Intelligent Fault Detection Based on Reinforcement Learning Technique on Distribution Networks

Tlotlollo S. Hlalele [1, 2,]*, Yanxia Sun [1], and Zenghui Wang [2]

[1] Department of Electrical & Electronic Engineering Science, University of Johannesburg, Johannesburg, South Africa
[2] Department of Electrical Engineering, University of South Africa, Johannesburg, South Africa
*Correspondence: hlalets@unisa.ac.za (T.S.H.)

***Abstract*—The incorporation of distributed energy resources in the distribution networks changes the fault current level and makes the fault detection be more complex. There are several challenges brought by these heterogenous energy systems including power quality, voltage stability, reliability and protection. In this paper, a fault detection based on reinforcement learning approach is proposed. The heart of this approach is a Q learning approach which uses a non-adaptive multi-agent reinforcement learning algorithm to detect and identify nonlinear system faults, and the algorithm learns the policy by telling an agent what actions to take under what circumstances. Moreover, the Discrete Wavelet Transform (DWT) is utilized to extract coefficient values from the captured one-fourth cycle of the three-phase current signal during fault which occurs during the transient stage. The simulations and signal analysis for different faults are used to validate the proposed fault detection method in MATLAB environment. The simulation results show that different types of faults such as CA, AB, ABC and ABCG can be detected and the best correlation coefficient achieved is 0.87851.**

***Keywords*—fault detection, distributed energy resources, reinforcement learning**

I. Introduction

Due to the increasing demand of safety and reliability, Fault Detection and Identification (FDI) has received considerable attention [1]. The core of this methodology is an on-line approximator, alluded to as Fault Tracking Approximator (FTA). Uniquely in contrast to the next approximators, the FTA utilizes iterative calculations to distinguish and recognize nonlinear framework shortcomings, even within the sight of model vulnerability, which is persuaded by prescient control hypothesis and iterative learning control hypothesis. FDI has pulled in numerous scientists in late years. Multi-agent approach is seen being employed for power system recovery based on fault classification in the work of Meskina and Doggaz *et al*. [2]. This arrangement has the upside of advancing the assignment of force framework recovery. A fault detection, isolation and recovery system was employed analyse and validate the approach. Faults in electrical infrastructure of the micro-grid become more problematic in island mode as every source becomes more critical [3]. In the work of Adewole and Tzoneva *et al*. [4], Discrete Wavelet Transform (DWT) was utilized in the analysis and extraction of the trademark highlights from shortcoming transient signs of the three phase line current estimations acquired at a solitary substation relaying point, instead of the two fold finished methodology utilized in the current literature. Entropy Per Unit (EPU) records were thereafter figured from the DWT disintegration and were utilized as contribution to multi-facet ANN models filling in as FSI classifiers and FL indicators individually. A fuzzy-based intelligent fault identification and characterization scheme is produced for distribution line integrated with Distributed Generators (DG's) by Chaitanya and Yadav [5]. For this situation two unique Fuzzy Inference Systems (FIS) were demonstrated in each phase to identify the fault. The main FIS recognizes the high magnitude of fault current related with typical shunt faults and the second FiS perceives the little extent of current attributable to event of HIF. The scheme utilizes the features extracted from the Teager energy operator. In the work of Sarwar and Mehmood *et al.* [6], a precise High Impedance Fault (HIF) disclosure and detachment plot in a power dispersion network is proposed. The suggested scheme utilizes the information obtained from voltage and current sensors. An intelligent methodology for High Impedance Fault (HIF) identification in power scattering feeders using progressed signal-handling strategies, for example, time-time and time-repeat changes joined with neural network Mis introduced by Samantaray and Panigrahi *et al*. [7]. In the work of Lin and Duan *et al.* [8], a novel fault detection method based on SVDD is proposed for distribution systems with DERS. The method uses global-area data to detect outliers in the power system and effectively increases the detection accuracy. In the work of Månsson and Kallioniemi *et al.* [9], the study aimed to differentiate the data sets containing faults from data sets which are well performing substations. The application of neural network

Manuscript received August 1, 2022; revised August 25, 2022; accepted September 19, 2022; published May 15, 2023.

for the detection of fault and classification is proposed by Heo and Lee [10].

Due to the integration of dispersed energy resources into distribution networks, the amount of fault current fluctuates, making fault detection more difficult to detect and diagnose. These heterogeneous energy systems provide several issues, including those related to power quality, voltage stability, dependability, and protection. In this paper fault detection method based on reinforcement learning is described. Here various categories of faults are recognized and categorized, and an IEEE 39 bus system is used to extract the data for the faults CA, AB, ABC and ABCG, and the Discrete Wavelet Transform are used to extract features. These features are then used as states for reinforcement learning. The Artificial Neural Network is trained for Q-table in order to predict the qualities of the unknown states. The main contributions of this study are

1) A non-adaptive multi-agent reinforcement learning algorithm is developed to detect and identify nonlinear system faults of power systems;
2) The DWT is utilized to extract coefficient values from the captured one-fourth cycle of the three-phase current signal during fault;
3) The different types of faults such as CA, AB, ABC and ABCG can be detected with a good correlation coefficient.

The remaining of this paper is arranged as follows: Section II explains the Multi-agent reinforcement learning, Section III deals with fault detection and classification, Section IV provide results and analysis, and finally Section V give the conclusion.

