iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 **Issue:** IV **Month of publication:** April 2023

DOI: https://doi.org/10.22214/ijraset.2023.50867

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue IV Apr 2023- Available at www.ijraset.com

Fault Detection and Classification using Machine Learning

Dr. B. Suresh Kumar[1], V. Abhinav Varma[2], S. Devika Rani[3], A. Nishanth[4]
Chaitanya Bharathi Institute of Technology

Abstract: ***Machine learning plays a crucial role in predicting and clas- sifying faults in electrical power systems. The complexity and dynamic nature of these systems make them vulnerable to disturbances and elec- trical faults. Detecting faults in the circuit can aid in maintaining the system by preventing potential damage from occurring. It is particularly important to be able to locate faults in transmission lines to minimize power and revenue losses. In this study, MATLAB software will be uti- lized to simulate and locate faults in transmission lines. A transmission line model will be designed, and a fault toolbox will be employed to cre- ate various faults. These faults will be saved and used to train machine learning models to identify the best algorithm with high precision and accuracy.***

Keywords: Machine learning· Fault classification· MATLAB· Power sys-tems.

I. INTRODUCTION

The task of fault detection and classification (FDC) holds significant importance in a range of industries, including transportation, energy, and manufacturing. The primary objective of FDC is to recognize any abnormal conditions present in a given system or process and categorize them into distinct fault categories. This process can aid in the prevention of system failures, reduction of down- time, and improvement of overall productivity. With the emergence of machine learning (ML), FDC has become more efficient as it can learn patterns and re- lationships automatically from large amounts of data. ML algorithms can be trained on data collected from MATLAB Simulink models to accurately detect and classify faults. Additionally, ML-based FDC has the capability to adapt to changing conditions and learn from past experiences, making it more effective than conventional rule-based methods.

II. RELATED WORK

Goh et al.[1] emphasize the importance of fault location in electric power systems, which can be caused by various factors such as natural events, human error, and equipment malfunctions. Effective fault location is critical to ensure continuous system operation, minimize revenue loss, and prevent harm to the environment and equipment. The authors suggest that a traveling-based method is superior for detecting faults in transmission lines as it is not affected by load variation, series capacitor banks, or high grounding resistance. However, this approach may require more expensive devices than other methods, even though it is more efficient.

In order for this study to be successful, it is crucial to develop a reliable model that can simulate a three-phase power system network. Without an accurate simulation environment, any algorithm tested will not be considered effective. Tharani et al. [2] demonstrated the effectiveness of a MATLAB-based simulation model for three-phase power system networks. This study highlights the conve- nience and reliability of MATLAB software for analyzing and simulating faults in transmission lines within a three-phase power system network.

Karekar, Thakur, and Manju [3] conducted a study on analyzing and de- tecting faults in transmission lines through a novel approach. The study in- volved testing L-G fault, LL-G fault, and LLL-G fault using MATLAB Sim- ulation with the Simulink and sim-power toolbox. The researchers found that the use of MATLAB software was efficient for this purpose. The system used in the study consisted of 33Kv generators, three-phase simulators, and distributed parameter lines. The three-phase simulators were utilized to simulate faults at the mid-position, while the distributed parameter lines represented faults on the transmission line.

Approximately 80 percent of transmission line disturbances are caused by undetected faults. As faults can arise due to various environmental factors, such as natural disasters, it is essential to implement techniques that ensure a con- tinuous power supply within the system. A continuous power supply is crucial because unaccounted faults can lead to the gradual degradation of power quality across transmission lines. Therefore, it is imperative to have a fast and accurate fault detection process in place to ensure stability, performance, and efficiency[4]. Since transmission networks have multiple lines, fault detection using fault de- tectors can be prohibitively expensive.

©IJRASET: All Rights are Reserved

Therefore, it is important to explore alter- native methods such as impedance-based techniques and traveling-wave-based fault location and integrate them into algorithms and online simulations to find an efficient means of detecting faults.

With the rapid advancements and constant changes in technology, the de- mand for power continues to rise steadily. As a result, it is imperative to en- sure the safe, efficient, and maximum delivery of power. Rajput and Chaturvedi conducted a MATLAB simulation which demonstrated that the software is a proficient tool for executing algorithms to detect faults in transmission lines [5]. Out of the various simulations and techniques tested, online simulations proved to be a viable option, as they are cost-effective and easily modifiable.

