ADSUJSR 13 (1) (2025) 89 – 107
Adamawa State University Journal of Scientific Research
(ADSUJSR)
http://www.adsujsr.edu.ng
ISSN: 2705-1900 (Online); ISSN: 2251-0702 (Print)

DESIGN AND SIMULATION OF A FAULTS PREDICTION AND LOCALIZATION MODEL ON TRANSMISSION LINE USING ARTIFICIAL NEURAL NETWORK

[a]Mohammed A., [b]Danladi A.
Department of Pure and Applied Physics
Adamawa State University, Mubi.
Corresponding Author's Email: aminuclube2020@gmail.com

Received in July 2025
Accepted in August 2025

Abstract

This study presents the design and development of a Fault Prediction and Localization Model for transmission lines using Artificial Neural Networks (ANNs). The model predicts, localizes, and notifies precise fault locations, enhancing the reliability and efficiency of power transmission systems. The research work focuses on modeling the NIPP 218 km transmission line from Egbin to Benin and utilizing an artificial neural network (ANN) with back propagation to identify and locate various faults on the transmission line. Different faults including single line to ground, double line to ground, and balanced faults were simulated at a distance of 20 km from the sending end. The ANN was trained to identify the type of fault and predict its location. MATLAB SIMULINK was used to simulate the system. Results show that the ANN correctly identified all fault types and predicted fault locations with an accuracy of approximately 99.55%. The research demonstrates the capability of ANNs in fault identification and location prediction in power transmission lines when properly trained.

Keywords: Fault Prediction, Fault Localization, Transmission Lines, Artificial Neural Networks (ANNs), Smart Grids

INTRODUCTION

Power Transmission commences with transportation of 330 kV voltages via transmission lines to the transmission sub-station where it is step-down via transformer to 132 kV, this is then transported again via transmission lines to injection substations for further step-down to 33 kV and at this point electricity distribution to the end-users begins (Ananthan et al., 2017).

According to Buckminister (2016), the Nigerian levels of voltage transmission ranges from 330 kV, 132 kV to 33 kV, and these different voltage transmission levels enhanced transmission network interconnection which is referred to as the national grid.

Transmissions of power usually require various components to ensure power reliability and access. Transmission lines are among the power system components with the highest fault incidence rate that are beyond the control of man, since they are exposed to the environment (Babu et al., 2011). Balanced condition of power system exists at normal operation, this balanced condition is offset when fault occurred, faults on the power system can be caused by several factors such as blowing of wind, collapse of a tree, failure due to mechanical fault in transformer, lightning, storms, fog, salt spray on dirty insulators and a host of others (Adepoju et al., 2013).

There is no fault-free system and it is neither practical nor economical to build a fault-free system. Power failure occurs when there is a disturbance in the electrical power transmission/distribution line (Shalangwa et al., 2009). The various cases of abnormal circumstances such as natural events, physical accidents, equipment failure, and disoperation generate faults in the power system. The consequences of faults are traumatic amplification of current flow, increasing heat produced in the conductors leading to the major cause of damage (Shalangwa et al., 2009).

Yadav and Dash (2014) said the rapid expansion in the power system has led to substantial growth in the number of transmission overhead lines in service over the past decades. These lines encounter faults that must be mitigated before returning the line to operation to recover the consumers' connection. Existing fault detection mechanisms (relays and digital fault recorders) are slow to communicate the fault characteristics upstream to the substations and control centers for action to be taken quickly.

Xiao et al. (2018) described Fault as a massive flow of current that could harm the machinery. When the current is high, the voltage is low and the electricity throughout the system can be disrupted, which may influence the protection of devices. Often, voltage less than its minimal level may cause apparatus malfunction. To provide network safety, it is essential to analyze an electrical system during defective situations (Xiao et al., 2018). Regarding the delivery network, the most critical aspect of power lines is the electrical transmission from the generation plant to the loading centers. The power systems experts have become the subject for detection and analysis of faults since the establishment of the delivery and transmission system (Panteli & Mancarella, 2015).

Existing fault detection mechanisms (relays and digital fault recorders) are slow to communicate the fault characteristics upstream to the substations and control centers for action to be taken quickly (Mrabet et al., 2022).

Zhang et al. (2020) said fault location is one of the most essential techniques to maintain the stable operation of power systems. A fast and accurate fault location allows operators to restore power grids faster and avoid economic losses. For large systems, more coupling effects of transmission lines require more complex feature engineering, and incomplete features can easily introduce large errors.

To overcome this, it was encouraged that a deep learning approach without manual feature extraction should be introduced to the fault localization models. It is a challenging task for power system operators (PSO) to supply uninterrupted electric power to end-users. Although fault intrusion is beyond human control, it is essentially important to accurately detect, classify and locate the fault location (Atuchukwu et al., 2023).

Mrabet et al. (2022) further explained that the Impedance-based faults detection system on electrical transmission lines has several limitations, including sensitivity to system conditions, limited fault detection capability, inability to pinpoint fault location, and vulnerability to network topology and inaccurate system parameters.

The existing fault detection mechanisms in electrical power transmission systems are inadequate, leading to delayed identification and location of faults, resulting in significant costs, wasted energy, and resources. The inability to accurately predict and pinpoint fault locations in real-time hinders the efficient restoration of power, compromising the reliability and trustworthiness of the power system. Therefore, there is a need for a more effective and efficient fault prediction and location system to minimize downtime, reduce costs, and ensure the reliable operation of power transmission systems.

In this research work, the transmission line will be modeled with blocks of MATLAB/Simulink, an ANN architecture will be design and develop using MATLAB's Simulink, suitable for faults prediction and localization. Data for all possible electrical faults on transmission line including types, locations for each fault, relevant parameters (i.e., impedance, voltage, current, temperature) will be gathered and extract relevant features. This data will be generated and store programmatically on every 1cm of the modeled transmission line to cover all the kilometers length of the Line. The generated data will be used to train ANN to validate its performance using metrics such as accuracy, precision and recall. Then trained ANN model will predict, localize any fault scenario impose on the transmission line and give the precise fault characteristic and location. The neural network (NN) is an intelligent data management system that uses back propagation algorithms to provide high-accuracy predictions (Abhishek et al., 2012).

Zhou and Cao (2018) further described the layers as, input neurons in the first layer accept data and convey it without modification via synapses to the second layer

(Hidden layer) of neurons, hidden layer process inputs using linear combinations and activation functions enhancing the network's expressive power (e.g., ReLU, Sigmoid, Tanh, and Softmax). Introduce non-linearity, enabling the network to learn complex relationships and the third layer of output neurons it generates final predictions or outputs, often using a softmax activation function, produce probabilities or classifications for multi-class problems.

The aim of this research is to design a Faults Prediction and Localization Model using Artificial Neural Network (ANN) for transmission lines. The specific objectives are: to develop a Simulink model of the Egbin to Benin transmission line, develop an ANN algorithm for fault localization and prediction and to simulate, localize, and predict power system faults on the transmission line model.

MATERIALS AND METHODS

Materials

The research works models' transmission line of a power system network, designs an artificial neural network to detect and identify faults on the line. Therefore, in this research work, part of the 330 kV Nigerian National Integrated Project (NIPP) is employed as a case study. The longest transmission line in the network, which is the line from the Egbin generating station (bus 10) to Benin transmitting station (bus 6), is employed as the case study. Figure 3.1 presents the single line diagram of the longest line in the NIPP power system network.