II. Multi-agent Reinforcement Learning

The introduction of intelligent algorithms has resulted in the division of the detection systems. Engineers utilize a mathematical model of the power-grids to assess the kind of problem and then disconnect the associated breakers from the network in the traditional detection technique, which is still in use today. It is not safe to clear faults with typical relaying systems since the relays order some superfluous breakers to disconnect a healthy transmission line from the network in a variety of situations. Such disconnections are not optimal, and they result in the downing of a larger portion of the system as a result. Intelligent algorithms and learning approaches, on the other hand, have shown to be more reliable than traditional systems when compared to them. Fuzzy logic, and genetic algorithms are all methodologies that are commonly employed in the identification and diagnosis of power grid faults and outages.

Reinforcement learning is a method of training machine learning models. The agents in this methodology are taught to attain objectives, and the system will come up with a solution via trial and error. It determines performance penalties and rewards, with the primary goal of maximizing rewards in an unpredictable and possibly complicated environment. In this approach, a game-like circumstance occurs, and artificial intelligence is confronted with those situations in reinforcement learning. The game's rule is to solve it without offering any hints or suggestions, which is the designer's reward philosophy. The model starts with random trails and progresses to complex methods for completing the job and maximizing the rewards. Reinforcement learning differs from supervised learning in that supervised learning requires labeling for input or output as well as suboptimal behaviors to be corrected in a clear and precise manner, whereas reinforcement learning does not [11]. The balance between exploitation and exploration is the goal of reinforcement learning. Many reinforcement learning techniques for this context use dynamic programming paradigms since the environment is defined by a Markov decision process (MDP). The difference between traditional dynamic programming techniques and reinforcement learning algorithms is that it does not need an understanding of a precise mathematical model of the Markov decision process and is meant for big MDPs where accurate approaches are unfeasible. Reinforcement learning requires early exploratory techniques, such as randomly selecting actions without utilizing a probability distribution that has been calculated and demonstrates poor performance.

A. Environment

The environment wherein the agent works and is influenced. The agent's present state and action are fed into the environment, which then outputs the both reward and the owner's future state. If you're the actor, the environment could be the physical and social rules that govern how your actions are processed as well as what happens next as shown in Fig. 1.

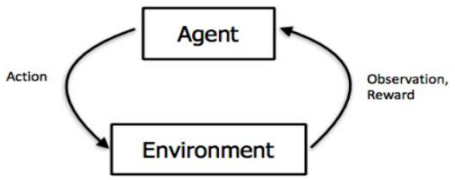

Figure 1. Reinforcement learning model [12].

B. Agent

Super Mario navigating a computer game or a drone delivering goods are examples of agents. The agent is the algorithm/program that make a decision based on its previous experiences and its environment.

C. Reward

Benefits are the arithmetical criteria that the agent achieves for executing a certain activity at a specified state(s) in the environment. The numeric number may be particularly dependent on the agent's activity. We focus on increasing the value the agent obtains from the present state rather than maximizing the cumulative reward in reinforcement learning.

D. Action

An action is a set of all the potential moves that the agent can do. Agents generally pick from a list of different, potential actions, even though the action is self-explanatory.

In this paper, we are using reinforcement learning for fault detection and classification. This fault analysis helps to increase the productivity of the smart grid. A smart grid is generally an energy network that allows for the two-way flow of power and data, as well as the use of digital communications technology to detect, react, and pro-actively respond to changes in use and a variety of other concerns. Smart networks can self-heal and empower electricity users to take an active role in their energy supply. Unless otherwise specified, any equipment connected to the electric utility system is intended to be utilized within the specified voltage range. Voltage drops may be found in every component of the system. When compared to customers who are situated at the tail end or far end of the distribution system, consumers who are electrically linked on the major distribution feeder near the substation will have the highest voltage levels (higher voltage drops attributes to the poor voltage levels).

Algorithm 1: Non-adaptive Multi-agent reinforcement algorithm, for resource allocation

1. Initialize
2. i=0
3. **For** all states $s_j \in S_j, a_j \in A_j$ do
   Initialize the allocation strategy,
   $\pi^i(s_j, a_j), Q^i(s_j, a_j)$
   end for
4. Evaluate the states at step $i, s_j = s_j^i$
5. **While** (True) do
   Select an action according to using the Boltzmann distribution (1)
   If $a_j^i == class_j^i$
   $R(s_j^i, a_j^i) = 1$
   else
   $R(s_j^i, a_j^i) = 0$
   end
6. Update $Q^{i+1}(s_j, a_j), \pi^{i+1}(s_j, a_j)$
7. Increment t by 1 and update the states as $s_j^i = s_j^{i+1}$
8. **end while**

where,

a_j^i = i^{th} action of a j^{th} agent

s_j^i = i^{th} state of the j^{th} agent

s_j^{i+1} = i+1th state of the j^{th} agent

S_j =set of states of a j^{th} agent

A_j =set of actions of the j^{th} agent i= time

$$\pi^i(s_j, a_j) = \frac{e^{Q^i(s_j, a_j)/\tau}}{\sum_{a \in A} e^{Q^i(s_j, a_j)/\tau}} \quad (1)$$

$$Q_j^{i+1}(s_j, a_j) = (1-\alpha) Q_j^i(s_j, a_j) + \alpha^i \left\{ \sum_{a_{-i} \in A_{-i}} \left[Z(s_j, a_j, a_{-j}) \prod_{x \in U/\{u\}} \pi_x^i(s_x, a_x) \right] + \gamma \max_{a_j \in A_j} Q_j^i(s_j', a_j) \right\} \quad (2)$$

