \]  \hspace{1cm} (4.11)

Step 14: Perform the specified number of iteration until the stopping
conditions are reached. The stopping condition can be specified number of iterations or until mean square error (MSE) reaches a minimum value or a set goal is met.

Step 15: Stop the learning process of the algorithm.

The algorithm is trained in this case of controller model until the mean square error reaches a minimum value and the computational time incurred during the process flow of the control action is also observed. In this thesis, the proposed FSNN controller is applied for handling the fault ride through of the grid connected DFIG.

4.4 FAULT RIDE-THROUGH OF DFIG MODULE USING PROPOSED FUZZY SNN CONTROLLER

DFIG module with FRT employing proposed FSNN controller is as shown in Figure 4.2. GSC control system remains the same as that of the conventional DFIG module for the proposed control design and the necessary modifications are done at the rotor side converter control system module.
The operation of the FSNN controller starts only at the point when the ac voltage $V_s$ exceed more than 10% of the set reference voltage. The constraints that should be considered for protecting the DFIG are the DC-link overvoltage and the rotor over current. The DC-link over voltage and the rotor over current are set such that it does not exceed their set limits in the considered restoring period. Also, steps should be taken appropriately to transfer the additional energy generated in the rotor via the converters to the grid. The transfer of additional energy induced in the system enables the DC-link voltage and rotor current to maintain at their respective normal values. When the rotor current is reduced by fast transfer of the stored energy from rotor to grid, there is a chance that the DC-link voltage increases abruptly and it gets deviated from the normal limits.

**Figure 4.2  Proposed Fuzzy Spiking Neural Network Controller for the system**

The operation of the FSNN controller starts only at the point when the ac voltage $V_s$ exceed more than 10% of the set reference voltage. The constraints that should be considered for protecting the DFIG are the DC-link overvoltage and the rotor over current. The DC-link over voltage and the rotor over current are set such that it does not exceed their set limits in the considered restoring period. Also, steps should be taken appropriately to transfer the additional energy generated in the rotor via the converters to the grid. The transfer of additional energy induced in the system enables the DC-link voltage and rotor current to maintain at their respective normal values. When the rotor current is reduced by fast transfer of the stored energy from rotor to grid, there is a chance that the DC-link voltage increases abruptly and it gets deviated from the normal limits.
To design an effective and efficient fault ride through, the transition signals of the rotor current should consider the nominal values of the DC-link voltage. The proposed FSNN controller has the inputs as $v_{dc}^*$ and $i_r^*$ and the output of the neural network controller is the $N_{ctf}$. $v_{dc}^*$ and $i_r^*$ acts the inputs to the FSNN and are as represented in equation (2.28) and (2.29) of chapter 2. The parameter specification for the proposed FSNN controller is as given in Table 4.1.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Proposed FSNN controller model</th>
</tr>
</thead>
<tbody>
<tr>
<td>Weights and Bias</td>
<td>Randomly initialize between 0 to 1</td>
</tr>
<tr>
<td>Threshold</td>
<td>1.0</td>
</tr>
<tr>
<td>Learning rate</td>
<td>0.3</td>
</tr>
<tr>
<td>Maximum Iterations</td>
<td>100</td>
</tr>
<tr>
<td>Membership function</td>
<td>Triangular (0 to 1)</td>
</tr>
<tr>
<td>No. of hidden neurons</td>
<td>½ that of the input neurons</td>
</tr>
</tbody>
</table>

The fuzzy controller is constructed by considering the DC-link voltage and rotor current of the system as conditional variables and the membership function of these values act as input to the spiking neuronal model i.e., the fuzzy module get two inputs.

$$Q_{member} = \Phi(v_{dc}^*, i_r^*)$$  \hspace{1cm} (4.12)

Where $v_{dc}^*$ and $i_r^*$ denote DC-link voltage and rotor current error respectively and Q results the membership function values that acts as input to the SNN model. The membership functions are defined for the inputs ($v_{dc}^*, i_r^*$). For the considered input variables, seven linguistics - Negative Large (NL), Negative Medium (NM), Negative Small (NS), Zero (ZE), Positive Small (PS), Positive
Medium (PM) and Positive Large (PL) are assigned. The membership plot of \( v_{dc,i_r}^* \) are as shown in Figure 4.3.

**Figure 4.3  Membership functions of \( (v_{dc,i_r}^*,i_r^*) \)**

Table 4.2 shows the generated set of membership functions entering the spiking neuronal model based on which the neuron fires. Then the SNN acts to compute the firing times and proceeds with the learning process and attempts to determine \( N_{crf} \) which acts as a quantity to correct the output \( V_{qr} \) of the current controller.
### Table 4.2 Sample Membership function values of Fuzzy Model to SNN controller