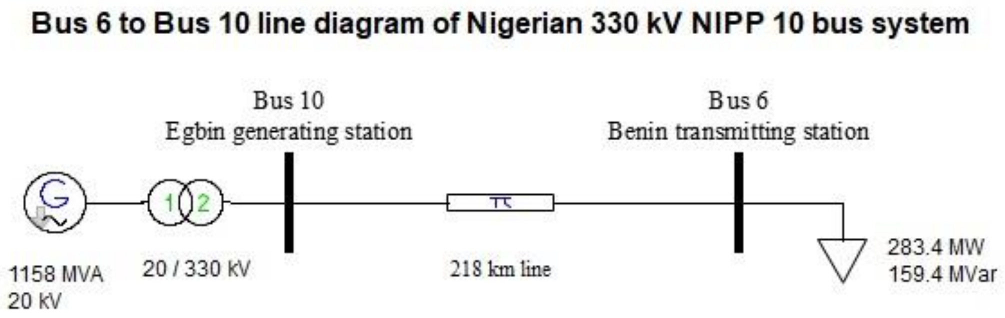

Figure1: Bus 6 to Bus 10 of the NIPP network

The length of the longest line in the NIPP network is the 218 km line. This makes any research work on this line worthwhile. The line will be modelled using MATLAB SIMULINK's power system library (*powerlib*) installed in a Hewlett Packard computer with a processor speed of 2.5 GHz and 8 Gbyte of installed RAM. Information about the 218 km NIPP line is presented in table1.

Table 1: Line Parameters for the NIPP 218 km line

Node		Length of Line	Resistance	Inductance
From	To	(Km)	(Ω/km)	(H/km)
Bus 6	Bus 10	218	0.01273	0.9337 x 10^{-3}

Table 2: Load parameter for the NIPP 218 km line

Bus Number	Bus type	Active Power (MVA)	Reactive power (MVar)
6	Slack	-	-
10	Load	283.4	159.4

Methods

The technique that is employed in order to achieve the milestone activities of this research work is hereby presented:

i. Develop the NIPP 218 km line model using the power system library in MATLAB SIMULINK
ii. Design the Artificial Neural network for fault identification and location
iii. Train the ANN
iv. Simulate the entire design

Simulink model of the NIPP 218 km line (base case)

The NIPP 218 km line Simulink model is presented in figure 2

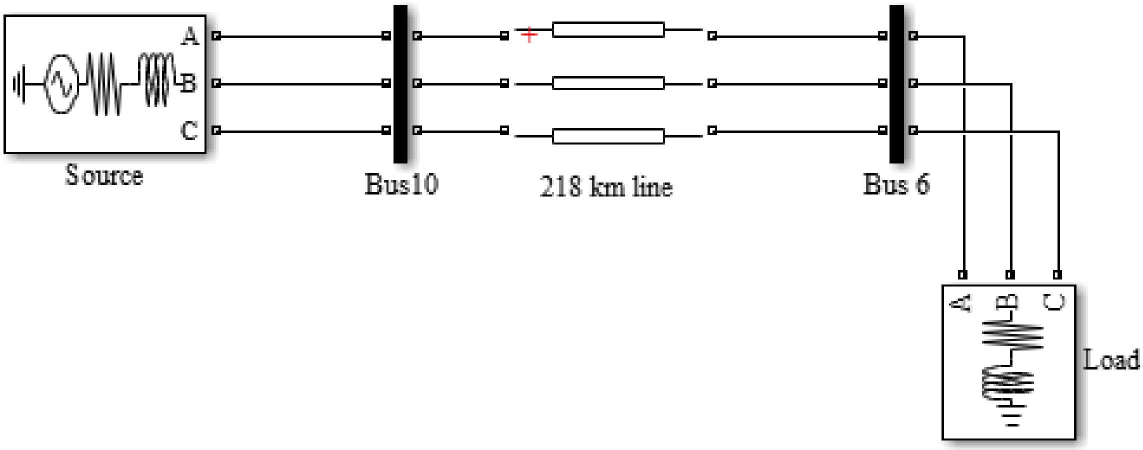

Figure 2: Simulink base case model of the NIPP 218 km line

Figure 2 presets a three phase Simulink model of the NIPP 218 km line. The model comprises of several blocks that includes the source block, the bus block, the line block and the load block.

The source block models the power generation at the Egbin generating station and the transformer. To model the information about the generating station and the transformer, the source block is doubled clicked and information presented in table 2.

The bus block is used to model the type of bus, that is, if the bus is a slack, generator or load bus. In this research work, the bus types are the slack and load busses. Therefore, in the dialog box of busses 10 and 6, the slack and load bus is entered respectively.

The 218 km line models the transmission line of the NIPP network. Information about the length of the line, the resistance and inductance of the line, as presented in table 3.1 is entered into the dialog box of the line block.

The load block models the load that is attached to the load bus (bus 6). Information about the load, presented in table 2, is keyed into the dialog box of the load block.

This model presented in figure 2.0 is without fault and therefore constitute the base case model. However, in this research work, the different power system faults will be simulated. To achieve this, the model of figure 2.0 is modified and presented in figure 3.0

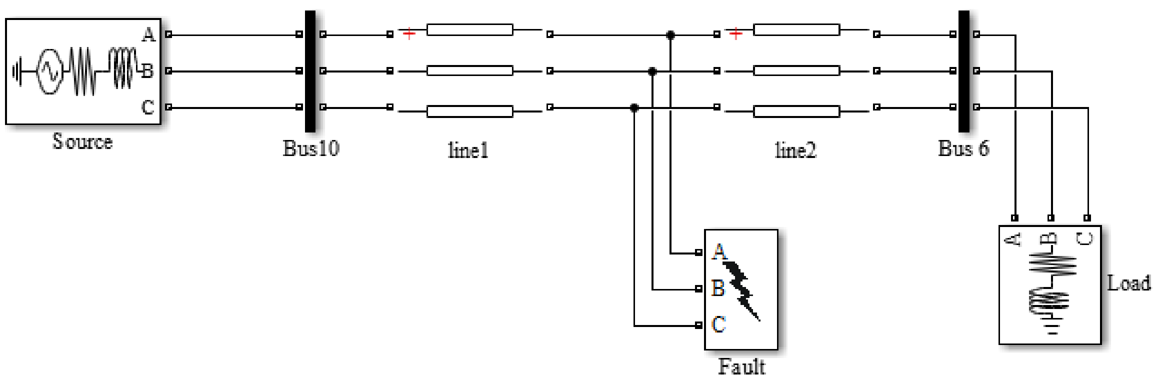

Figure 3: Simulink model of the NIPP 218 km line with fault block

The model presented in figure 3 has the 218 km line split into two. If a scenario where a fault occurs at 15 km from the sending end of the transmission line (bus 10) is considered, then to simulate this scenario, the transmission line will be split into two. The first line (line 1) will be set at 15 km while the second line (line 2) will be set at 203 km (218 km minus 15 km).

The fault block is introduced at the point the fault occurs. This means it is place at 15 km from the sending end (bus 10). In the dialog box of the fault block, the type of fault intended to be simulated is entered. Also, the time the fault occurs and fault levels are all entered. When a fault occurs, huge amount of fault current begins to flow. In this research work, information about the fault current is collected at different locations of the line. This information is collected by programmatically placing the fault block at about 10 cm at a time, throughout the length of the 218 km line. This is done programmatically since placing the faults at this minute distance (10 cm at a time) manually will be extremely difficult, inconvenient and time consuming. The information about the fault currents collected will now be used to train an artificial neural network (ANN) that will be used to identify the location of the fault.

Artificial Neural Network Design

In this research work an artificial neural network (ANN) will be designed. Whenever there is a fault in the 128km line, a fault current is drawn into the system. The magnitude of the current depends on or is a function of the type of fault and its location on the 128 km line. The means that a neural network that is capable of function approximation will be suitable for this purpose.

The design of the ANN is presented in the following manner:

i. Multiple input single neuron design
ii. Layer of Neurons
iii. Multiple layers of Neurons

Multiple input single neuron

Typically, a neuron has more than one input. A neuron with R inputs is shown in Figure 4. The individual inputs $p_1, p_2, \ldots p_R$ are each weighted by corresponding elements $w_{1,1}, w_{1,2}, \ldots w_{1,R}$ of the weight matrix **W**.