The agent is not obliged to interact, but they must be able to monitor the effected joint actions and the received separate gain. According to Drugan's work [13], Markov Decision Process is characterized by Number of states $S = \{s_1, s_2, \dots, s_n\}$ where s_t is a state in S; Number of actions $A = \{a_1, a_2, \dots a_M\}$ accessible to the agent per state s ; Alteration dissemination $T(s'|s, a)$, records a set comprised of a state s and an action a to a prospect dissemination of state s';

A reward function $R: S \times A \times S \to R$ provides the probable reward when the agent builds the alteration from state s to state s' via action a. r_t represents the instant scalar reward gained at time t, where

$$r_t = R(S_{t+1} = s' | s_t = s, a_t = a)$$

$$= E\{r_t | s_{t+1} = s', s_t = s, a_t = a \quad (3)$$

The dissemination of the resulting states and rewards is autonomous of the historical through the present state and action, such that

$$T(s_{t+1}|s_t a_t) = T(s_{t+1}|s_t, a_t, \dots\dots, s_1 a_1) \quad (4)$$

The action selection mechanism in MDP is $policy\ \pi: S \times A \to [0,1]$ that stipulates a prospect of choosing a in an exact s. The probable return in a state s.

$$V^\pi(s) = E_\pi[R_t | s_t = s] =$$

$$E_\pi[\sum_{t=0}^{\infty} \gamma^t . r_t | s_t = s] \quad (5)$$

where γ is the discount factor and R_t signifies the gain. $V^\pi$is the gain of an agent resulting the policy π. The action value function for policy π, $Q^\pi(s, a)$, is the anticipated gain when acquiring action a in state s under the policy π. Therefore $Q^\pi(s, a) = E_\pi[R_t | s_t = s, a_t = a]$. The MDP's goal is to discover the preeminent policy π^* that exploits the probable gain. The optimal s_{value} for any state $is\ V^*(s) = max_\pi V^\pi(s)$.

Strategies used to find optimal policies, i.e., classification criterion for reinforcement learning approach are as follows;

- $Value\ iteration$ updates each iteration according to the policy given value function such that the current value function updates to intuitive.

$$V_{t+1}^{\pi}(s) = \max_{a \in A} \frac{\sum_{s' \in S} T(s'|s,a)}{(R(s'|s,a) + \gamma . V_t^{\pi}(s'))} \quad (6)$$

- Policy reiteration progresses the feature of the policy π over, after assessing the value function $V^{\pi}\ of\ the\ fixed\ policy\ \pi$.

Direct policy search. Here, it is not necessary to realize the value function.

For independent learning;

$$Q_i(s,a_i) = Q_i(s,a_i) + \alpha[R_i(s,a) + \gamma max\ a'_i\ Q_i(s',a'_i) - Q_i(s,a_i)] \quad (7)$$

In this case the agent overlooks the actions and gains of other agents.

For coordinated reinforcement learning, the agent must harmonize its action with a few agents and acts self-sufficiently within the environment.

$$Q_i(s_i,a_i) = Q_i(s_i,a_i) + \alpha[R(s,a) + \gamma \max a'\ Q(s',a') - Q(s,a)] \quad (8)$$

This method is disseminated and generates to enormous storage and computational savings in the action space. Distributed value functions;

$$Q_i(s_i,a_i) = (1-\alpha)Q_i(s_i,a_i) + \alpha[R_i(s,a)\ \gamma \sum_{j \in (f(i,j) \neq 0} f(i,j) \max a'_i] \quad (9)$$

III. Fault Detection and Classification

According to the characteristics of each fault type, various fault types happen on distribution network due to varying quantities of change in currents and voltages on the load bus and Distributed Generation (DG) bus in comparison to normal conditions. To identify fault types that happened just on the distribution line, the proposed method only required a current signal obtained from the substation, DG, and load bus. A fault classification system was developed using fault current signals with distinct properties for each kind of failure. Discrete wavelet transform (DWT) was utilized to extract coefficient values from the captured one-fourth cycle of the three-phase current signal during fault which occurs during the transient stage. These values then were matched to the negative sequence elements' DWT coefficient to construct a decision tree parameter. Parameters used in fault classification are all standardized to the same wavelet scale. The following are the parameters that are being considered:

A. Maximum Parameter at Fault Occurrence

If the distribution systems are under fault constraint, the highest coefficient of DWT of three-phase signals at one-fourth cycle is used.

On each phase and zero sequence signal, the maximum parameter is as follows:

- The phase A component's highest coefficient is A_{max}.
- The phase B component's highest coefficient is B_{max}.
- The phase C component's highest coefficient is C_{max}.
- The highest coefficient of the zero sequence components is called Z_{max}.

B. Comparison Parameter

In each step, the comparison parameter is utilized to identify the state of a fault condition.