Learning to Optimize" is a research paper presented at the International Conference on Machine Learning (ICML) in 2018[6]. The paper proposes a new approach to optimization that learns to optimize using deep neural networks. The traditional optimization methods used in machine learning involve finding the best values for a set of parameters in a given function. However, these methods can be slow and inefficient for complex functions, and they often require man- ual tuning. Overall, "Learning to Optimize" presents an innovative approach to optimization that has the potential to improve the efficiency and effectiveness of machine learning algorithms. The paper provides valuable insights into the application of deep learning to optimization problems and has implications for the development of more efficient and effective machine learning systems.

III. STATEMENT OF THE PROBLEM

The objective of this study is to comprehend the distinct behaviors exhibited by different fault types in a three-phase transmission line and to detect and classify faults with machine learning. The researchers aim to simulate the po- tential outcomes of various types of faults, including line-to-ground, line-to-line, double-line-to-ground, and triple-line-to-ground. By comprehending the patterns and behaviors of these fault types, it will be easier to identify and resolve them in real-world scenarios. Initiating a transmission line simulation in Matlab and Simulink can be chal- lenging, particularly for beginners. The required control blocks and libraries may not be immediately clear, and the correct settings and parameters require some explanation. This documentation may not fully convey the functions of certain block parameters, particularly in real-world equivalents. Consequently, this study provides an overview of how to create a transmission line system and generatefaults within it to understand faults and their behaviors and to be able to collect data from MATLAB and detect and classify faults with various machine learning algorithms by using data collected.

IV. DESCRIPTION OF THE MATLAB SYSTEM

In order to simulate faults in transmission lines effectively, a MATLAB Simulink transmission line circuit was created. This model was based on previous models that had been studied. The circuit consists of subsystems that utilize 5 cascaded blocks of the Three-Phase Pi Section Lines Subsystem with connection ports at each end. On the left side of the circuit, a Three-Phase source acts as a voltage source, while a load is created by adding another three-phase source with an offset of -30 degrees on the opposite end. To measure voltage and current at each line, a Three-Phase V-I Measurement is included. Additionally, a Three- Phase Fault is added to the system to simulate faults and compare stable systems with those that have faults. To ensure accurate results and decreased simulation time, a graphical user interface, a Powergui block with a sample time of 3e-5, was added to the circuit. The Simulink model can be viewed in Figure 1.

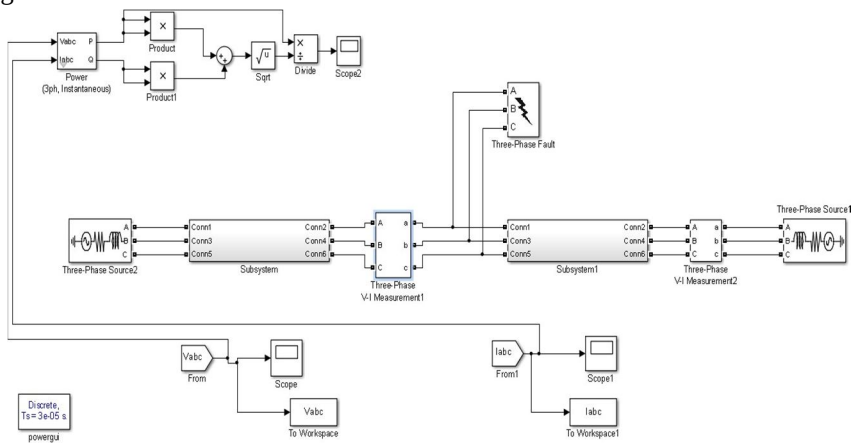


Fig. 1. Simulink block model of the Transmission Line System featuring a connected fault block.

V. METHODOLOGY

A. Methodology for creating MATLAB Simulink Model

1) Model Preparation

Created a Simulink model as described in figure 1, and set parameters as described above. we began by constructing a transmission line block system in Simulink. The specific blocks used were the following: 1. Three-Phase Pi Section Line 2. Connec- tion Port 3. Three-Phase Fault 4. Three-Phase V-I Measurement 5. Three-Phase Source 6. Powergui 7. Scope 8. From 9. To Workspace

We initiated the creation of a Transmission Line system by generating a Three-Phase Pi Line subsystem. This involved importing five Three-Phase Pi Line blocks and six connection port blocks into the workspace. In Matlab R2013, these blocks should be found in Simscape > Electrical > Specialized Power Systems > Fundamental Blocks > Elements.