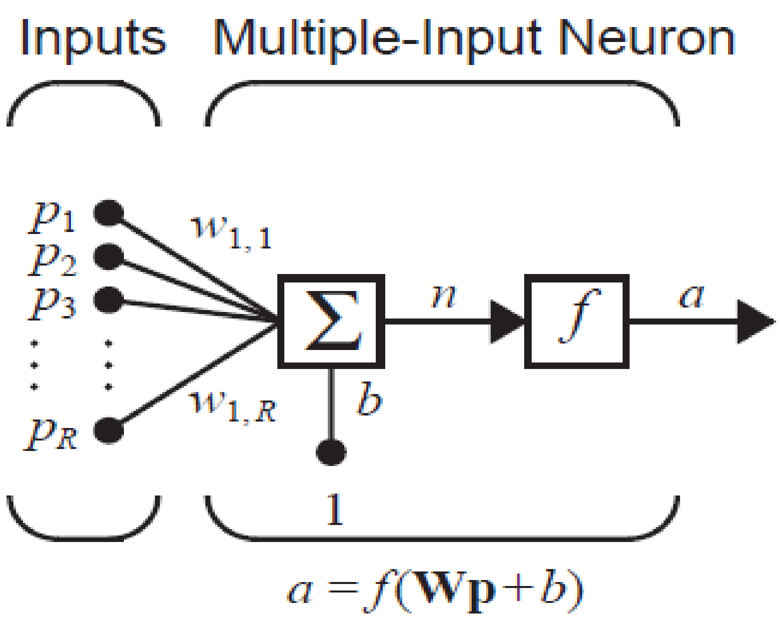

Figure 4: Multiple input neuron

From figure 4 it is seen that the neuron has multiple inputs. These inputs will represent the different currents generated by the various faults at different distances. The weighs are random numbers which will be automatically adjusted to arrive at the output of the neural network. The automatic adjustment is call training. The size of the weights depends on the size of the inputs. For instance, if there are R inputs, then the weights will be a *1xR* matrix (single row matrix) and b is the bias.

From figure 4:

$$n = w_{1,1}p_1 + w_{1,2}p_2 + \cdots + w_{1,R}p_R + b \tag{1}$$

In simplified form

$$n = \boldsymbol{WP} + b \tag{2}$$

The output can therefore be written as:

$$a = f(\boldsymbol{WP} + b) \tag{3}$$

Where f is an activation function that will be presented in subsequent sections.

Layer of Neurons

A single-layer network of S neurons is shown in Figure 5. Note that each of the inputs is connected to each of the neurons and that the weight matrix now has S rows.

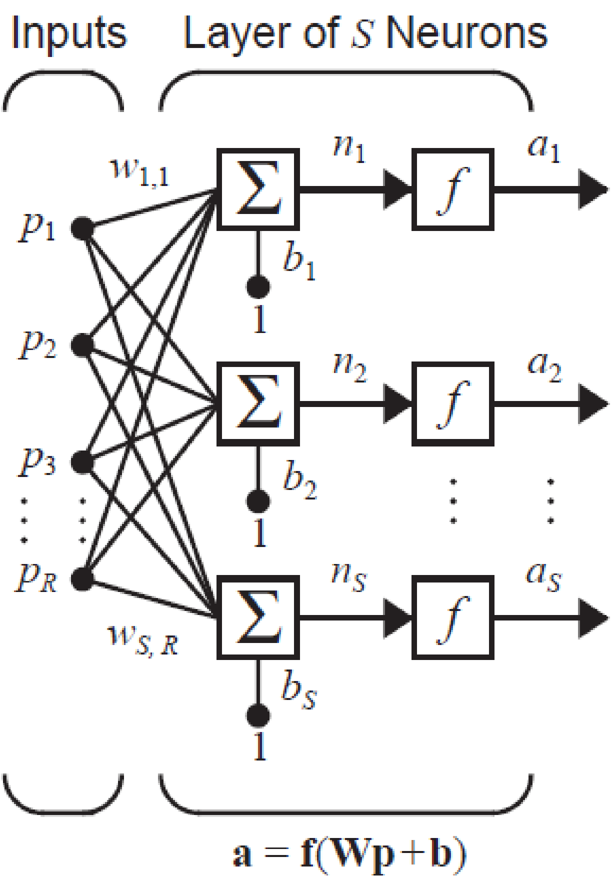

Figure 5: Layer of S Neurons

The layer includes the weight matrix, the summers, the bias vector, the activation function boxes and the output vector. Each element of the input vector **p** is connected to each neuron through the weight matrix **W**. Each neuron has a bias b_i, a summer, an activation function f and an output a_i. Taken together, the outputs form the output vector is **a** The input vector elements enter the network through the weight matrix **W**

$$\mathbf{W} = \begin{bmatrix} w_{1,1} & w_{1,2} & \cdots & w_{1,R} \\ w_{2,1} & w_{2,2} & \cdots & w_{2,R} \\ \vdots & \vdots & & \vdots \\ w_{S,1} & w_{S,2} & \cdots & w_{S,R} \end{bmatrix} . \qquad (4)$$

The row indices of the elements of matrix **W** indicate the destination neuron associated with that weight, while the column indices indicate the source of the input for that weight. Thus the indices in $w_{3,2}$ say that this weight represents the connection to the third neuron from the second source.

Multiple layers of Neurons

The function approximation neural network that will be designed in this research work is multiple layer neural network with back propagation. Figure 6 presents the design of the neural network for this research work.

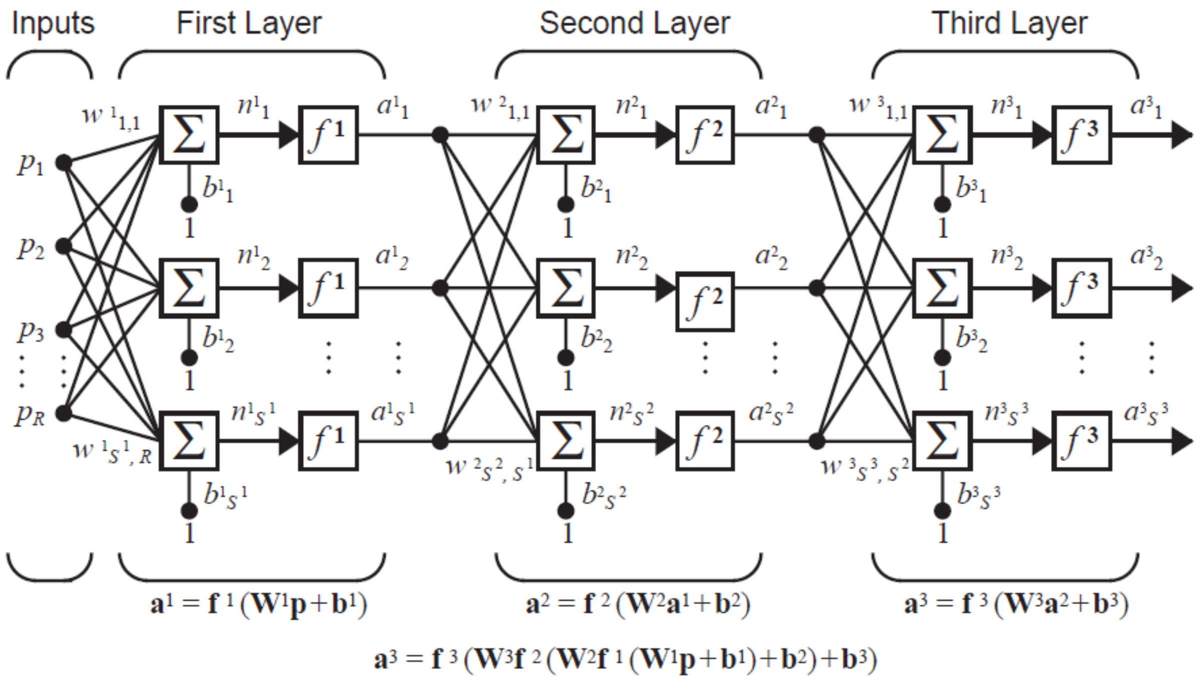

Figure 6: Artificial Neural Network design

The above ANN is seen to possess three layers, three activation functions (f^1, f^2, f^3) and inputs to the neural network as the network moves from one stage to the other. The neural network is first of all trained using a training set. Training entails updating the weight and bias values using the training set. After training, the neural network will have an idea of what the output would look like. If there is a difference between the target output and the output from the neural network, the error (difference) is fed back into the network and the weights and bias adjusted to reduce the error. This process continues in a loop until the error becomes zero. This process is called back propagation.

