COMILLAS
UNIVERSIDAD PONTIFICIA
ICAI

MASTER’S DEGREE IN INDUSTRIAL ENGINEERING

FINAL MASTER THESIS

FAULT ANALYSIS OF MESHED NETWORKS USING ARTIFICIAL INTELLIGENCE

Author: Alberto Santiago Salazar

Director: Adam Dysko

Co-Director: Ciaran Higgins

Madrid

August 2025

This dissertation was typeset with Word 2016. The font families used are Times New roman and Book Antiqua. Unless otherwise noted, all figures were created by the author using Microsoft PowerPoint® and Python®.
AI tools were used to enhance the clarity and quality of the writing throughout this document.

FAULT ANALYSIS OF MESHED NETWORKS USING ARTIFICIAL INTELLIGENCE

Alberto Santiago Salazar, Author, Adam Dysko, Director, Ciaran Higgins, Director

Abstract — *This project evaluates Machine Learning (ML) techniques for fault detection, classification, and location in meshed distribution networks using voltage readings from smart meters. Two Low Carbon Technology (LCT) scenarios were analysed: a baseline representing the current network and a high LCT penetration scenario. Three fault classification models—Gradient Boosted Decision Trees (GBDT), Support Vector Machines (SVM), and Graph Neural Networks (GNN)—achieved over 99% accuracy in both scenarios, with the SVM model demonstrating superior computational efficiency. A GNN-based fault location model achieved 65% top-1 accuracy and over 98% top-10 accuracy. Real fault data from SPEN validated the SVM and GBDT models, with the SVM correctly classifying most samples. These results demonstrate that AI and ML techniques are effective for fault diagnosis in current and future networks, enhancing reliability and operational efficiency.*

Index Terms — Machine Learning, Fault location, Fault Classification, Graph Neural Networks, Distribution Networks.

I. INTRODUCTION

The distribution network is undergoing transformations caused by the increasing integration of Low Carbon Technologies (LCTs) such as photovoltaics (PV), electric vehicles (EVs), and heat pumps. These technologies bring many advantages, such as the reduction of greenhouse gas emissions and the support to energy independence, which are key to combat climate change, however their they also introduce technical challenges that affect traditional protection and monitoring systems by introducing complexities such as fluctuating fault levels, phase imbalances, harmonic disturbances, and irregular fault current profiles. At the same time, meshed low-voltage network configurations are being explored by DSOs to improve flexibility, redundancy, and resilience. Unlike radial networks, meshed systems have multiple current paths for a same load and can introduce bidirectional flows, which complicates fault analysis. All of these factors can affect the reliability of conventional protection schemes by reducing their sensitivity and selectivity, which is why developing advanced diagnostic methods becomes crucial for maintaining reliability and fast fault response under these new grid conditions. The objective of the project is to develop AI/ML models for fault analysis, with the goal of evaluating their feasibility for real-world deployment by DSOs. The three models that have been developed for fault detection and classification are Gradient Boosted Decision Trees (GBDT), Support Vector Machines (SVM), and Graph Neural Networks (GNN), while the model that has been developed for fault location is another GNN model. These algorithms have been analysed taking into account model accuracy, ease of implementation, computational complexity, interpretability and scalability.

The developed models have been trained and evaluated under two different scenarios: one which resembles the current state of the network and another with high LCT integration, both based on meshed topologies. The training data for these models has been generated using OpenDSS, an open-source distribution system simulator that allows to adapt the load behaviour, to integrate DER, like PV systems, and model different electrical faults. The simulations have been done in real meshed LV network models provided by SPEN, allowing the project to have realistic electrical and topological characteristics.

To validate the models beyond simulated data, real fault event data provided by SPEN has also been used to test the models. This comparison enables the assessment of the models' practical utility, their capacity to generalize to real-world conditions and their potential for future integration into DSO fault management systems. The analysis of all the model will allow to determine whether AI/ML techniques can enhance existing fault detection frameworks and support more efficient and automated network operation strategies in the context of smart grids.

II. THEORETICAL BACKGROUND AND STATE OF THE ART

A. Meshed Distribution Networks

Electrical distribution networks have traditionally been operated in a radial configuration, meaning that the power flows go in one direction from distribution substations to customers following a tree-like structure in which electrical lines are separated into branches without interconnection between them. Because of that, low impedance fault detection and isolation in radial networks are relatively straightforward tasks. The protection devices are coordinated in cascade, which ensures that the device that trips is the closest one to the fault, minimizing the impact of the fault and maintaining service continuity for as many clients as possible. This scheme has proven effective and simple, leveraging the fact that any fault current will originate from the upstream source and flow downstream [1], however, this topology has limitations in terms of reliability and redundancy, since a fault anywhere along a feeder or a line can isolate all downstream customers until the fault is cleared. In order to tackle this problems, distribution companies are moving towards meshed or looped network configurations, especially in urban areas. In these topologies the feeders are interconnected at multiple points, allowing power to be delivered to the loads through different paths, some advantages of these systems are the reduction of power losses, improved voltage profiles due to a reduced voltage drop, more flexibility, the enhancement of power growth postponing costly investments to increase line capacity [2]. However, meshed networks also introduce significant challenges for protection design. The most immediate issue is the increase in short-circuit current levels. The interconnection of parallel lines reduces the total short-circuit impedance compared to radial configurations, causing a current that can potentially exceed the interrupting capacity of circuit breakers and switchgear originally sized for radial conditions. If fault levels are not controlled, equipment may be damaged or fail to clear faults in time. Another major challenge is coordination and selectivity. In a radial feeder, the direction of fault current is predictable and fixed, allowing protective relays to be graded in a clear upstream–downstream sequence. In a meshed system, however, the fault current can flow in either direction, making it difficult to determine the "upstream" relay. Without directional sensitivity, overcurrent protection may trip not only the faulted section but also adjacent healthy feeders, causing widespread outages. Because of this, traditional protection schemes like overcurrent relays and fuses face big limitations.

To address the aforementioned challenges, utilities are moving towards Active Network Management (ANM) techniques. ANM may include real-time monitoring of many elements of the network such as feeder flows and voltages, electric vehicles charging stations or inverters of DERs; automated control of switches, inverters, and other actuators to maintain stability and also integrates information with the marketing control system, production management system and each substation system, allowing the acquisition and sensing, optimization management, risk control, and fault handling of all the grid [3].

In parallel, machine learning and artificial intelligence techniques are increasingly being explored to enhance fault detection capabilities, particularly in dynamic and data-rich environments. These developments are especially relevant given the growing penetration of distributed generation and the increasing demand for reliability, which are driving interest in the 'smart meshing' of feeders. Such an evolution in network topology will inevitably require more advanced, adaptive, and data-driven fault detection and protection methods capable of handling bidirectional flows and rapidly changing system conditions.