As part of the theoretical assumptions of an ideal transmission line system,we did not modify the default settings of the Three-Phase Pi Section Line blocks. These blocks were then connected in the manner depicted below. The blocks were then connected in the manner shown below.

After setting up the Three-Phase Pi Section Line blocks as described above, we selected all the blocks and used the Ctrl+G keyboard shortcut to create a subsystem block.

The Three-Phase Source block will act as the voltage source of the transmis-sion line circuit.

The Voltage (Vabc) and Current (Iabc) measurements will be taken on this block.

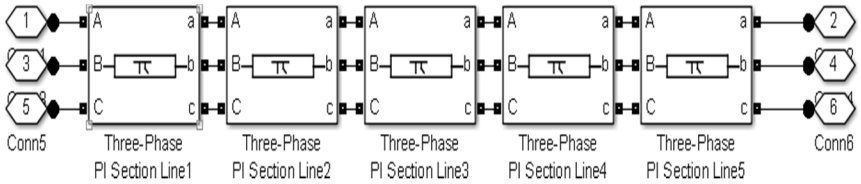


Fig. 2. The Three-Phase Pi Section Line subsystem block.

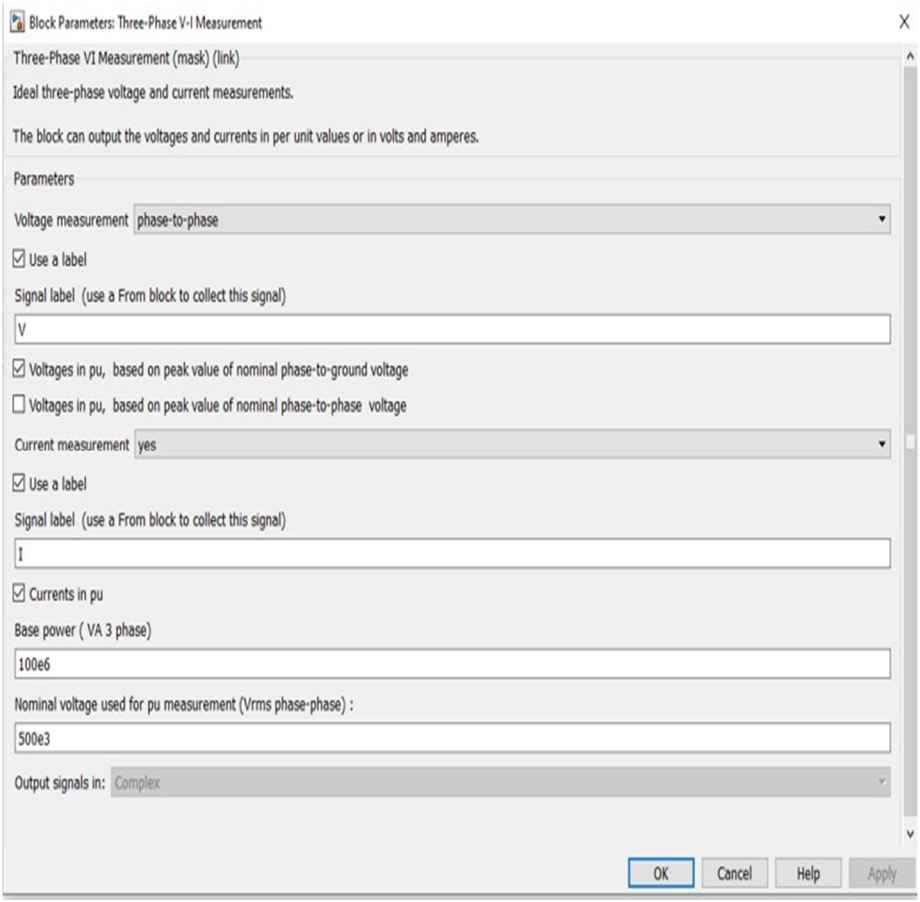


Fig. 3. Parameters on first Three-Phase V-I Measurement block.