Training sets and Training

In this research work, fault current information / data will form the training sets and these data will be generated programmatically by simulating each of the faults and placing these faults at specific distance. The distance will be increased at an interval of 10m for each of the fault simulated. Table 3 presents the manner the training set will be generated.

Table 3: Training sets

S/N	Fault	Distance (km)	Current Data width
1	Balanced Fault	1x10^{-2}:1x10^{-2}:153	15, 300
2	Line to line	1x10^{-2}:1x10^{-2}:153	15, 300
3	Single line to ground	1x10^{-2}:1x10^{-2}:153	15, 300
4	Double line to ground	1x10^{-2}:1x10^{-2}:153	15, 300

After generating the training sets, the next step is to training the neural network. The neural network in this research work will be trained using the steepest decent rule. Consider equation (5)

$$\boldsymbol{W}^m(k+1) = \boldsymbol{W}^m(k) - \alpha \boldsymbol{s}^m(\boldsymbol{a}^{m-1})^T \quad (5)$$

$$\boldsymbol{b}^m(k+1) = \boldsymbol{b}^m(k) - \alpha \boldsymbol{s}^m \quad (6)$$

Where:

W is the weights as described in equation (4)

b is the bias

k represents the iteration number

m represents the neural network layer

α is a constant called learning rate (selected between 1 and 4)

s is given as:

$$\mathbf{s}^m = \frac{\partial \hat{F}}{\partial \mathbf{n}^m} = \left(\frac{\partial \mathbf{n}^{m+1}}{\partial \mathbf{n}^m}\right)^T \frac{\partial \hat{F}}{\partial \mathbf{n}^{m+1}} = \dot{\mathbf{F}}^m(\mathbf{n}^m)(\mathbf{W}^{m+1})^T \frac{\partial \hat{F}}{\partial \mathbf{n}^{m+1}} \quad (7)$$

and

$$\dot{\mathbf{F}}^m(\mathbf{n}^m) = \begin{bmatrix} \dot{f}^m(n_1^m) & 0 & \dots & 0 \\ 0 & \dot{f}^m(n_2^m) & \dots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \dots & \dot{f}^m(n_{S^m}^m) \end{bmatrix}.$$

a is output from the neural network

Activation function

In this research work, the activation function for each of the three layers of the designed neural network will be the hyperbolic tangent sigmoid function this function is suitable for designs that does not require a 1 or 0 as its solution. However, the ANN that will identify the type of fault would employ the same hyperbolic tangent sigmoid function at two of its layers while the outer layer will employ the hard limit function as this function is suitable for designs that require either a zero or a one. For instance, presence of a double line to ground fault will output a 1 and absence of the fault will output a 0. The hyperbolic tangent sigmoid function is presented in equation (8) while the hard limit function is presented in equation (9).

Hyperbolic tangent sigmoid function

$$a = \frac{e^n - e^n}{e^n + e^n} \tag{8}$$

a is the output and n is the input (see figure 4)

Hard limit function

$$\begin{aligned} a &= 0 \quad n < 0 \\ a &= 1 \quad n \geq 0 \end{aligned} \tag{9}$$

Fault identification and localization

As the fault changes and as the distance of the fault changes, the fault current also changes. This change simulates a function, thus a neural network with function approximation capabilities is designed. After training the neural network, the fault current data is fed into the network. The network does a lot of computation on the data using the equations provided. The final output, from the last tangent sigmoid function is used to approximate the current data for that particular fault and its location.

Simulation

Simulation will be carried out using the codes appended in appendix I. A MATLAB screen shot of the designed ANN is illustrated in the figure 3.7.

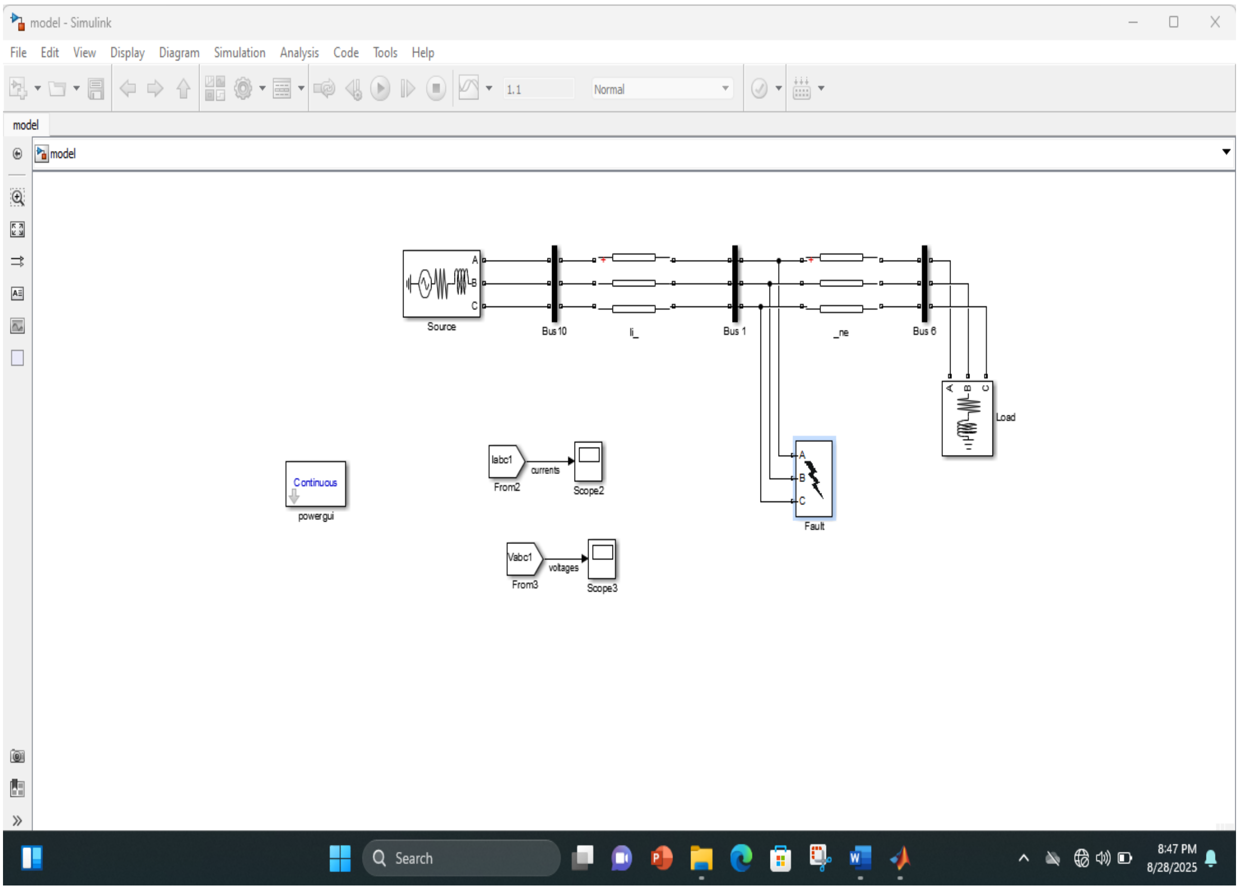

Figure 3.7 MATLAB design of the ANN network

Also, the algorithm used in the simulation is hereby presented:

i. Generate the training sets

ii. Train the ANN using equations (3.5) and (3.6)

iii. Select fault current for an arbitrary fault type and distance

iv. Use the fault current generated to test the neural network using the weight calculated after training.

RESULTS AND DISCUSSIONS

Results

For proper evaluation of the designed system, known faults and their distances will be simulated and the results presented. Next, fault currents for the arbitrarily selected faults and their location will be fed to the already trained ANN to see if it can identify such faults and their location.

The scenarios for proper evaluation are presented.

i. No fault (base case)
ii. Single line to ground fault at phase A
iii. Single line to ground fault at phase B
iv. Single line to ground fault at phase C
v. Double line to ground fault at phase AB
vi. Double line to ground fault at phase AC
vii. Double line to ground fault at phase BC
viii. Balanced fault
ix. ANN result

All distances from the scenarios present above will be 20 km (arbitrarily selected) from the sending end.