B. Low Carbon Technologies

Electrical distribution networks worldwide are experiencing the rapid integration of Low Carbon Technologies, which include distributed renewable generation and new electric loads that support decarbonization. Notable LCTs impacting distribution grids are photovoltaic (PV) solar panels, electric vehicles (EVs), electric heat pumps, micro-CHP (Combined Heat and Power) units, and energy storage systems (ESS). The penetration of these technologies has been increasing through the years and will keep increasing due to policy targets for carbon reduction and consumer adoption of green energy solutions. This can be proved by the fact that many countries have set ambitious goals for decarbonization, EV adoption and electrified heating. Spain's National Integrated Energy and Climate Plan (PNIEC) focuses on reducing greenhouse emissions by 32% relative to 1990 levels, achieve an 81% share of renewable energy in electricity generation by 2030 and transition to a 100% renewable generation by 2050. Additionally, among other initiatives, the plan sets a target of 22.5 GW of energy storage capacity to support the integration of renewable energy sources [4]. In the same way, The UK's Electricity North West (ENWL) reports that penetrations of technologies like PV, EVs, and heat pumps are likely to increase significantly in the near future, affecting LV networks. According to ENWL's most recent Distribution Future Electricity Scenarios report, published in January 2025, it is expected that by 2040 the energy demand will double driven by the adoption of 3 million electric vehicles and 1.2 million heat pumps [5]. This transition means that the once passive distribution network (characterized by one-way power flow from substations to loads) is evolving into an active network with generation and new types of demand. The evolution and penetration of LCTs bring new opportunities and challenges for distribution companies. On one hand, LCTs like rooftop PV and local wind generation can supply clean energy locally, potentially reducing peak power drawn from the grid and losses. EVs and batteries can provide flexible demand or storage that might be leveraged for grid support in the future (vehicle-to-grid services, load shifting, etc.). On the other hand, these technologies were not taken into account in the original design of most distribution systems. Historically, LV feeders were built assuming all customers are consumers only, and network capacity was calculated for certain peak demand per household – with no generation or high-power new loads like EV chargers in mind. As a result, substantial LCT uptake can drive the network beyond its limits unless changes are made.

The increasing adoption of LCTs can create adverse effects on distribution networks. The most common issues identified are: voltage regulation, thermal overloading, increased fault levels, power quality degradation and challenges to protection schemes.

- Voltage Regulation: Distributed generation like PV can cause the voltage to rise in feeders especially at times of low load and high generation, whereas concentrated new loads like EV charging clusters or heat pumps in winter can cause deeper voltage drops. Moreover, if the deployment of LCT is uneven among the different phases of the network, which is something that can happen specially with rooftop PV systems and with home EV chargers, the voltage valance can be even more compromised. That is why maintaining all customers within the required voltage range becomes more challenging and may require advanced voltage control equipment, like on-load tap changers, voltage regulators, capacitor banks or power electronic components like STATCOMS or FACTS. These technologies require advanced monitoring and control strategies. In this context, data driven approaches are crucial for maintaining voltage stability.

- Thermal Overloads: The addition of multiple LCTs demands can push line and transformer loading beyond their rated capacities particularly during evening peaks or seasonal cold periods. Likewise, cables and overhead lines may run hotter due to both increased peak currents and longer durations of high load (e.g., an EV charging for hours), potentially leading to asset aging or failure if not addressed through network reinforcement [6].
- Fault Levels and Protection: With the deployment of many LCTs, the short-circuit current profile of the network changes, this can difficult the protection coordination, especially in meshed or bidirectional flow conditions and can lead to the development of new protection strategies. Even if inverter-based technologies like PV and HPs usually have a limited fault current contribution, a big amount of penetration in a network can still cause this effects.
- Power Quality and Losses: The high penetration of power electronic devices can degrade power quality through the injection of harmonics, voltage flicker, or rapid fluctuations in load and generation. While individual devices meet standards (IEEE 519, etc.), the aggregate effect of dozens of inverters can raise background harmonic distortion. On the losses side, a modest level of local generation can reduce net current flow and losses, but as penetration rises, network losses can actually increase due to circulating currents and periods of reverse flow that force power through more stages of transformation.

Essentially, distribution networks face a more dynamic and less predictable power flow pattern as LCTs proliferate, which affects everything from component aging to the efficiency of operation [6]. Distribution operators are addressing these challenges through a combination of measures, which are the reinforcement of existing infrastructure and the deployment of smart grid elements like sensors and voltage controllers. This project takes advantage of the increase deployment of smart meters to develop ML models for fault analysis, allowing a secure and efficient integration of LCTs.

C. Artificial Intelligence and Machine Learning in Power System Fault Diagnosis

The complexity that has been introduced in the networks by the meshed configurations and distributed resources has increased the interest in more intelligent fault diagnosis techniques. The widespread deployment of smart meters and digital relays, combined with improved computing capabilities, has enabled the increasing application of knowledge-based approaches, such as Artificial Intelligence (AI) and Machine Learning (ML), in power system fault diagnosis over recent decades. These techniques have been used to achieve faster and more accurate fault detection, classification, and location, even under the challenging conditions of modern grids.

Several AI/ML approaches have been developed throughout the years, starting by expert systems in the 1980s-90s, which mimicked human operator actions for fault diagnosis and clearance. In order to manage uncertainty, fuzzy logic systems emerged and have been used for fault classification and even adaptive relay setting; these systems model partial truths rather than binary logic and handle uncertainties in fault patterns.

The most widely used AI-based algorithms for fault location are Artificial Neural Networks (ANNs) due to their ability to learn complex relationships between inputs and outputs, their flexibility and their high precision [7]. Researchers have successfully trained neural network models to recognize different fault types on transmission lines and distribution feeders, achieving high accuracy in simulations.

Beyond ANNs, many other ML techniques have been applied to electrical fault analysis. The main ones are Support Vector Machines (SVMs), which separate data by searching for a linear optimal hyperplane that acts as boundary among classes; decision tree algorithms (DTs), which have a hierarchical shape structure and classify data by asking questions about the data features; and clustering algorithms like k-means, which determine the class of new data points based on their similarity to known training data [8]. Each of these techniques offer unique strengths in handling the dynamic conditions of the modern grids.

Currently, some distribution companies are beginning to implement ML/AI models and techniques to manage electrical faults.

Pacific Gas and Electric (PG&E) has developed an ensemble model based on decision trees which uses smart meter data, asset allocation, weather conditions and load data to predict transformers failures. Between April 2021 to February 2022, over 270 predictions were reviewed, with 64% confirming relevant transformer anomalies [9]. In parallel, PG&E is developing a project aimed to predict sustained outages using meter data, historical outage records and weather data, but it is still in the Minimum Viable Product stage.