When the value is greater than the comparative value, this parameter is used to define the issue phase. It may be calculated by dividing the greatest coefficient of each component even by zero-sequence elements in each phase. On each step, the comparison parameter is as follows:

A_{com}= in phase A, the comparison parameter is used to discover faults.

B_{com}= in phase B, the comparison parameter is used to discover faults

C_{com}= in phase C, the comparison parameter is used to discover faults

C. Check Parameter

In the decision tree, the check parameters are utilized to identify the fault kinds. These parameters are derived from the comparison parameter's highest value. The following is a list of the check parameters:

Ph_{max} is the greatest value from the comparison A_{com}, B_{com}, C_{com}.

Ph_{min} is the least value from the comparison A_{com}, B_{com}, C_{com}.

A three-phase fault is identified by comparing the lowest and highest check values. A comparison variable for each phase is used to identify the phase where the problem occurred. A ground signal faults if the negative sequence components following the fault are five times higher than normal, and the largest source current component is 10–12.

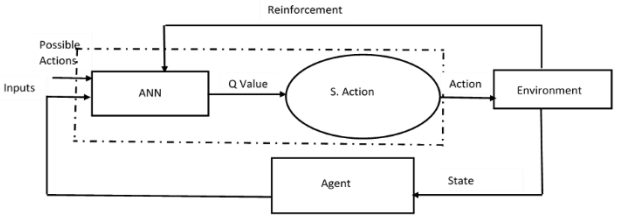

Figure 2. The structure of reinforcement learning based on ANN.

In Fig. 2, the agent tries to control the environment in the reinforcement learning problem. From one state to the next, if the agent succeeds, it will be rewarded, but if they fail it will not be rewarded. To develop a sheet for our

agent, we used a Q-learning which is simple yet effective approach.

IV. Analysis and Results

Due to the integration of dispersed energy resources into distribution networks, the amount of fault current fluctuates, making fault detection more difficult to detect and diagnose. These heterogeneous energy systems provide several issues, including those related to power quality, voltage stability, dependability, and protection. Faults of various categories are recognized and categorized accordingly.

The process of fault detection is such that the inputs to the network are three phase voltage and currents. ABC denotes the three phases of the distribution lines and G denotes the ground.

A. For Fault C

The algorithms and progress are shown in the diagram in Fig. 3 for fault at C. A random data-division method was used to model the neural network in this case. The mean squared error has been implemented to evaluate the model's performance. The model has been implemented over 19 epochs, and the gradient descent value was 1.00e−06.

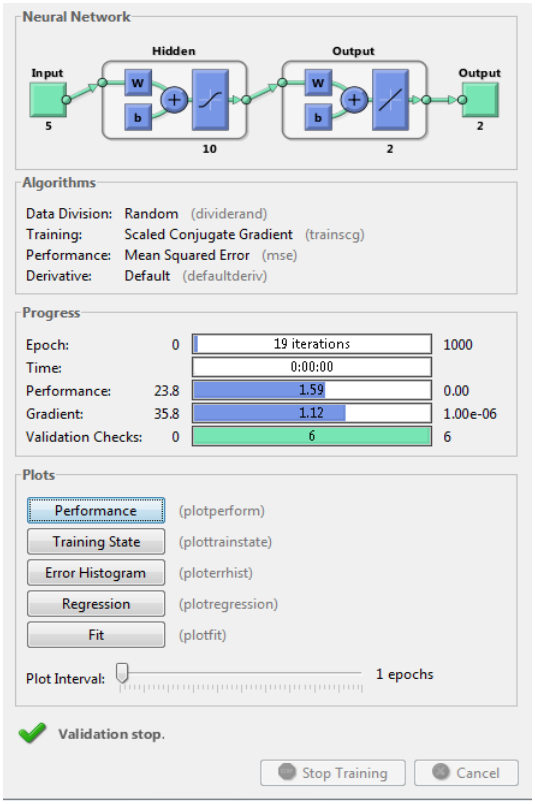

Figure 3. Training process.

The model's best validation performance is depicted in the graph in Fig. 4. Mean Squared Error is the statistical metric used to assess performance. The best epoch value is the one with the least mean square error.

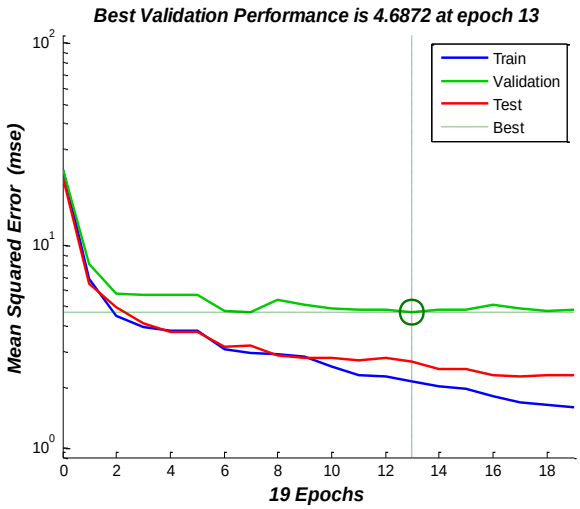

Figure 4. Validation performance.