The simulation parameters are presented in table 4

Table 4: Simulation parameters

S/N	Fault time (s)	Fault duration (s)	Simulation time (s)
1	0.3667	0.3667	1.1

No fault (base case)

Simulation result for base case is presented in figure 7

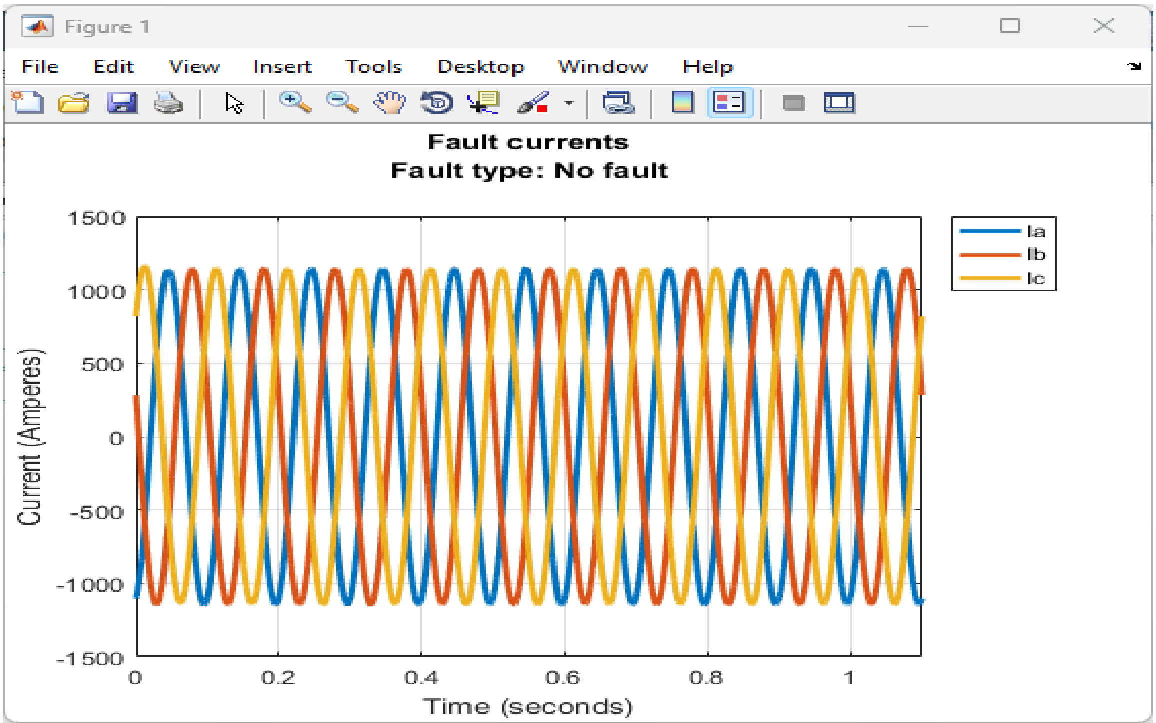

Figure 7: Fault currents for base case

Single line to ground fault phase A

Simulation result for single line to ground fault phase A is presented in figure 8

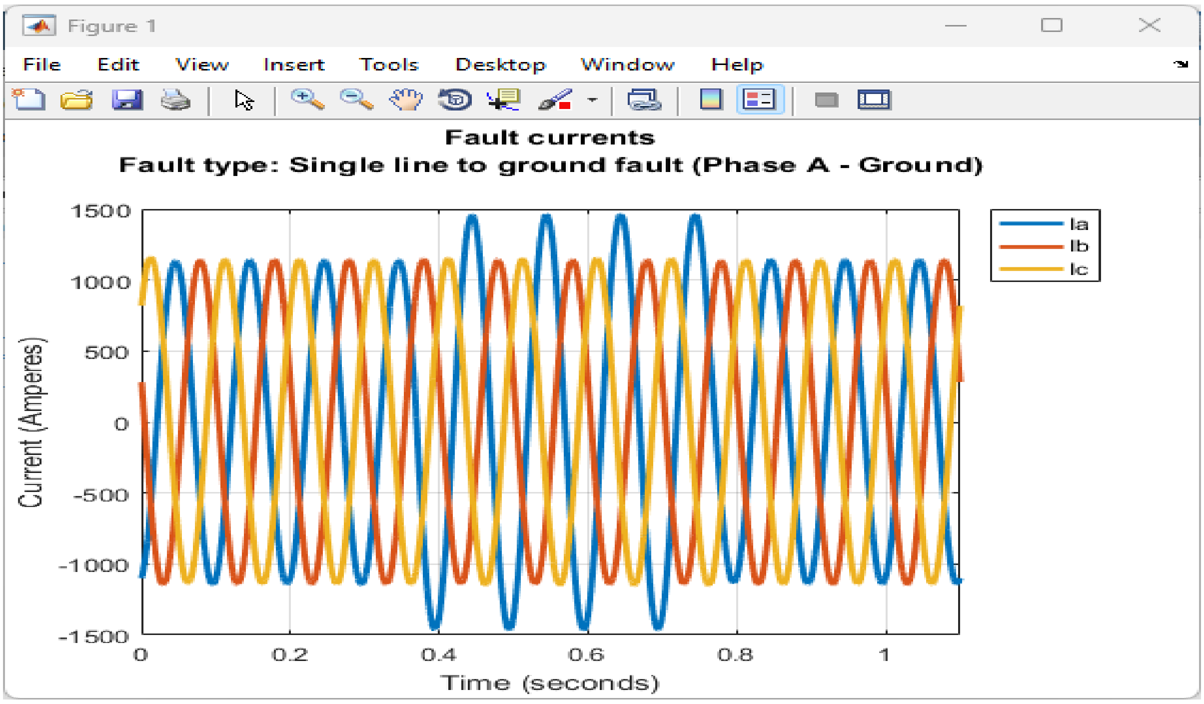

Figure 8: Fault current for single line to ground (phase A)

Single line to ground fault phase B

Simulation result for single line to ground fault phase B is presented in figure 9

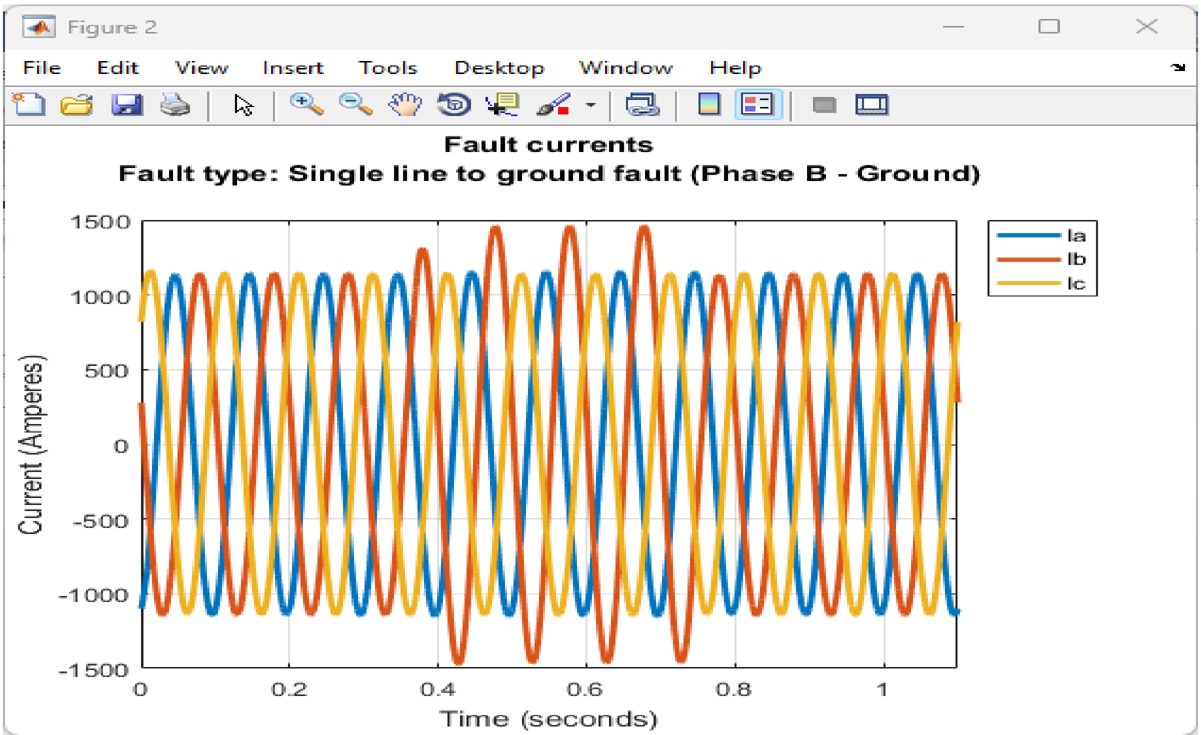

Figure 9: Fault current for single line to ground (phase B)