State Grid Corporation of China (SGCC) has implemented AI-driven strategies to enable the development of self-healing grids by having different sensors in the grid equipped with AI capabilities which allows them to independently route power and address faults. This strategy has reduced fault resolution time from hours to less than 5 seconds, with the grid being able to automatically locate faults and change its topology [10]. Additionally, SGCC applies machine learning algorithms to identify patterns in historical outage data, which allows them to predict and mitigate faults.

Other leading utilities, like Korea Electric Power (KEPCO), Electricite de France (EDF) and China Southern Power Grid (CSPG) hold patents related with the use of AI techniques in distribution grids. KEPCO focuses on fault diagnosis and management, while CSPG is developing AI-assisted fault detection tools to enhance response times and network resilience [11].

D. Data Acquisition and Utilization

Throughout the past years the availability of data readings from the distribution network has increased thanks to the deployment of Advanced Metering Infrastructure (AMI). Smart meters, which are installed between the customer loads and the network, capture parameters like voltage, current, power or power factor in intervals that can go from second to minutes. This data can provide help in scheduling power plants, operation of subsystems or maintenance for power equipment. These meters also permit two-way communication between the utility and the meter, allowing utilities not only to collect the data centrally but also to control the functioning of the smart meter remotely.

As the speed at which new data is generated increases, the volume of measurements becomes too large to be stored and analysed using traditional database technology, that is why several initiatives on how to use this data are being studied by DSOs to leverage the immense amount of data to better understand the functioning of the network and improve its reliability. One example is state estimation, in which smart meter data is fed into models that estimate voltage profiles across distribution grids, improving system reliability by enhancing network visibility. Another data application is power flow and load forecasting by using past energy consumptions from each smart meter and weather forecast data, allowing to prioritize grid investments by predicting the load at a desired time.

III. METHODOLOGY

This project can be divided in two different parts, the first one is the simulation of the electrical faults and the different LCT scenarios using the software OpenDSS in order to get the data needed to train the ML model. The second part is the training and testing of the ML models using specific Python libraries and the respective analysis of results.

A. Software

1) OpenDSS

The software used in this project to simulate electrical faults and network operating conditions is OpenDSS (Open Distribution System Simulator), an open-source simulation tool developed by the Electric Power Research Institute (EPRI). OpenDSS was chosen for this project since it is designed for comprehensive analysis of electric power distribution systems, it is very useful for studying unbalanced, three-phase networks under a wide range of operating conditions and because all SPEN network models were available in this format.

OpenDSS solves power flow problems using the fixed-point iterative method, which enables the accurate modeling of unbalanced and non-linear systems. The software operates using a text-based input system through .dss script files, where each file defines various electrical components in the system such as transformers, lines and loads (resistive, inductive, or capacitive). Each element is assigned to a bus, which typically contains three electrical nodes corresponding to the three phases. OpenDSS can also simulate electrical faults. Faults can be defined at any bus within the network by specifying parameters such as the phases involved (e.g., A-B, B-C, A-Ground) and the fault resistance in ohms. This allows the modeling of single-phase, two-phase, and three-phase short-circuit events, providing flexibility in fault characterization. Simulation outputs in OpenDSS include detailed electrical quantities such as voltage magnitudes and angles, current flows, real and reactive power flows, and network losses. These results are stored in output files or can be exported through COM interfaces, enabling further analysis using external tools such as Python, as done in this project.

2) Python

Python is the programming language used in the project for its extensive number of scientific libraries and its use in machine learning applications. It was used to manage data preprocessing, to interact with OpenDSSS and to implement the ML models. The main libraries used were:

- py_dss_interface. This library was used as a wrapper for the OpenDSS interface, enabling simulations of different networks, applying faults, extracting data and automating workflows directly from the Python scripts.
- ScikitLearn. This library is widely used in ML applications and was used to implement the GBDT and SVM models. The library also allows hyperparameter tuning and performance evaluation metrics such as confusion matrices or classification reports.
- Tensorflow and Keras. This library, which was developed by google, was used to develop the GNN models. Along with its application programming interface Keras, Tensorflow allows graph-based computation, making this library suitable for modeling meshed LV networks

B. Network Model

The project focused on the study of a specific SPEN network, which was selected due to its meshed topology with three different feeders and its amount of fault alarms that were recorded by smart meters of that network.

The model was obtained in *dss* format, which had all the electrical components along with their parameters. A *Json* file which contained the exact coordinates of every node and the id names of the smart meters of the network was also obtained.

The network used in this project is a low-voltage distribution network located in an urban area close to Liverpool, it has three feeders that represent step-down transformers that convert voltage from 11 kV to 415 V and has 341 different buses. The network is structured around a central main loop, which gives it its meshed topology and provides multiple paths for the energy to reach the loads. In addition to the main loop, the upper-right section of the network is a radial branch that connects to different loads and to a feeder. Figure 1 shows the location of all the buses of the network, where there are 125 buses highlighted in yellow which represent the consumers or electrical loads with smart meters in the system. These costumers have an average maximum consumption of 5 kW, which means that the grid has a peak demand of 625 kW.

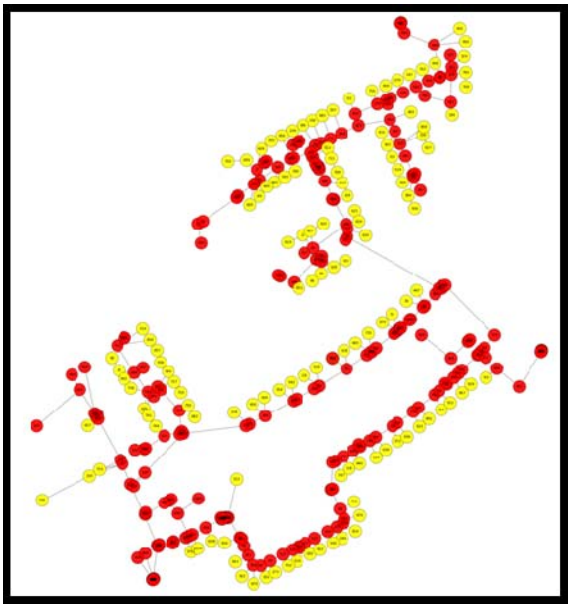

Figure 1. Network Loads

C. *LCT Scenarios*

In order to evaluate the performance of the models under evolving grid conditions two different LCT penetration scenarios were modelled.

1) *Baseline Scenario (Scenario 1)*

This scenario is a representation of the current state of the network, without extra LCTs beyond the ones that are already present. In order to simulate this scenario, all the loads of the simulation were assigned a load-shape that corresponded with a typical residential consumption profile with no influence of DG. The load-shape has two peaks, one in the morning that corresponds to the residential and commercial start up activities and other one in the evening when customers are at home, and during night-time the demand is relatively low. This baseline configuration serves to assess the models accuracy in traditional networks.