<table>
<thead>
<tr>
<th>Input parameters</th>
<th>Membership values</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1.0</td>
</tr>
<tr>
<td></td>
<td>0.1</td>
</tr>
<tr>
<td></td>
<td>0.4</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
</tr>
<tr>
<td></td>
<td>0.2</td>
</tr>
<tr>
<td></td>
<td>0.1</td>
</tr>
<tr>
<td></td>
<td>0</td>
</tr>
</tbody>
</table>

|                  | 0.8   | 0.6   | 0.6   | 0.6   | 0.4   | 0.4   | 0.1 |
|                  | 0.2   | 0.4   | 0.4   | 0.5   | 0.8   | 0.6   | 0.4 |
|                  | 0.4   | 0.4   | 0.6   | 1.0   | 0.6   | 0.4   | 0.1 |
|                  | 0.3   | 0.3   | 0.3   | 0.3   | 0.7   | 0.2   | 0.2 |
|                  | 0.2   | 0.2   | 0.4   | 0.4   | 0.6   | 0.4   | 0.2 |
|                  | 0.1   | 0.2   | 0.2   | 0.6   | 0.6   | 0.3   | 0.1 |
|                  | 0     | 0.1   | 0.3   | 0.5   | 0.5   | 0.3   | 0 |

Figure 4.4 shows the three dimensional graph for the FSNN output $N_{crf}$ with respect to $V_{dc}^*$ and $i_r^*$. The proposed FSNN controller can be employed for various sizes of machines and as well for multi machine models. The training of the NN controller and the fuzzy reasoning process remains the same in all cases.
4.5 SIMULATION RESULTS OF PROPOSED FUZZY SPIKING NEURAL NETWORK CONTROLLER

The proposed fuzzy spiking neural network controller is validated by applying the said control strategy for a 1.5 MW wind turbines connected to a 25 kV distribution system exporting power to a 120 kV grid through a 30 km 25 kV feeder. The electrical system, Theodoros et al. 2011, considered for validating the proposed control design is the same as given in Figure 2.9 of chapter 2. The specifications for the parameters of the DFIG system and the grid system, Theodoros et al. 2014, considered for simulation remains same as that considered in Table 2.4 of chapter 2. The complete simulation process is carried out for wind speed of 12 m/s. The entire proposed control strategy was run in MATLABR2009 environment and executed in Intel Core2 Duo.
Processor with 2.27GHz speed and 2.00 GB RAM. Simulink environment in MATLAB is used to model the converter modules.

The work is done to handle three phase symmetrical grid faults and the occurrences of these faults are noted for 0.5 seconds. The simulation response is noted for both the traditional DFIG control system and as well that of the proposed control system design. The simulation plots of the traditional control systems are the same as that simulated in chapter 2 and shown in Figure 2.10 through Figure 2.17. From the traditional controller response it can be noted that this system requires an auxiliary system for fault ride. During the process of simulation it is observed that the DFIG along with the proposed controller handles the complete response at fault periods and after fault periods and gets ride-through the fault eliminating the applicability of auxiliary hardware.

![Graph](image)

**Figure 4.5** Response of three phase stator voltage for proposed Fuzzy Spiking Neural Network Controller
Figure 4.6  Response of rotor current for proposed Fuzzy Spiking Neural Network Controller

Figure 4.7  Response of stator current for proposed Fuzzy Spiking Neural Network Controller
Figure 4.8  Response of wind turbine output active power for proposed Fuzzy Spiking Neural Network Controller

Figure 4.9  Response of wind turbine output reactive power for proposed Fuzzy Spiking Neural Network Controller
Figure 4.10  Response of rotor speed for proposed Fuzzy Spiking Neural Network Controller

Figure 4.11  Response of $q$-component of rotor voltage for proposed Fuzzy Spiking Neural Network Controller
From the traditional controller simulation plots, it is observed that the entire response of the system varies in a non-linear manner resulting in more fluctuations in the grid side. The conventional control design is modified in a manner to enable appropriate handling of ride-through the fault conditions. The proposed FSNN controller is simulated for 100 generations and the entire response observed during the simulation response are as shown through Figure 4.5 through Figure 4.12. The computational process of the neuronal model is studied at 85% voltage dip.

The simulation responses noted for the control parameters employing proposed controller proves the removal of fluctuations that occurred during the conventional control design and as well is noted to reach the steady state at the earliest. Further, over voltages noted at DC-link point and over currents at the rotor side are noted to be well within the limit of the maximum set threshold values. Henceforth, the capacitor damage is prevented.
and fault and post fault occurrence to the grid are controlled in an effective manner. Figure 4.6 and Figure 4.7 shows the rotor current and stator current with fluctuations reduced and achieving the steady state value quickly.

The output responses of the wind turbine for output active and reactive power are shown in Figure 4.8 and Figure 4.9. In the proposed controller design, RSC will not be cut during the fault condition and this RSC provides the required reactive power to the grid and this handles the sufficient voltage drop, if it occurs in the system. The developed FSNN controller prevents more amount of reactive power getting transferred from DFIG after the fault. In the proposed controller module, DFIG supplies the grid with the required amount of reactive power and this gets sustained with that of the grid voltage.