Single line to ground fault phase C

Simulation result for single line to ground fault phase C is presented in figure 10

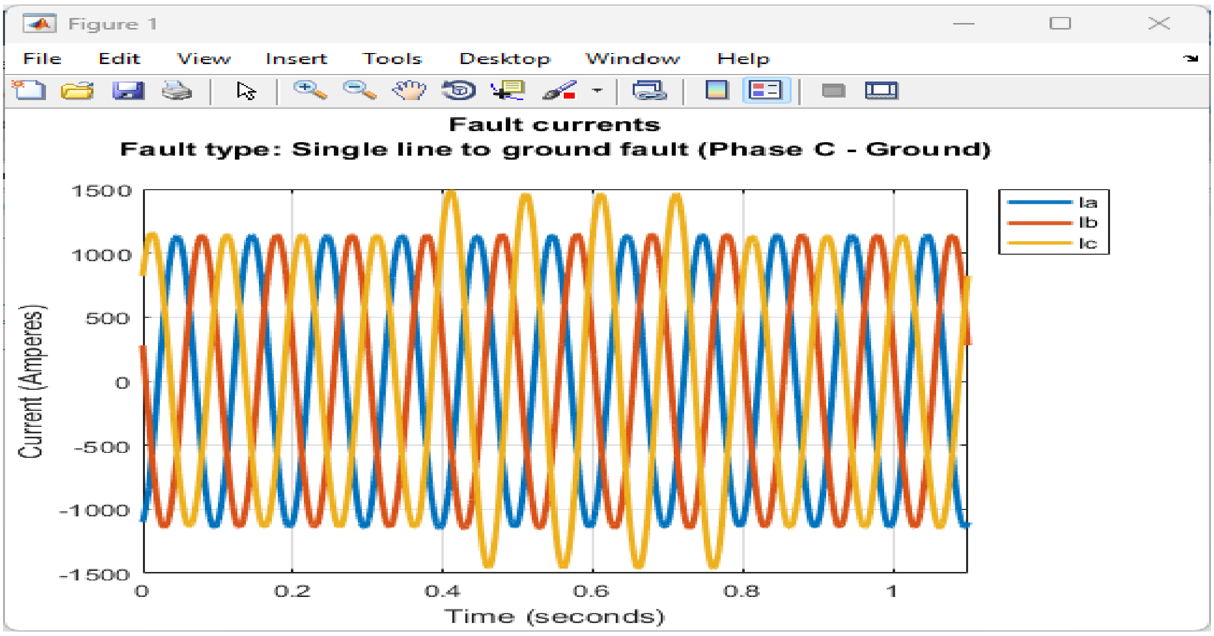

Figure 10: Fault current for single line to ground (phase C)

Double line to ground fault phase AB

Simulation result for double line to ground fault phase AB is presented in figure 11

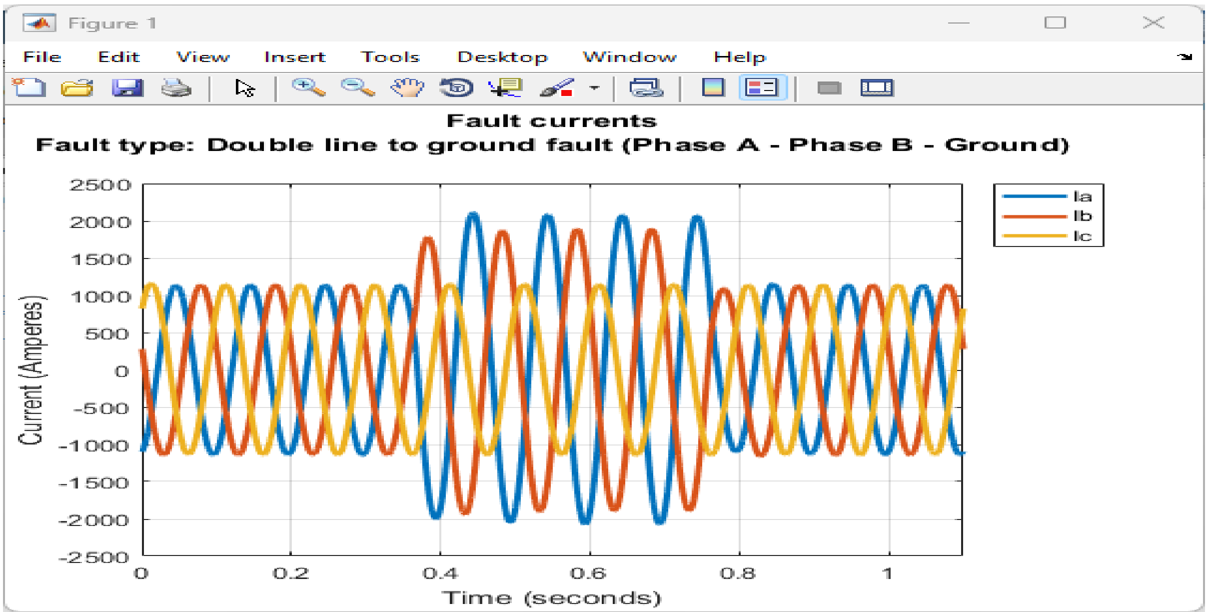

Figure 11: Fault current for double line to ground (phase AB)

Double line to ground fault phase AC

Simulation result for double line to ground fault phase AC is presented in figure 12

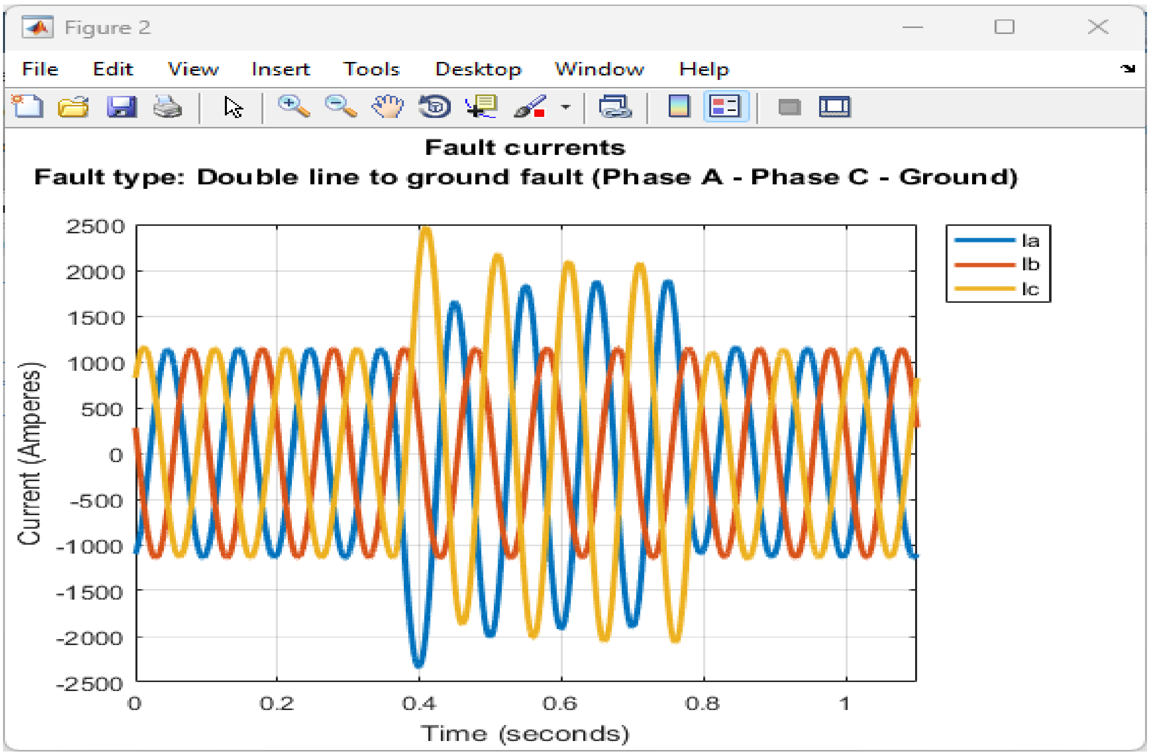

Figure 12: Fault current for double line to ground (phase AC)