2) *High LCT Penetration Scenario (Scenario 2)*

This scenario acts as a projection of how the distribution network will operate under a context of high LCTs adoption. In the simulations, three key LCTs were be implemented across the network to reflect their growing presence in LV grids. PV systems were included in the model as two large photovoltaic plants, while the effect of EVs and heat pumps was modelled by modifying the demand profile of the loads. By doing these modifications, the power flows were bidirectional and more complex. These changes allowed to analyse the resilience and adaptability of the developed models, and if their performance remained within accepted values, it would demonstrate their suitability for supporting the integration of these emerging technologies in a secure and controlled way.

The new load shape was configured following the following parameters:

In the case of heat pumps, projections and policies estimate that by 2030 in Europe and the UK approximately 40% of households will have electric heat pumps [12]. These systems will have low power continuous modes that will operate constantly throughout the day to maintain consistent indoor temperatures. The average power that heat pumps will consume will be similar to the base household load. Taking everything into account, the effect in the load shape will result in an additive load equal to 0.4 per unit (pu) across all of the base load.

For electric vehicles, the adoption forecast estimates that there will be a 30% penetration among households by 2030. To reflect the expected charging behaviour, the demand was concentrated during night time hours, since smart charging strategies like time-of-use (ToU) tariffs will be adopted. These strategies will flatten the overall demand curve by fitting this new demand during times when prices are low or when the network capacity is not being used. As a result, the overnight valley in the base load shape will evolve into a constant curve with a higher average demand.

All of these changes can be seen in Figure 2, where both load shapes (baseline in gray and high penetration in blue) are compared.

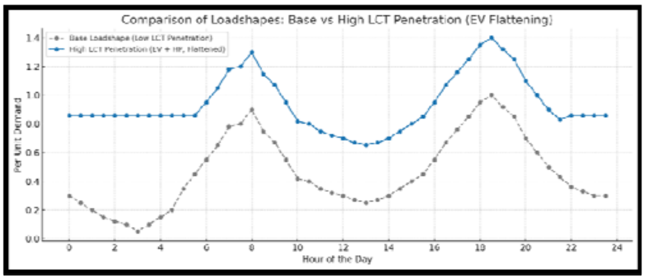

Figure 2. LCT Scenarios Load-shape

The two PV plants were implemented in the network by using a specific OpenDSS command. This command allowed defining parameters such as the number of phases of the system, the bus where the plant was going to be connected, the nominal voltage of the plant, the peak power generation under maximum solar irradiance, the installation's power factor and a solar generation profile. Both plants were configured as three-phase systems, each generating 75 kW at unity power factor (pf = 1). Since the total network consumption was approximately 625 kW, the combined PV output represented about 25% of the total demand. The daily generation profile used for both plants is a generic solar curve, which peaks at midday when solar irradiance is highest.

D. Data Recording

The only electrical variable that was recorded in the project was rms voltage magnitude. This was decided based on the real fault data that was provided by SPEN, which consisted of rms voltage readings and no current measurements, due to customer privacy considerations. Because of this, the simulations were designed to only capture voltage measures, ensuring that the developed models could be applied directly to SPENs data to predict scenarios. The daily mode was used in the OpenDSS simulations to record the voltage magnitudes for every step of the daily load shape used.

Since the ML models used for the project require two main inputs, which are X (data features) and y (targets), three arrays were generated for each simulation to match this requirement. The feature set (X) consisted of a Voltage_Magnitude_Readings Array, and two separate targets arrays (y1 and y2) were created, the Fault_Type_One-Hot Array for the classification model and the Fault_Location_One-Hot Array for the fault location one. The structure of these arrays is the following:

- Voltage_Magnitude_Readings: This array is the primary input for the ML models. Its shape varies depending on the model, for the Gradient Boost Decision Tree and for the Support Vector Machine the shape is (number of nodes with smart meters, 1) and for the Graph Neural Network the shape is (number of buses, number of voltages per bus), where the number of voltages is 3, since each bus has three nodes with a different voltage. Each column represents a different bus of the network and each row contains the steady state voltage magnitudes for that bus.
- Fault_Type_One-Hot: This array is the label for the fault classification model. Its shape is (number of fault types, 1), where each column represents all the possible scenarios (no fault, SLG on each of the three phases, LL for each phase pair (A-B, A-C, B-C), and three-phase faults). All entries are zeros except for a 1 in the columns corresponding to the fault type that is being simulated.
- Fault_Location_One-Hot: This array is the label for the GNN fault location model. Its shape is (number of buses in the network, 1). All entries are zeros except for a '1' in the position corresponding to the bus where the fault is located.

The data collection process is done with a Python function collect_data, which simulates the scenarios and saves each array into their respective dataset (X_data, Y_fault_type, Y_fault_location).

E. Fault Simulation using OpenDSS

The next step after having all the scenarios ready is to simulate all the faults. This process employed the Monte Carlo method, which consists of repeatedly doing random samples of an event to create a large number of scenarios, to ensure that the data set had a wide variety of possible fault conditions.

The faults that were simulated included the most common types of faults in distribution systems: single line-to-ground faults on each phase; line-to-line faults between phases A-B, A-C and B-C; and three-phase faults, which makes a total of 7 different fault scenarios. Since the objective of the project was to evaluate model performance for both low and high impedance short circuits, the fault impedance values ranged between 0.01 to 1 Ω.

The fault generation process was done in Python, which was used as the interface to interact with OpenDSS, following the next sequence:

1. Scenario definition: A fault_scenarios array was created, containing the name of each fault type, along with the phases that were involved.
2. Bus selection: A fault_bus array was created containing a predefined number of buses, which were randomly chosen from a list of all the network buses.
3. Automation loop: A Python loop was used to automate the execution of all fault scenarios. The loop iterated over every bus in fault_scenarios, simulating all the defined scenarios in fault_bus. For each iteration, a random fault resistance was chosen form the range 0.01 to 1 Ω. Then, each parameter (location, fault type, and impedance) was passed to OpenDSS via the py-dss-interface library for them to be simulated.
4. Data recording: after the execution the resulting electrical measurements were recorded using the collect_data function.

F. Training the Machine Learning Algorithms

This section describes the design, configuration and training of the ML models. The specific architecture, code and hyper parameters of each model is explained along with its results and achieved accuracy.

1) Gradient Boosted Decision Trees (GBDT)

The GBDT model was implemented using the Scikit-learn library's GradientBoosingClassifier. The models objective was to classify each sample into one of eight possible scenarios using Voltage_Magnitude_Readings as input features (X) and Fault_Type_One-Hot as target (y).

The code used to implement the model in Python had the following procedure:

- The target array was modified from a one-hot encoding to an integer label, which means that in the original array every sample had all the columns as zero except the one which represented the target with a one but the modified array consisted of only one number which corresponded to the fault class (e.g. y'=0,0,0,1,0,0,0 becomes y=4).
- The dataset was split into training and testing sets with an 80/20 ratio for training/test. Each set had a different purpose, the train set was used to fit the model, enabling it to learn the relationships between inputs and outputs, while the test set was used to evaluate the models accuracy using unseen data.
- The ML model was initialized and the hyper parameters were adjusted. The hyperparameters used in the model were number_of_trees, which sets the number of boosting stages that the model has, learning_rate, which controls the size of the steps taken by the optimizer, max_depth, which sets the maximum branches of the individual trees and random_state

which controls reproducibility.