The second Three-Phase V-I Measurement block did not perform any mea- surements. Instead, it functioned as a relay bus for the circuit.In this block select voltage and current measurements as 'no'.
To decrease simulation computation time, the Powergui block was set to Discrete-time. The sample time was set to 3e-5, although this value is arbitrary. Lower sample times will generate more accurate results at the expense of longer computation times. Prior to running any simulations, no faults were enabled in the Fault block.

2) Signal Selection
Selected the signal's voltage and current as parameters for data collection and evaluation from respective scopes.
Simulation: simulated Simulink model with the selected signals to generate data. The simulation can be run for a specific time period, here we chose it around 0.5 or until a specific event occurs, such as a fault or a change in the system operation.

3) Data collection and Preprocessing
To fetch the data for an individualfault: choose runtime (0.5s) to observe fault graph > click on save > click on Run > select scope> history > click on save data to workspace> variable name:I_xyz (for current scope), V_xyz(for voltage scope) > format: Structure with time > click on apply and okay > select To workspace > save format: Time series > click on apply and okay > click on Matlab home tab open variables > To collect current scope data: click on I_xyz > signals > values(Ia, Ib, Ic),To collect Voltage scope data: click on V_xyz > signals > values(Va, Vb, Vc).
Here, select all values and copy them to the Excel sheet in the format: Ia, Ib, Ic, Va, Vb, Vc. Now, remove steady-state values from the data fetched by observing the scope graph (i.e, remove data before and after the fault occurred)because we need only fault region data for our machine learning analysis, herewe included data from 0.02s to 0.08s (because in this time range, the fault is observed in both I, V scopes), but for the NO-fault case, we have collected data in the entire 0.5s range. Repeat the same process for fetching data of all individual faults and add the data collected to the Excel sheet in the format specified.

- columns of collected data are in the format: G, C, B, A, Ia, Ib, Ic, Va, Vb,Vc.
- binary code for [G C B A] is given as follows:

  1- fault case, **0**- no fault case.
- 0 0 0 0 - No-Fault,
- 1 0 0 1 - LG fault (Between Phase A and Gnd),
- 1 0 1 0 - LG fault (Between Phase B and Gnd),
- 1 1 0 0 - LG fault (Between Phase C and Gnd),
- 0 0 1 1 - LL fault (Between Phase A and Phase B),
- 0 1 1 0 - LL fault (Between Phase B and Phase C),
- 0 1 0 1 - LL fault (Between Phase C and Phase A),
- 1 0 1 1 - LLG Fault (Between Phases A, B and ground),
- 1 1 1 0 - LLG Fault (Between Phases B, C and ground),
- 1 1 0 1 - LLG Fault (Between Phases C, A and ground),
- 0 1 1 1 - LLL Fault(Between all three phases),
- 1 1 1 1 - LLLG fault(Three phase symmetrical fault).

Here's a link to Data Set click here.

B. Methodology For Fault Detection And Classification Using Machine Learning In Power System Considered

1) Data Collection: Collected fault data from MATLAB workspace. This data includes signals relevant to the system's operation, such as voltages and currents only in the range where the fault occurred, and added 4 columns[G, C, B, A] to represent various faults in binary format.

2) Data Labeling: Labeled the extracted features with the corresponding fault type as shown above in the data collection and preprocessing section of Matlab methodologies.

3) Data Visualization: Data visualization plays a significant role in fault de- tection and classification using machine learning. It helps in identifying patterns and anomalies that may indicate faults, understanding the relationship between different variables and their impact on the system, and aiding in feature selec- tion for building accurate machine learning models. Here, the data visualization libraries used are Seaborn, and Matplotlib.

4) *Model Training:* Trained the selected machine learning model using the labeled data. This involves using a portion of the labeled data to train the model(Train data: 80 percent) and another portion to validate the model's per- formance.
5) *Model Testing:* Tested the trained model using additional data that was not used in the training phase(Test data: 20 percent). This is necessary to evaluate the model's performance on new, unseen data.
6) *Data Encoding:* Labeled the extracted features with the corresponding fault type. as shown in the figure 4.
7) *Model selection:* By performing all suitable machine learning models an optimal model was chosen for the task of fault detection and classification. Which possesses the necessary capabilities to effectively handle the unique features of the collected power system data. This model has been shown to achieve a higher degree of accuracy in accurately classifying different types of faults.

This link guides you through Machine Learning Code in Github: click here.