Double line to ground fault phase BC

Simulation result for double line to ground fault phase BC is presented in figure 13

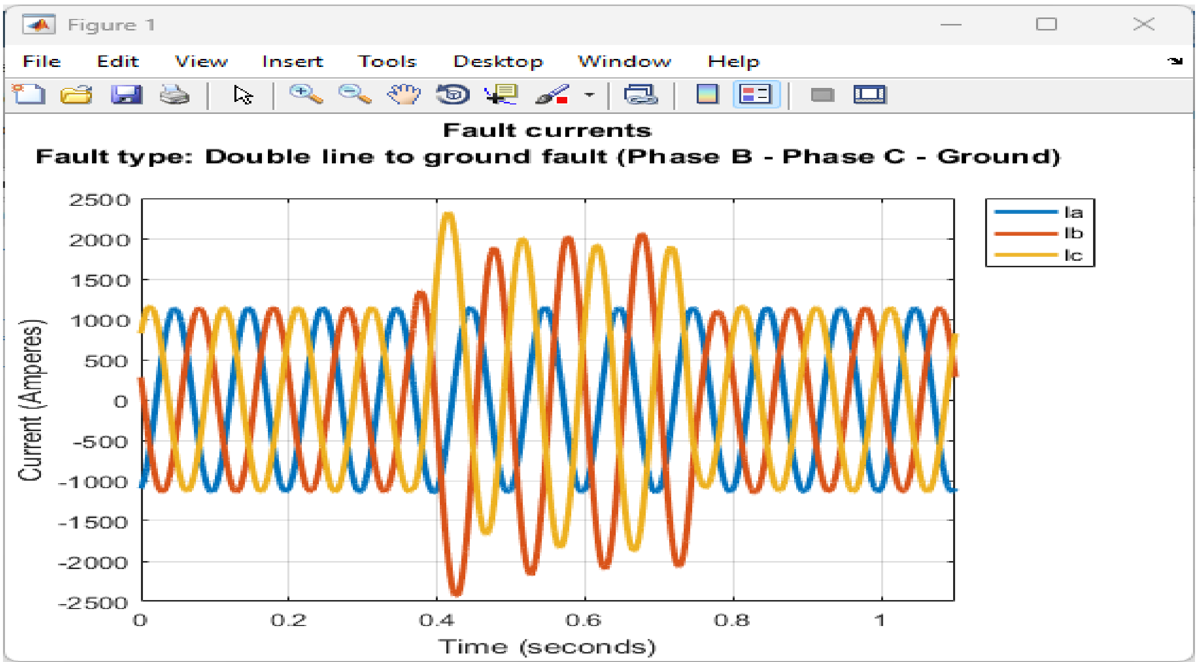

Figure 13: Fault current for double line to ground (phase BC)

Balanced fault (ABC)

Simulation result for balanced fault phase ABC is presented in figure 14

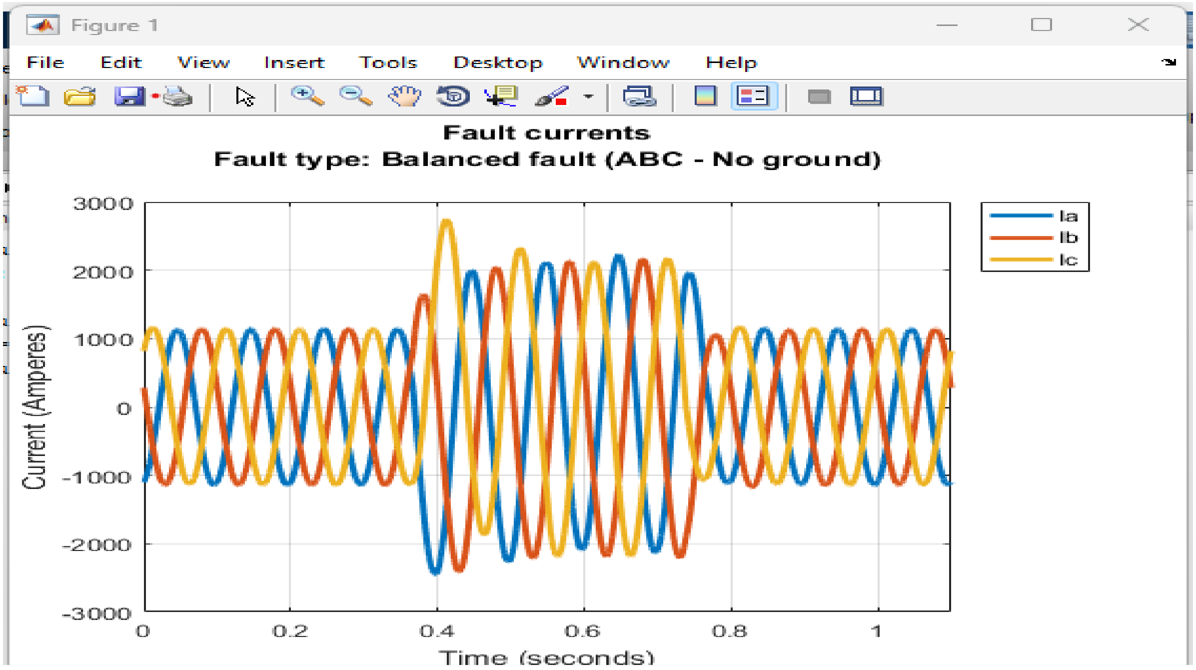

Figure 14: Balanced Fault current (phase ABC)

ANN result

The currents generated from the fault results presented is fed to the ANN and the result presented in table 4.

Table 5: ANN result

S/N	Fault type	ANN output (fault type)	ANN output (distance) (km)	Number of iterations (epochs)
1	No fault	1	-	5, 000
2	Single line to ground (phase A)	1	19.91	10, 000
3	Single line to ground (phase B)	1	19.93	10, 000
4	Single line to ground (phase C)	1	19.90	10, 000
5	Double line to ground (AB)	1	19.94	10, 000
6	Double line to ground (AC)	1	19.94	10, 000
7	Double line to ground (BC)	1	19.95	10, 000
8	Balanced fault	1	19.91	10, 000

Discussions

Discussions for the results is presented in the same manner the scenarios were presented.

From figure 7, simulation was carried out without any faults and the graph shows the normal sinusoidal nature of the alternating currents of all three phases. No spike in current is seen. The ANN was able to identify the fact that there was no fault by out putting a 1 from its hard limit function of its outer layer. This is seen in table 5.