- The model was trained using the training data set.
- The accuracy was calculated with the test data set.

In order to set the optimal hyperparameters a grid search with cross-validation was performed, giving the following results:

- o number_of_trees =100
- o learning_rate=0.1
- o max_depth=3
- o random_state=42
- o verbose=1

After training the models with datasets from the LCT scenarios 1 and 2 defined in section III.C, the following results were obtained:

For the model from scenario 1 (baseline) the accuracy was of 99.82% with a training time of 40 minutes and 50 seconds. The confusion matrix can be visualized in Figure 3, where it can be seen predicted class with the highest misclassification rate was class 0, which is no fault.

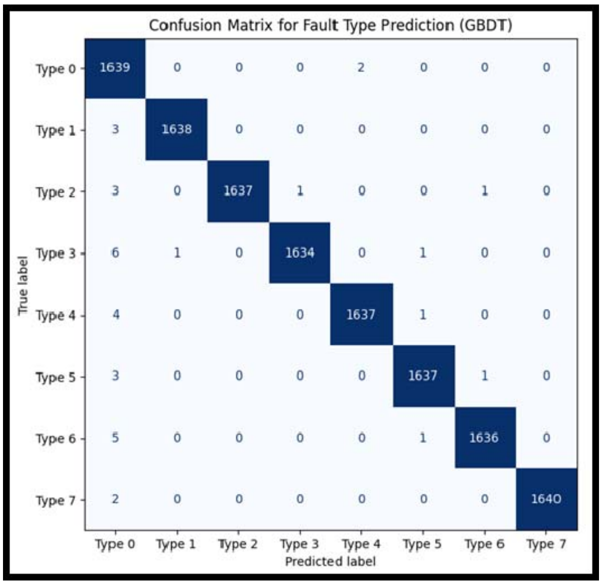

Figure 3. GBDT Scenario 1 Confusion Matrix

For the model from scenario 2 (high LCT penetration), the models accuracy is 99.76% with a similar training time to model 1 of 42 minutes and 14 seconds. Its decision matrix can be visualized in Figure 4.

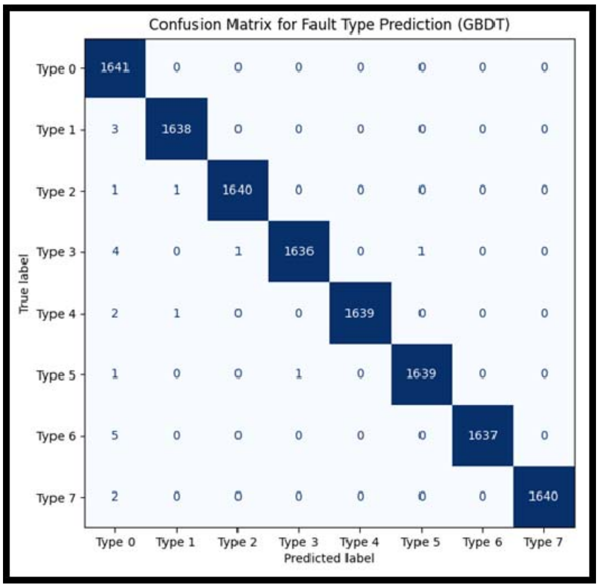

Figure 4. GBDT Scenario 2 Confusion Matrix

One of the individual trees that is part of the final ensemble of trees can be visualized in Figure 5, where it can be seen the typical tree structure. Each internal node of the tree includes:

- The decision rule for the node that decides the split.
- The friedman_mse value, which represents the mean squared error produced by the split [13].
- The number of samples that fall to that node.
- The predicted output value for the node, which in a GBDT represents the "pseudo-residuals" that are added to the overall prediction. The model's final output for a sample is the sum of the values from all the trees it passes through.

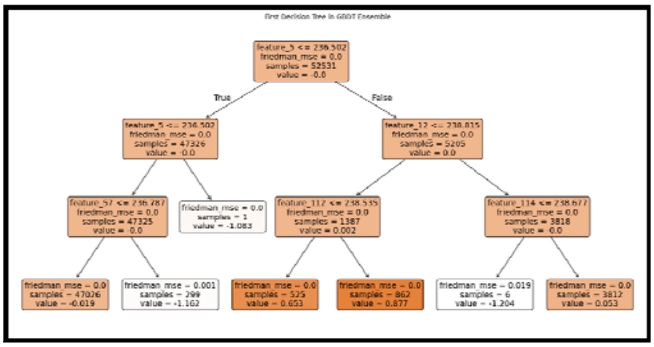

Figure 5. Decision Tree in GBDT Ensemble

2) Support Vector Machine

The SVM model was implemented using the Scikit-learn library's SVC class. Its objective was also to classify each sample into one of eight possible scenarios, using the Voltage Magnitude Array as input features (X) and the Fault Type One-Hot Array as the target (y).

The code used to implement the model in Python followed the same general process as the GBDT model: transforming the target from one-hot encoding to integer labels, splitting the dataset into training and testing sets, initializing the model, training it on the training data, and finally evaluating accuracy on the test set. The only difference was the hyperparameters settings, which were: the kernel type; gamma ‘σ’, which is a kernel coefficient that controls the shape of the decision boundary; the regularization parameter; and random_state.

In order to set the most optimal hyperparameters a grid search with cross validation was used, resulting in:

- Radial basis function (RBF) kernel, which is a common kernel for SVM whose formula is $K(x,x')=exp(-\|x-x'\|^2/(2\sigma^2))$, where x are two feature vectors [14].
- σ='scale', which calculates automatically a suitable gamma value based on the training data.
- C=1.0
- random_state=42

After training two models, with datasets from the LCT scenarios 1 and 2, the results obtained were the following:

The first SVM model achieved an accuracy of 99.76% and a training time of 5 minute and 7 seconds. As seen in Figure 6, the class with the most misclassifications is the no-fault class (class 0).