Line A Line B Line C	[0]
Line A Line B Line C to Ground Fault	[1]
Line A Line B to Ground Fault	[2]
Line A Line C to Ground Fault	[3]
Line A to Ground Fault	[4]
Line A to Line B Fault	[5]
Line A to Line C Fault	[6]
Line B Line C to Ground Fault	[7]
Line B to Ground Fault	[8]
Line B to Line C Fault	[9]
Line C to Ground Fault	[10]
NO Fault	[11]

Fig. 4. Label encoding

VI. RESULTS AND DISCUSSION

A. *MATLAB-fault block settings and scope results*

When no fault is selected the system is said to be in a steady state. Similarly, if any of the lines and ground are selected then it is line to ground fault. Now to illustrate the scope view let's select phase C and ground, block parameters, andscope graphs as shown in Figs 5 to 7. Similarly, simulated all fault types.

B. *Prediction Results From Different Machine Learning Algorithms*

Machine learning algorithms and their obtained accuracies are:
1) Logistic regression's model accuracy score: 43.18%
2) Decision Tree model accuracy score: 88.02%
3) Support Vector Machine's accuracy score: 76.17%
4) Random Forest's accuracy score: 89.06 %

After performing these algorithms we came to the conclusion that random forest and decision tree are working best for this type of problem with greater accuracy. Comparatively Random forest is giving more accuracy than Decision tree.

VII. ANALYSIS OF DATA

The simulations conducted previously involve a fault where one phase is grounded.The results show that the voltage of the grounded phase fluctuates, while the voltages of the other phases are only slightly affected. The current in the faulted

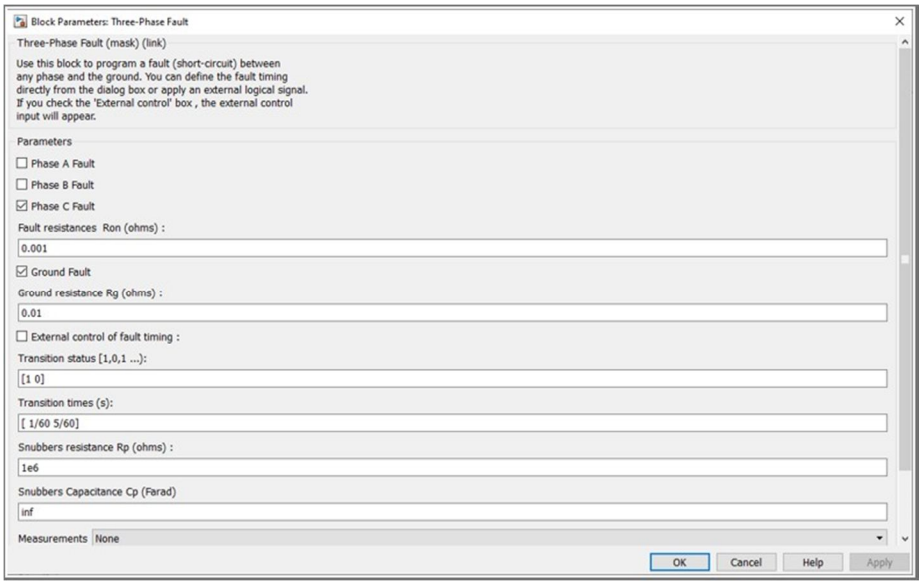


Fig. 5. Parameters on 3 phase fault block (ex: fault between line C and Ground.)

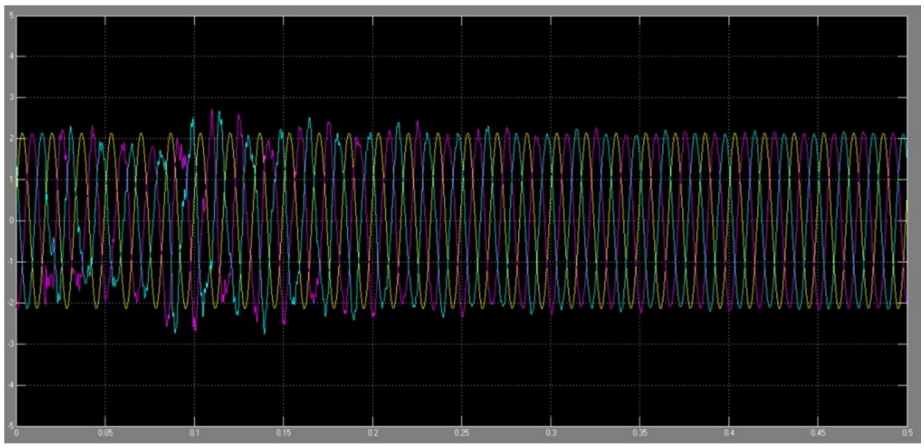


Fig. 6. voltage scope of fault between line C and Ground.