B. For Fault AB

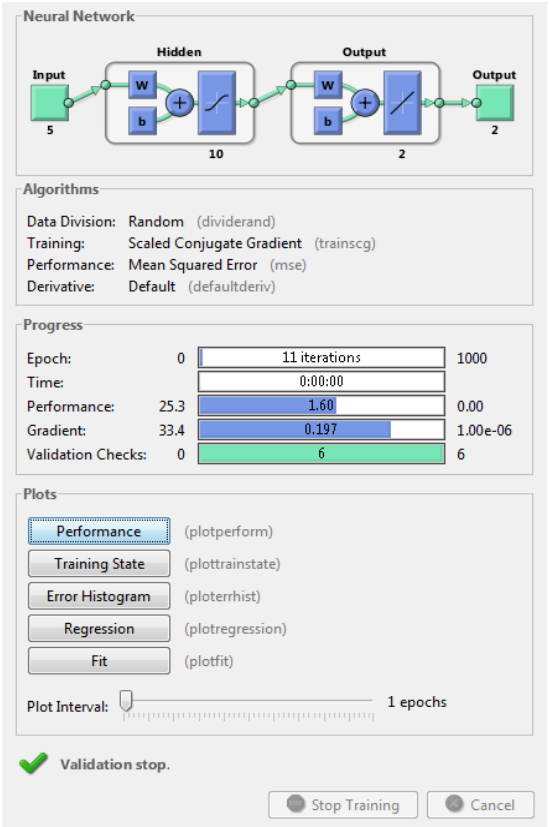

Figure 5. Training process.

The algorithms and progress are shown in the diagram in Fig. 5 at fault AB. A random data-division method was used to model the neural network in this case. The mean squared error has been implemented to evaluate the model's performance. The model has been implemented over 11 epochs, and the gradient descent value was 1.00e−06.

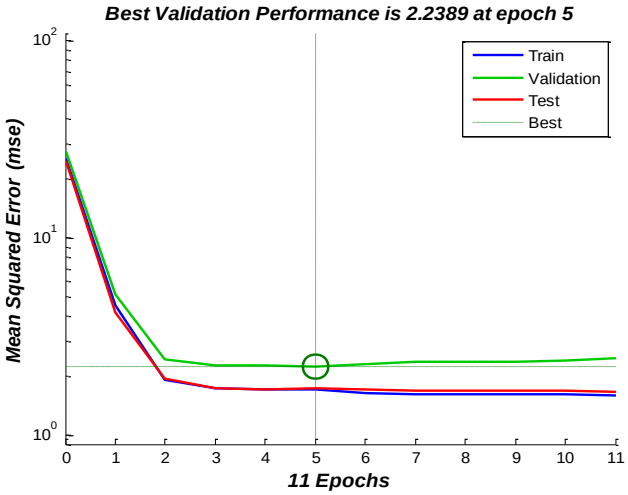

Figure 6. Validation performance.

The model's best validation performance is depicted in the graph above in Fig. 6. Mean Squared Error is the statistical metric used to assess performance. The best epoch value is the one with the least mean square error.

C. For Fault BC

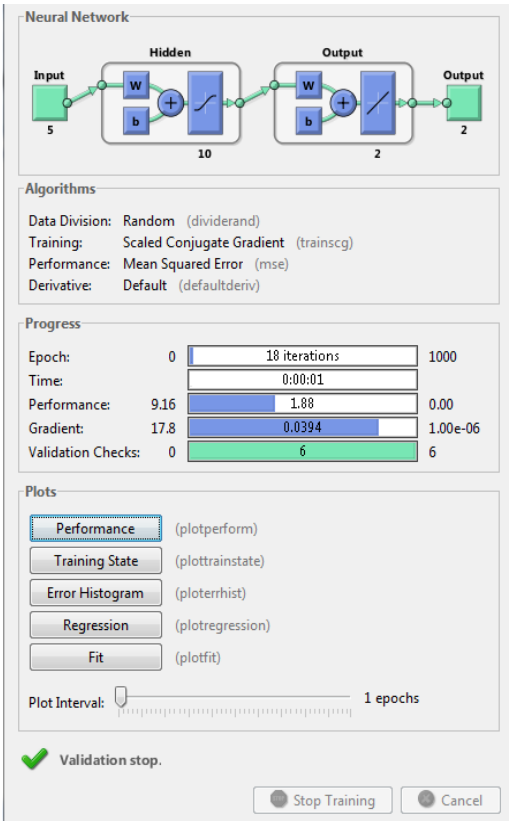

Figure 7. Training process.

The algorithms and progress are shown in the diagram above in Fig. 7 for fault at BC. A random data-division method was used to model the neural network in this case. The mean squared error has been implemented to evaluate the model's performance. The model has been implemented over 18 epochs, and the gradient descent value was 1.00e−06.

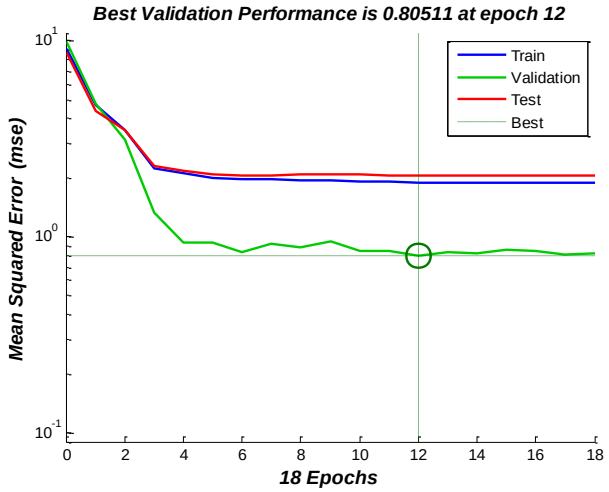

Figure 8. Validation performance.