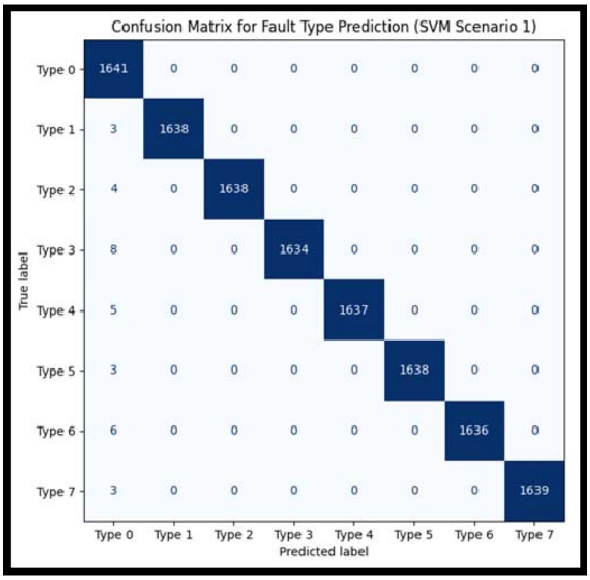

Figure 6. SVM Scenario 1 Confusion Matrix

For the scenario 2 model, an accuracy of 99.73% was achieved, which shows that the model maintains its performance despite the increased complexity of high LCT penetration. The model’s confusion matrix can be visualized in Figure 7.

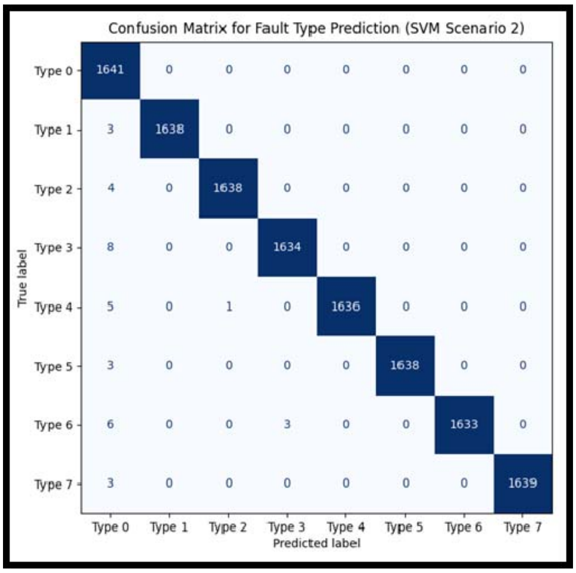

Figure 7. SVM Scenario 2 Confusion Matrix

3) Graph Neural Network

Graph Neural Networks were selected among all the possible Neural Network architectures to take advantage of the inherent graph topology of electrical distribution networks, since all buses have to be interconnected by lines, and each bus electrical state is influenced by that of its neighbors. By representing the network as a graph it is possible to explicitly model these relationships.

Two different GNN models were developed, one dedicated to fault classification, like with the GBDT and SVM models, and other one for fault location, which was able to localize the bus were the fault had occurred. Both models share the same input structure, which consists of:

- A graph where each node corresponds to a bus in the network with three voltage magnitudes that represent each phase of the bus.
- The network adjacency matrix, which was obtained by analysing the dss file and seing the connection among all buses.

This models, unlike the previous GBDT and SVM models, use the voltages measurements from all the nodes in the network. This is crucial, since if only the voltages from smart meters were used, the resulting adjacency matrix for the measured buses would result in a matrix were all nodes would be connected to each other, since all the nodes that previously represented different bifurcations and intersections would collapse into a single node.

a) Fault Classification Model

The GNN model was implemented in Python using the Spektral library for handling graph-structured data within the Tensorflow framework. The methodology has the following stages:

- The adjacency matrix is pre-processed using symmetric normalization, which is a standard procedure for NNs that ensures that the information is passed between connected nodes. The normalized matrix and the Voltage_Magnitude_Readings dataset are then encapsulated within a custom Spektral dataset class, which transforms the data in a format that can be processed by the GNN, where each sample is treated as an independent graph.

• The data is split into training and testing sets, and then a BatchLoader, which is in charge of feeding batches of data to the model during training, assigns every sample to a batch.

• The model is defined with the following parameters:

o Three graph convolutional layers with 64 neurons each with a ReLU activation function.

o A graph-level pooling layer which aggregates all the learned features of all the nodes in to a single vector.

o A final dense layer with a softmax activation function. This activation function converts a vector of real numbers into a probability distribution, where each element represents the probability of belonging to a specific class, and the sum of all elements equals 1, which is why this layer takes the vector created by the graph-level pooling and outputs a probability distribution over the number of possible fault types.

• The model is trained using the adam optimizer, which is typical optimization algorithm in NNs, and categorical_crossentropy as the loss function that is being minimized. The training process has an early stopping function which prevents overfitting by stopping the training when the model's performance on the validation set stops improving.

After training two identical fault classification GNN models with datasets from the LCT scenarios 1 and 2, the models' performances were the following:

For the model trained with data from scenario 1 (baseline), the accuracy was a 99.71% and the model's training time was 30 minutes and 2 seconds. The model's confusion matrix can be visualized in Figure 8.

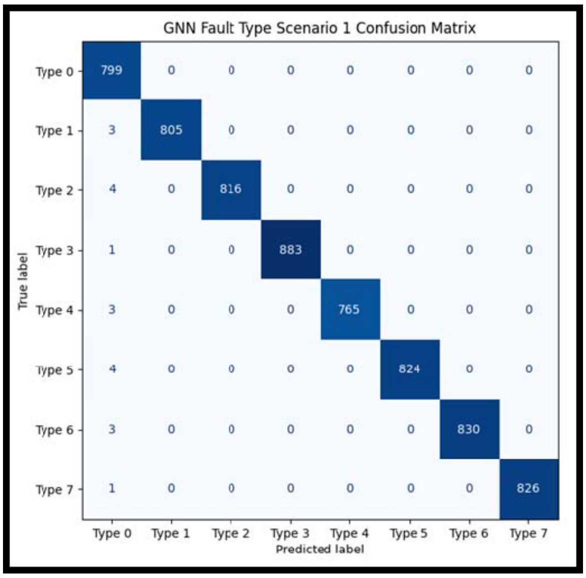

Figure 8. GNN Fault Classification Scenario 1 Confusion Matrix

For the model trained with data from scenario 2 (high LCT penetration), the accuracy was a 99.74% and the model's training time was 32 minutes and 27 seconds. This model's confusion matrix can be visualized in Figure 9.