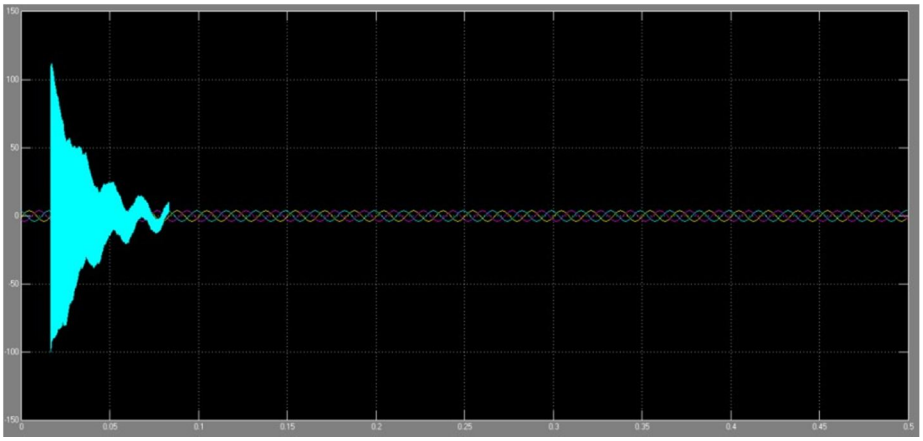


Fig. 7. current scope of fault between line C and Ground.)

Phase peaks abruptly at around 0.02 seconds, causing destabilization, whereas the currents in the other phases remain stable throughout the simulation. The fault can occur when A and B, B and C, or A and C are connected to the ground. The voltage experiences significant fluctuations during the fault, while the cur- rent in the faulted phase peaks at the beginning of the fault and continues to fluctuate. The current in the non-faulted phase remains stable. The results indi- cate that the fault occurred at approximately 0.02 seconds, causing the voltage to drop to 0 and the current in the faulted phase to peak. After approximately 1 second, both voltage and current stabilize. For line-to-line faults and beyond, the behavior is similar to that of multi-line-to-ground faults.