The model's best validation performance is depicted in the graph above in Fig. 8. Mean Squared Error is the statistical metric used to assess performance. The best epoch value is the one with the least mean square error.

D. Fault CA

A random data-division method was used to model the neural network in this case. The mean squared error has been implemented to evaluate the model's performance. The model has been implemented over 54 epochs, and the gradient descent value was 1.00e−06.

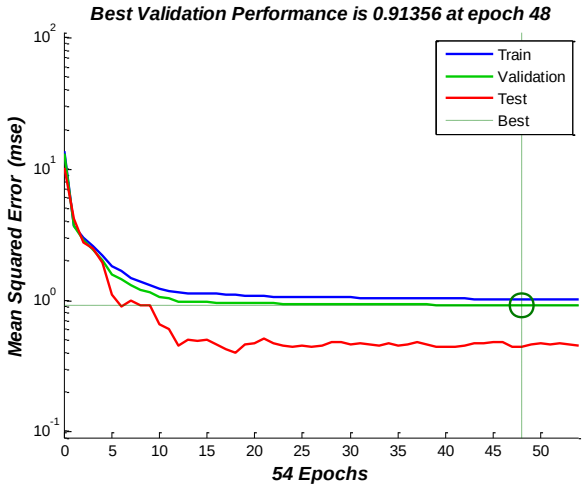

Figure 9. Validation performance.

The model's best validation performance is depicted in the graph above in Fig. 9. Mean Squared Error is the statistical metric used to assess performance. The best epoch value is the one with the least mean square

E. Fault ABC

The algorithms and progress are shown in the diagram in Fig. 10. A random data-division method was used to model the neural network in this case. The mean squared error has been implemented to evaluate the model's performance. The model has been implemented over 56 epochs, and the gradient descent value was 1.00e−06.

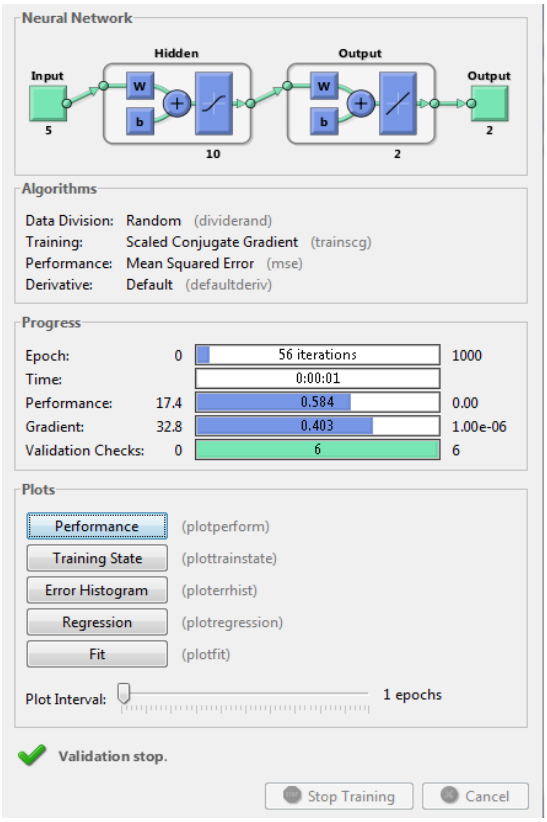

Figure 10. Training process.

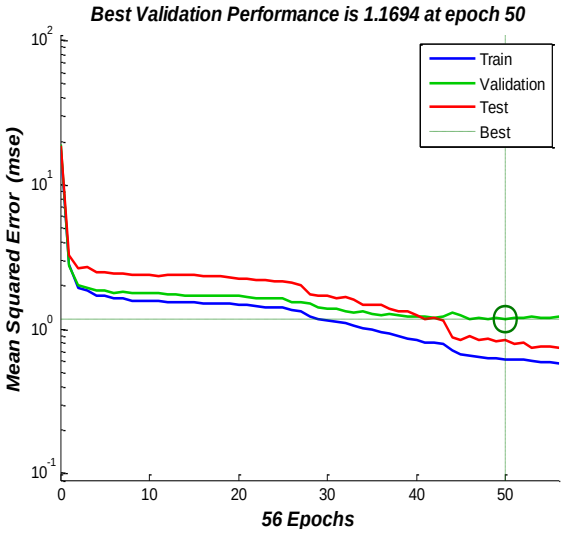

Figure 11. Validation performance.

The model's best validation performance is depicted in the graph above in Fig. 11. Mean Squared Error is the statistical metric used to assess performance. The best epoch value is the one with the least mean square error.

F. Fault ABCG

The algorithms and progress are shown in the diagram in Fig. 12. A random data-division method was used to model the neural network in this case. The mean squared error has been implemented to evaluate the model's performance. The model has been implemented over 25 epochs, and the gradient descent value was 1.00e−06.