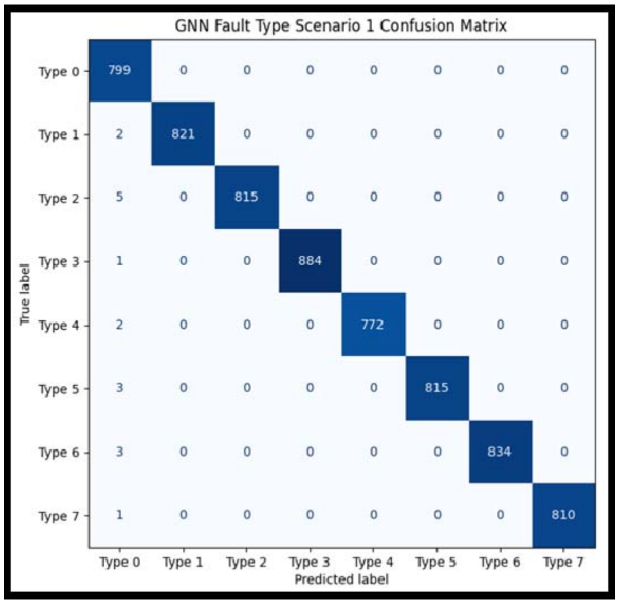

Figure 9. GNN Fault Classification Scenario 2 Confusion Matrix

4) Fault Location Model

The methodology that has been used to implement the GNN fault location model is similar to the one used for the GNN fault classification model, including the data preprocessing and splitting to training and testing sets. Therefore, only the distinct model architecture will be described:

The fault location model is designed for a node level classification task to identify which bus is faulted. The architecture of the model is defined with the following parameters:

• Three graph convolutional layers with 60 neurons each with a ReLU activation function.

• A final dense layer is applied, which processes each node's feature vector independently. This is followed by a softmax activation function, which produces a probability distribution across all buses. The predicted node where the fault has occurred is the one with the highest probability.

Two location models were trained with datasets from the LCT scenarios 1 and 2, to evaluate their performance. Both models trained with 20 epochs and took around 80 minutes to train.

The model trained with data from Scenario 1 (baseline) achieved a top-1 accuracy of 65.83%, which means that the model correctly identified the faulted bus as its single highest-probability prediction in over 65% of cases. This accuracy improves if we take the top ranked nodes that have the highest probability. If we take the top-3 buses the probability of the faulted bus being among the three predictions increases to

87.73%, if we take the top-5 buses the probability increases to 94.10% and if we take the top-10 buses the probability increases to 98.36%.

The model trained with data from Scenario 2 (high LCT penetration) achieved a slightly lower accuracy of 62.80%, which may be because the power flows from scenario 2 are more complex. Similar to the first model, the accuracy increased with a higher number of top predictions. The specific increase in accuracy for both models can be seen in Figure 10.

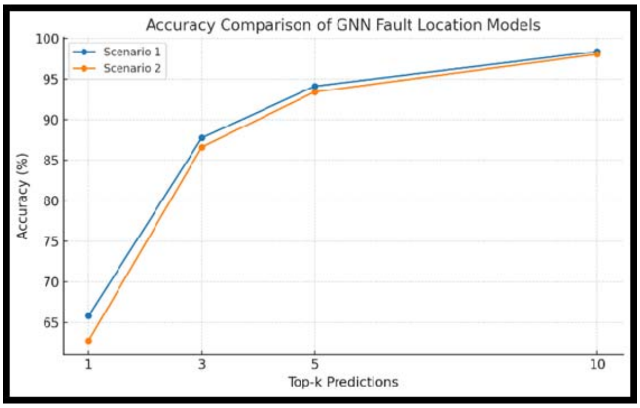

Figure 10. Accuracy Comparison of GNN Fault Location Models

The model predictions were visualized on the network. Figure 11 shows in yellow the top-10 buses with the highest probability of being the faulted bus locations for a sample, with the correct faulted bus being among the top-10 set, and in red the other buses in the network. The image shows how all the top-10 predicted nodes are clustered in a single localized area.

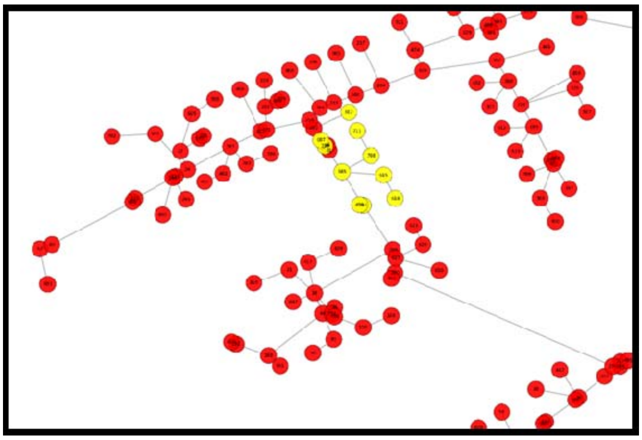

Figure 11. Top-10 Location Model Predictions

This clustering proves that the GNN is not simply learning to classify the correct node in isolation, but is successfully leveraging the graph structure to understand how the fault propagates. With this result, even if the model doesn't predict the exact faulted bus, it would allow operators to narrow down their search to a specific area of the grid.

G. Testing with Real Data from SPEN

To evaluate the real world applicability of the ML models, real fault data was used to test the GBDT and SVM models. This data was provided by SPEN and consisted of alarm records from the network, along with smart meter rms voltage readings taken every 30 minutes for the day on which each alarm occurred.

In order to be able to use the data in the model a preprocessing was done to the raw dataset for it to be in the same format as the model's input:

- The raw dataset contained various types of alarms, which included overvoltage, frequency deviation, under voltage and power outage. Since this study focuses on short circuits, only under voltage and power outage data was considered.
- The voltage readings were recorded in 30 minute intervals, starting at 00:00 from that day. Due to that, only the reading immediately following the alarm timestamp was selected. For example, if an alarm occurred at 02:18 the voltage readings used would be from 02:30. This can lead to readings that do not correspond to the fault, since if these faults are detected by the network they can be cleared in a very short time.
- Erroneous or faulty meter data was removed. For example, one smart meter consistently reported a voltage of 0V for all measures so it was removed from the dataset.
- The location of the meters that had available readings was identified in the network model, the network under analysis contains 125 smart meters, but only 37 had readings for the relevant events.

After preprocessing the data, the final dataset contained 70 samples, each containing voltage readings from 36 smart meters. Figure 12 shows a heatmap of the final dataset, where it can be seen how the majority of the samples do not have any voltage variation and all the voltages are around the 240 V range. However, there are 7 samples that could suggest a possible fault event, since some of their nodes have notable voltage drops.

This dataset was not used to test the GNN models, since these models require readings from all nodes of the network, which were not available, but for future implementations, machine learning techniques for imputing missing data could be used to implement this model in real-world scenarios.

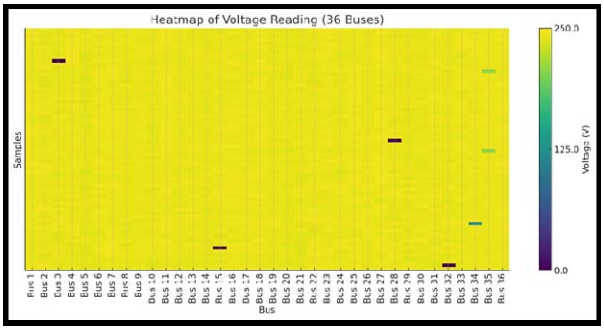

Figure 12. Real Fault Data Heatmap

The SVM and GBDT models were retrained using a simulated dataset, using only the readings from the nodes that had smart meters in the real dataset. After training these models, the real data was analysed to see the predictions.