From figure 8 a single line to ground fault is simulated at phase A. A huge spike in the current for phase A is seen indicating a fault at that phase. The ANN properly identified the fault and predicted the distance to be 19.91 km instead of 20 km (see table 5) with a percentage error of 0.45 %

From figure 9 a single line to ground fault is simulated at phase B. A huge spike in the current for phase B is seen indicating a fault at that phase. The ANN properly identified the fault and predicted the distance to be 19.93 km instead of 20 km (see table 5) with a percentage error of 0.35 %

From figure 10 a single line to ground fault is simulated at phase C. A huge spike in the current for phase C is seen indicating a fault at that phase. The ANN properly identified the fault and predicted the distance to be 19.90 km instead of 20 km (see table 5) with a percentage error of 0.5 %

From figure 11 a double line to ground fault is simulated at phase AB. A huge spike in the current for phase AB is seen indicating faults at those phases. The ANN properly identified the fault and predicted the distance to be 19.94 km instead of 20 km (see table 5) with a percentage error of 0.3 %

From figure 12 a double line to ground fault is simulated at phase AC. A huge spike in the current for phase AC is seen indicating faults at those phases. The ANN properly identified the fault and predicted the distance to be 19.94 km instead of 20 km (see table 5) with a percentage error of 0.3 %

From figure 13 a double line to ground fault is simulated at phase BC. A huge spike in the current for phase BC is seen indicating faults at those phases. The ANN properly identified the fault and predicted the distance to be 19.95 km instead of 20 km (see table 5) with a percentage error of 0.25 %

From figure 14 a balanced fault is simulated at phase ABC. A huge spike in the current for all phases, ABC is seen indicating faults at those phases. The ANN properly identified the fault and predicted the distance to be 19.91 km instead of 20 km (see table 5) with a percentage error of 0.45 %

In conclusion, the ANN correctly predicted the faults and approximately predicted the correct distances with a worse percentage error of 0.45 % or accuracy of 99.55 % (100 -45). This percentage error is minute but becomes a problem when the line gets very long, in the range of thousands of kilometers.

CONCLUSION AND RECOMMENDATION

Conclusion

This research work modelled the NIPP 218 km transmission line using MATLAB SIMULINKS

Simpower systems software. After modelling the line an ANN is designed to identify the various power system faults and predict their location. The ANN had three layers of neuron and is designed with back propagation capabilities. The following strong points can be concluded from this research work.

i. The ANN predicted all fault types correctly
ii. The ANN predicted, approximately, the fault location
iii. Accuracy of fault location is 99.55 %

This research has shown that the ANN is capable of locating faults and also identifying the type of faults if properly trained.

Recommendation

Fault location accuracy in this research work is at 99.55 %. This seems to be good but becomes a problem when the line length is large. This accuracy is connected to the number of layers employed when designing the ANN. It is therefore recommended that the number of layers in the ANN designed be increased to 5 and above instead of the 3 layers employed in this research. However, increasing the number of layers impacts the speed of processing the ANN and for relatively cheap computers this takes hours to complete.

Also, advanced designs like SMS capability, Internet of things (Iot) capabilities is also recommended in future work. This makes it possible for an authorized employee to be able to track these faults using mobile devices and not necessarily on a workstation computer.

Acknowledgement

I'm deeply grateful to my supervisor, Prof. Ali Shalangwa, for his exceptional supervision, insightful feedback, and continuous encouragement. The support and resources from Department of Pure and Applied Physics were instrumental in completing this thesis. I'm also thankful to my family and friends for their love, patience, and understanding throughout this journey.

CONFLICT OF INTEREST

There is no conflict of interest between the Authors.

REFERENCES

Abhishek, K., Kumar, A., Ranjan, R., & Kumar, S. (2012). A Rainfall Prediction Model using Artificial Neural Network. *IEEE Control and System Graduate Research Colloquium (ICSGRC 2012)*, 82–87.

Adepoju, G. A., Tijani, M. A., Sanusi, M. A., & Olatunji, D. O. (2013). Three-phase fault currents evaluation for Nigerian 28-bus 330 kV transmission system. *International Journal of Engineering Research and Applications*, *3*(2), 126–132.

Atuchukwu, J. A., Ochogwu, S. O., Okonkwo, I. I., & Ogboh, V. C. (2023). Detection of Faults on Nigeria 330kv Power Transmission Lines Using Artificial Intelligence (AI). *Iconic Research and Engineering Journals (IRE-Journals)*, *6*(7), 390–404.

Babu, K. V., Tripathy, M., & Singh, A. K. (2011). Recent techniques used in transmission line protection: a review. *International Journal of Engineering, Science and Technology*, *3*(3), 1–8.

Buckminister, R. (2016). *Transmission power lines across Nigeria*. Geni Global Energy.

Dash, Y., & Dubey, S. K. (2012). Quality prediction in object-oriented system by using ANN: a brief survey. *International Journal of Advanced Research in Computer Science and Software Engineering*, *2*(2), 23–34.

Mrabet, Z. El, Sugunaraj, N., Ranganathan, P., & Abhyankar, S. (2022). Random Forest Regressor-Based Approach for Detecting Fault Location and Duration in Power Systems. *Sensors*, *22*(458), 1–19. https://doi.org/10.3390/s22020458

Panteli, M., & Mancarella, P. (2015). Influence of extreme weather and climate change on the resilience of power systems: Impacts and possible mitigation strategies. *Electric Power Systems Research*, *127*, 259–270.

Shalangwa, D. A., Shinggu, D. Y., & Jonathan, T. (2009). An Analysis of Electric Power Faults in Mubi Undertaking Station, Adamawa State, Nigeria. *The Pacific Journal of Science and Technology*, *10*(number 2), 508–513.

Xiao, R., Xiang, Y., Wang, L., & Xie, K. (2018). Power system reliability evaluation incorporating dynamic thermal rating and network topology optimization. *IEEE Transactions on Power Systems*, *33*(6), 6000–6012.

Yadav, A., & Dash, Y. (2014). An Overview of Transmission Line Protection by Artificial Neural Network: Fault Detection, Fault Classification, Fault Location, and Fault Direction Discrimination. *Hindawi Publishing Corporation Advances in Artificial Neural Systems*, *2014*, 1–20.

Zhang, F., Liu, Q., Liu, Y., Tong, N., Chen, S., & Zhang, C. (2020). Novel Fault Location Method for Power Systems Based on Attention Mechanism and Double Structure GRU Neural Network. *IEEE Power & Energy Society Section-IEEE Access*, *8*(1), 75237–75248. https://doi.org/10.1109/ACCESS.2020.2988909

Zhou, Y., & Cao, R. (2018). The artificial neural network prediction algorithm research of rail-gun current and armature speed based on B-dot probes array. *Measurement*, *133*(2019), 47–55.

Appendix I

Codes for simulation

```
clc
clear

tic
freq       = 10;%     Hz
mdl        = 'model';
SAVE       = 1;
linewidth   = 2.0;

%========================================
% LINE
%========================================
flen        = 80;
line_len    = 218;
%========================================

%========================================
% SOURCE
%========================================
svoltage    = 20e3;
sbvoltage   = 20e3;
sflevel     = 1158e6;
%========================================

%========================================
% LOAD
%========================================
lap        = 283.4e6;  % Load active power
lip        = 159.4e6;  % Load inductive power
lnv        = 20e3;     % Load nominal voltage
%========================================
 ========================================
% SIMULATION STOP TIME
%========================================
sst = 1.1;
%========================================

%========================================
% FAULT TIME
%========================================
ft = [sst/3 sst*(2/3)];
%========================================

%========================================
% FAULT TYPE
%========================================
faultType = 'nF';
%========================================

ftype = SetFaultType(mdl,faultType);

SetAllParameters(mdl, line_len, flen, svoltage, sbvoltage,...
             sflevel, lap, lip, lnv, ft, freq, sst)

open_system(mdl, 'loadonly');

simr = sim(mdl);

rI = get(simr,'scopeI');
xI = rI.time;
yI = rI.signals.values;

rV = get(simr,'scopeV');
xV = rV.time;
yV = rV.signals.values;

close_system(mdl, SAVE);

figure('color','w')
plot(xI,yI,'linewidth',linewidth)
legend('Ia','Ib','Ic','location','Northeastoutside')
title({['Fault ' rI.signals.title ], ['Fault type: ' ftype], ' '})
xlabel('Time (seconds)')
ylabel('Current (Amperes)')
set(gca,'xlim',[0 sst])
grid on
 % figure('color','w')
% plot(xV, yV,'linewidth',linewidth);
% legend('Va','Vb','Vc','location','Northeastoutside')
% title({['Fault ' rV.signals.title], ['Fault type: ' ftype], ' '})
% xlabel('Time (seconds)')
% ylabel('Voltage')
% set(gca,'xlim',[0 sst])
% grid on
 fprintf('\n')
disp('simulation completed ...')
disp('=======================')
fprintf('simulation time = %5.2f seconds\n\n', toc)
```