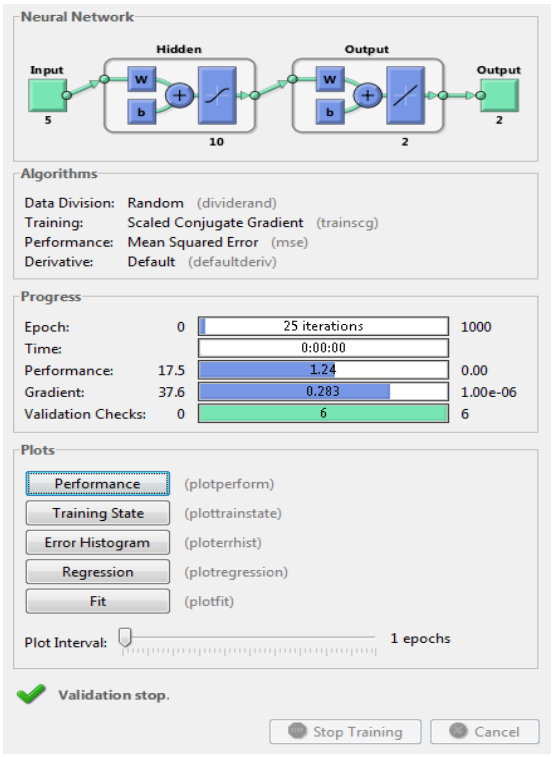

Figure 12. Training process.

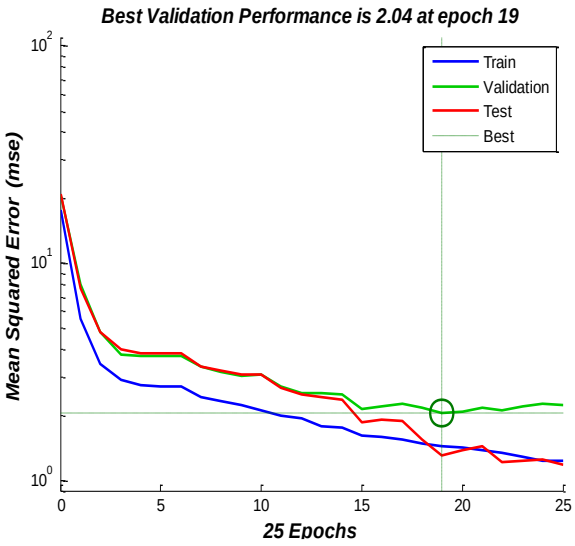

Figure 13. Validation performance.

The model's best validation performance is depicted in the graph above in Fig. 13. Mean Squared Error is the statistical metric used to assess performance. The best epoch value is the one with the least mean square error.

After training a feedforward neural network, the above histogram in Fig. 14 shows the errors between target and predicted values. These error values are negative because they indicate how predicted values differ from target values.

The number of samples from the dataset that fall into each bin is represented on the Y-axis. The zero-error line corresponds to the error axis's zero error value. In this case, the zero error point is contained within the bins of −0.1276 and 0.13.

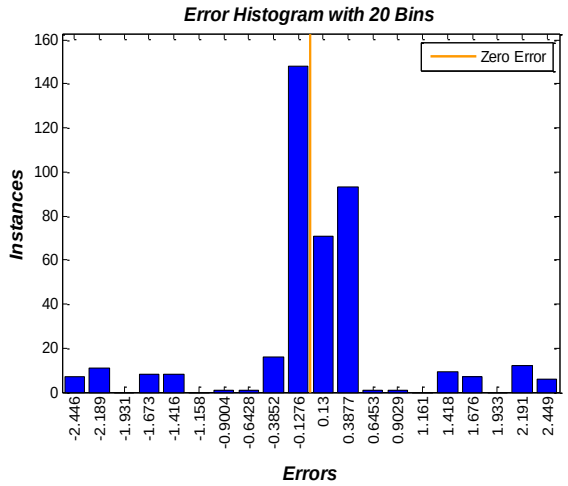

Figure 14. Instances versus errors.

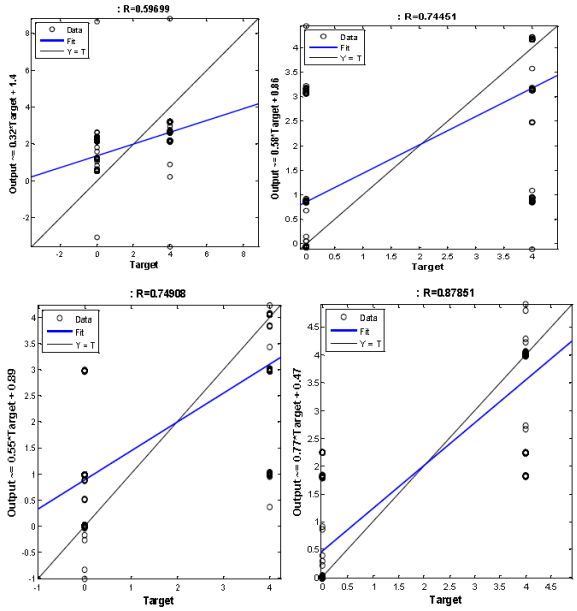

Figure 15. Regression plots of output versus target for the network.