During the occurrence of fault, the rotor speed of the proposed FSNN controller increases which is seen in Figure 4.10. The increase in speed of the rotor is observed because of the wind turbine to store huge energy. When fault occurs, there is a huge drop in the voltage and the wind power gets transferred as kinetic energy to the rotor without dissipating it to that of the grid. When the fault duration ends, the grid receives this energy and the increase in the rotor speed get automatically settled to the value before the fault has occurred.

Figure 4.11 and Figure 4.12 shows the q-component and the modified q-component of the rotor voltage of the proposed fuzzy spiking neural network controller design. The occurrence of transients is noted and the proposed FSNN controller acts in a manner to bring back the ac voltage to its set value resulting in the above said transient. At the time of simulation process, if there occurs a higher voltage dip then the DFIG is allowed to disconnect. This is the worst condition on getting connected to the grid. The
A complete simulation study is carried out for wind speed at 12 m/s and at a voltage dip of 85%. The proposed FSNN controller is found to ride-through the fault occurred. Table 4.3 shows the fitness values and mean square error values evolved during various iterations. The DC-link voltage and the rotor current are found to be maintained within the said permissible limits based on the control action. At the end of 100 iterations, it is noted that the mean square error and the fitness value has reached 0.0120 and 3.1724 respectively.

Table 4.3  Mean square error and Fitness function value evolved during iterations

<table>
<thead>
<tr>
<th>Iterations</th>
<th>Fitness function (f)</th>
<th>Mean square error</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>8.0065</td>
<td>0.3564</td>
</tr>
<tr>
<td>20</td>
<td>6.7856</td>
<td>0.3198</td>
</tr>
<tr>
<td>30</td>
<td>6.5780</td>
<td>0.2605</td>
</tr>
<tr>
<td>40</td>
<td>5.6746</td>
<td>0.2111</td>
</tr>
<tr>
<td>50</td>
<td>4.5674</td>
<td>0.1978</td>
</tr>
<tr>
<td>60</td>
<td>3.3980</td>
<td>0.1032</td>
</tr>
<tr>
<td>70</td>
<td>3.2986</td>
<td>0.0986</td>
</tr>
<tr>
<td>80</td>
<td>3.2629</td>
<td>0.0547</td>
</tr>
<tr>
<td>90</td>
<td>3.2453</td>
<td>0.0365</td>
</tr>
<tr>
<td>100</td>
<td>3.1724</td>
<td>0.0120</td>
</tr>
</tbody>
</table>

Table 4.4  Comparison of the fitness function values

<table>
<thead>
<tr>
<th>Methods employed</th>
<th>Authors</th>
<th>Optimal best Value of ‘f’</th>
</tr>
</thead>
<tbody>
<tr>
<td>GA Based approach</td>
<td>Theodoros et al. (2014)</td>
<td>9.5612</td>
</tr>
<tr>
<td>ANN Controller</td>
<td>Dong et al. (2011)</td>
<td>7.2314</td>
</tr>
<tr>
<td>ANFIS Controller</td>
<td>Govindarajan &amp; Raghavan (2014)</td>
<td>7.1230</td>
</tr>
<tr>
<td>Proposed DE-IDWNN Controller</td>
<td></td>
<td>4.0021</td>
</tr>
<tr>
<td>Proposed Hybrid PSO-GSO SNN IZ Controller</td>
<td></td>
<td>3.8996</td>
</tr>
<tr>
<td><strong>Proposed Fuzzy SNN Controller</strong></td>
<td></td>
<td><strong>3.1724</strong></td>
</tr>
</tbody>
</table>
From Table 4.4, it can be observed that the proposed controller achieves a minimal fitness function value of 3.1724 in comparison with that of the earlier other methods available in the literature proving the effectiveness of the proposed FSNN. Figure 4.13 show the variation of fitness function with respect to iterations and from Figure 4.13 it can be inferred that the learning process enables the set function to reach a minimum value within the specified number of iterations.

Figure 4.13 Variation of fitness function with respect to number of iterations
4.6 SUMMARY

This chapter presented a proposed fuzzy based spiking neural network to control the reactive power and eliminate the additional hardware modules required for controlling the fault at the grid. Fundamentally, spiking neural network converts the given input data sample into pulses based on the firing times incurred. But the conventional process of generating pulses does not take care of the complete time interval elapsed and henceforth this thesis focused on introducing the fuzzy system with membership function values for the considered input data, wherein the input data is transformed to respective membership functions with the complete interval of time for effective generation of pulses in SNN. This incorporation has resulted in effective solutions on the active and reactive power component of the grid connected DFIG with a control on speed of the rotor as well at the time of fault. The proposed controller design is validated for a case study of wind farm with 1.5 MW wind turbines connected to a 25 kV distribution system exporting power to a 120 kV grid. The simulation results computed prove the effectiveness of the FSNN controller to be better in comparison with that of the methods available in the literature.

Even though fuzzy membership function attempts to generate spikes over a wide range of time interval, the training process of SNN overcomes the local optima but gets stuck with that of the global convergence. As a result, the training process possesses computational burden and which has to be handled. Considering this, the forthcoming work presents a replacement of SNN with ELMAN NN model which performs effective control action for the grid connected DFIG.