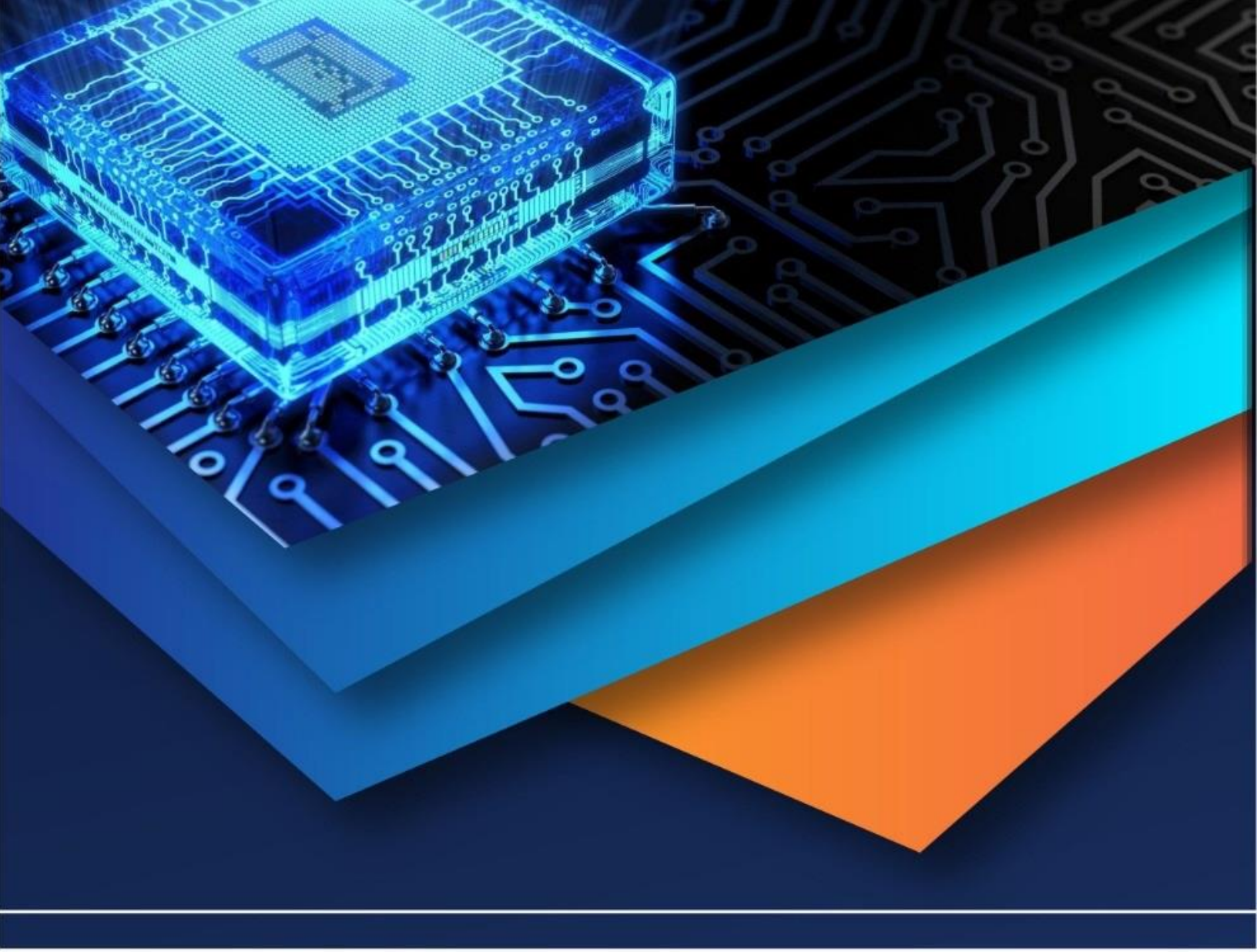
VIII. CONCLUSION

MATLAB Simulink can be a potent tool to simulate transmission line faults. Before creating our own simulation model, this study explored different trans- mission line fault detection techniques proposed in recent studies. They demon- strated how various algorithms can be used to assess faults. The study then explained how to create a Transmission Line simulation model in Simulink, in- cluding creating a cascaded Three-Phase Pi Line subsystem. The Three-Phase fault block was used for fault detection simulations and connected across the circuit to simulate different fault scenarios. These scenarios showed different outcomes, with voltage destabilization and lowering, and abrupt increases in current. The model was simulated in discrete time to reduce computation time. The data collected was then fed into all suitable machine-learning models and we chose a model with greater accuracy i.e Random forest. In summary, this study has successfully demonstrated the effectiveness of MATLAB and Machine learning in simulating and detecting transmission line faults and classifying them with high accuracy.

REFERENCES

[1] H.H Goh et al., “Fault Location Techniques in Electrical Power System: A Review,” Indonesian Journal of Electrical Engineering and Computer Science, Vol. 8, No. 1, pp. 206 - 212, 2017.
[2] C. Tharani, M. Nandhini, R. Sundar, and K. Nithiyananthan, “MATLAB based Simulations model for three phases Power System Network,” International Journal for Research in Applied Science Engineering Technology (IJRASET), vol. 4, p. 502-509, 2016.
[3] S. Karekar, V. Thakur and M., “A Novel Scheme of Transmission Line Faults Analysis and Detection by Using MATLAB Simulation,” International Journal of Engineering Research and General Science, Vol. 4, No. 1, 2016.
[4] M. Salehi, et al. “Determining minimum number and placement of fault detectors in transmission network for fault location observability,”International Journal of Electrical Power and Energy Systems, vol. 124, 2020.
[5] S. Rajput and K.T. Chaturvedi, “Transmission Line Fault Analysis by Us- ing Matlab Simulation,” IJREE - International Journal of Research in Electrical Engineering, vol. 3, no. 4, 2016.
[6] "Learning to Optimize" by Yutian Chen, Matthew W. Hoffman, Sergio Gomez Colmenarejo, Misha Denil, Timothy P. Lillicrap, and David Duvenaud. Conference: International Conference on Machine Learning (ICML), 2018.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)