The graphs in Fig. 15 above shows the linear relationship between the output and the target. When there is a relationship, the regression best fit line is sloped at an angle. The line depicts the strength of the relationship between the output and target variables.

TABLE I. PERFORMANCE COMPARISON BETWEEN RL METHOD AND ANN

RL		ANN	
R Value	Output	R Value	Output
0.59699	0.32	0.44199	0.29
0.74451	0.58	0.62204	0.47
0.74908	0.55	0.63502	0.45
0.87851	0.77	0.73700	0.69

In Table I, the correlation coefficients whose magnitude is between 0.3 and 0.5 indicate variable with a low correlation and the magnitude between 0.5 and 1 indicate a good correlation [13]. The best correlation coefficient for RL was found to be 0.87851, which indicate the satisfactory correlation between the targets and the output.

V. CONCLUSION

In this paper we have studied the application reinforcement learning for fault detection and classification in three phase distribution networks. To identify fault types that happened just on the distribution line, the proposed method only required a current signal obtained from the substation, DG, and load bus. A fault classification system was developed using fault current signals with distinct properties for each kind of failure. Discrete wavelet transform (DWT) was utilized to extract coefficient values from the captured one-fourth cycle of the three-phase current signal during fault occurs during the transient stage. The simulation results obtained prove that the satisfactory performance has been achieved and that the proposed method is practically implementable in that the different faults type have been detected such as AB, BC, ABC etc. The best validation performance achieved at 56 epochs is 1.1694 and the best correlation coefficient is 0.87851. The proposed method appears to be better when compared to other method where ANN is implemented for fault detection in power transmission network without distributed energy generation sources. The best validation performance achieved at 56 epochs here was 5.8095. In some literature, fuzzy logic has been employed for fault detection and classification, however the efficiency of this system is not high when compared to the proposed method because it majorly works on inaccurate inputs.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AUTHOR CONTRIBUTIONS

T. S. Hlalele conducted research and wrote the article. Z. Wang and Y. Sun reviewed and revised the article. All authors approved the final version of the paper.

FUNDING

This research is supported partially by South African National Research Foundation Grants (No. 120106, 141951 and 132797), South African National Research Foundation Incentive Grant (No. 132159), and GES fund of University of Johannesburg.

REFERENCES

[1] B. Yan, H. Su, and W. Ma, "Fault detection and identification for a class of nonlinear systems with model uncertainty," *Appl. Math. Model.*, vol. 40, no. 15–16, pp. 7368–7381, 2016.

[2] S. B. Meskina, N. Doggaz, and M. Khalgui, "An efficient simulator for fault detection and recovery in smart grids FDIRSY," in *Proc. 5th Int. Conf. Pervasive Embed. Comput. Commun. Syst.*, 2015, pp. 132–139.

[3] J. Hare, X. Shi, S. Gupta, and A. Bazzi, "Fault diagnostics in smart micro-grids: A survey," *Renew. Sustain. Energy Rev.*, vol. 60, pp. 1114–1124, 2016.

[4] A. C. Adewole, R. Tzoneva, and S. Behardien, "Distribution network fault section identification and fault location using wavelet entropy and neural networks," *Appl. Soft Comput. J.*, vol. 46, pp.

296–306, 2016.
[5] B. K. Chaitanya and A. Yadav, "An intelligent fault detection and classification scheme for distribution lines integrated with distributed generators," *Comput. Electr. Eng.*, vol. 69, no. September 2017, pp. 28–40, 2018.
[6] M. Sarwar, F. Mehmood, M. Abid, A. Q. Khan, and S. T. Gul, "High impedance fault detection and isolation in power distribution networks using support vector machines," *Journal of King Saud University—Engineering Sciences*, vol. 32, no. 8, pp. 524–535, 2020.
[7] S. R. Samantaray, B. K. Panigrahi, and P. K. Dash, "High impedance fault detection in power distribution networks using time-frequency transform and probabilistic neural network," *IET Generation, Transmission & Distribution*, vol. 2, issue 2, pp. 261–270, 2008.
[8] Z. Lin, D. Duan, Q. Y, X. Cheng, L. Yang, and S. Cui, "One-class classifier based fault detection in distribution systems with distributed energy resources," in *Proc. 2018 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, 2018, pp. 932–936.
[9] S. Månsson, P. J. Kallioniemi, K. Sernhed, and M. Thern, "A machine learning approach to fault detection in district heating substations," *Energy Procedia*, vol. 149, pp. 226–235, 2018.
[10] S. Heo and J. H. Lee, "Fault detection and classification using artificial neural networks," *IFAC-PapersOnLine*, vol. 51, no. 18, pp. 470–475, 2018.
[11] M. R. Tousi, S. H. Hosseinian, and M. B. Menhaj, "A multi-agent-based voltage control in power systems using distributed reinforcement learning," *SIMULATION*, vol. 87, no. 7, pp. 581–599, 2011.
[12] H. C. Kiliçkiran, B. Kekezoglu, and N. G. Paterakis, "Reinforcement learning for optimal protection coordination," in *Proc. International Conference on Smart Energy Systems and Technologies (SEST)*, 2018.
[13] M. M. Drugan, "Reinforcement learning versus evolutionary computation: A survey on hybrid algorithms," *Swarm Evol. Comput.*, vol. 44, pp. 228–246, 2019.