The SVM model classified 64 of the samples as 'no fault' and the remaining 4 samples were classified as 'single phase fault' two of them in phase 1, one in phase 2 and one in phase 3. The samples that were predicted as faults correspond to the cases with the lowest voltage recorded in one of their nodes, which as seen in Figure 12, are sample 10, 33, 64 and 69. These results are considered reasonable since the fault predictions align with the expected scenario in the presence of single nodes with low voltages and the samples where voltage remained constant throughout all nodes were correctly classified as 'no fault', with the exception of the three cases that had voltage readings bellow the typical values but not significant enough to be clearly classified as a fault.

The GBDT classified the majority of the samples as 'phase to phase faults', being 53 of them B-C faults and 16 A-C faults. This results do not seem to be correct predictions, since they are not consistent with the observed voltage profiles and they do not resemble the predictions made by the SVM model. Because of this, the predictions of the GBDT model are considered as incorrect.

IV. RESULTS ANALYSIS

After developing and training all the machine learning models, several key findings were established.

All the fault classification models (GBDT, SVM and GNN classification) had a high performance across both simulated LCT scenarios, being able to correctly classify the samples over 99% of the time. This proves that ML-based approaches using smart meter data (i.e. using rms voltage only) are a viable and effective solution for fault detection and classification in distribution networks. These fault analysis methods combined with predefined operational response strategies tailored to each fault type detected, could significantly improve network reliability and reduce the impact of faults. In addition, the fact that the models that were trained with the dataset from the high LCT penetration scenario maintained their high accuracy, which only a slight performance drop compared to the baseline scenario, confirms that these models are robust to the evolving grid conditions and can still be used in the future distribution grids with the same credibility. Table 1 shows the exact performance of all the classification models in both scenarios.

Table 1. Classification Model Accuracy (%)

Classification Model	Baseline Scenario	High LCT penetration Scenario
GBDT	99.82	99.76
SVM	99.76	99.73
GNN	99.71	99.74

Regarding the computational performance there were some notable differences observed in the time that took every model to train with a similar data set each. As summarized in Table 2, the SVM model had the shortest average training time, with 5 minutes, followed by the GNN classification model, with 31 minutes, being the GBDT the classification model that required the most training time. These differences are due to the algorithms that each model uses. While SVM are more lightweight in computation, the sequential nature of the GBDT makes it a computationally intensive to train. This training time difference is a factor that has to be taken into account by DSOs when deciding which machine learning model is the most suitable one.

Table 2. Model Training Time (min)

Model	Average Training Time (min)
GBDT Classification	41
SVM Classification	5
GNN Classification	31
GNN Fault Location	80

The GNN fault location model had a different performance compared to the classification models. For Scenario 1, the model was able to correctly detect the fault in with an accuracy of 65.83% for the top-1 prediction, this accuracy improved drastically if the prediction set was expanded. The top-5 predictions reached a 94.1% accuracy and a 98.36% accuracy was obtained for the top-10 predictions. In a similar way, the model for Scenario 2 maintained a similar performance than the baseline, only having a slightly lower accuracy, but still achieving an accuracy of over 98% for the top 10 predictions. This result is particularly good if it is compared to a random baseline, which would have a probability of 0.3% for one single bus and a 3% for ten random buses. Moreover, after representing the top-10 predictions of the model in the network all the predicted buses were clustered in a single area of the network, proving that the model really leverages the graph structure and understands how faults propagates.

After evaluating the SVM and the GBDT classification models with real data from SPEN showed a notable performance difference between models. The SVM model was able to correctly classify the majority of the dataset, identifying the samples with the biggest voltage drops as single phase faults and classifying the remaining samples with constant nominal voltage as 'no fault'. On the other hand, the GBDT model didn't predict correctly any samples, since it classified the majority of them as phase-to-phase faults without any voltage readings that support these results. This result can help decide which method is more suitable to implement, SVM may be a more attractive option based on this results, however, it also has to be taken into account that the available voltage readings on every sample was very limited, which likely contributed to the low performance of the GBDT model.

Overall, these results confirm that ML-based techniques can deliver high accuracy and computational efficiency. In addition, the use of GNN-based approaches brings advantages to fault location since they leverage the network topology.

V. RECOMMENDATIONS FOR IMPLEMENTATION

This section provides some potential improvements that can be implemented by SPEN or any DSO that decides to use machine learning models for fault analysis in distribution networks. This was elaborated by observing the limitations and challenges encountered during the development of the project. The improvements are the following:

- Increase the number of input features for the model. This enhancement would improve the model's accuracy and more scenarios could be simulated, giving the model more flexibility. In this project, only voltage readings were used as inputs for the model. The use of current readings, which were not used in this project due to client data privacy but are available for the DSO, would allow the model to classify faults with more precision. In addition, using external factors like weather conditions or seasonal patterns could also be used to increase the models complexity.
- Increase the number of target fault types. This project only focuses on analysing short circuits, however, the detection of other fault conditions like overvoltage, open circuits or voltage sags could be implemented. Since these models have the ability to understand all kinds of patterns they are suitable for multi-class classification. This would also improve the models functions by supporting more comprehensive fault monitoring strategies.
- Update the smart meter data acquisition strategy. One of the biggest challenges when analysing the real smart meter data provided by SPEN was that the voltage was only recorded every 30 minutes. Because of this, the voltage reading after a fault alarm could happen after the fault had been already cleared, not allowing for classification or location. Increasing the sampling frequency is not a feasible solution due to the amount of data that it would generate, therefore, a proposed strategy would be to configure smart meters to trigger additional recordings every time that an alarm is detected. In that way, the needed samples would be recorded, while minimizing the storage increase.
- Increase the smart meter coverage. The analysed network had 125 loads, which should be connected to a smart meter, however, in the real dataset only voltage readings from 36 smart meters were available per sample. If ML strategies are going to be implemented in the network, more data is needed to locate faults accurately. That is why increasing the number of smart meters that report their measurements would be very beneficial for model performance.
- Use validation techniques to improve data quality. During the preprocessing stage, it was detected that some of the smart meter readings were faulty and had to be removed, since they could distort the result. The use of validation techniques like range checks or outlier detection, can help ensure that all data points are correct or in a logical range. So before the data is feed to the model these techniques can detect anomalous readings and avoid corrupted samples.
- Use machine learning techniques to handle missing data. If a data set has an erroneous measure is detected, or the data from a smart meter is missing, a method to fill missing values is necessary. Some machine learning techniques, such as K-Nearest Neighbors (KNN) imputation or Multivariate Imputation by Chained Equations (MICE), can be used to supply missing values [15].

VI. CONCLUSION

The objective of the project was to evaluate the feasibility and performance of three supervised machine learning approaches for fault detection, classification and location in meshed distribution networks. In order to see if these models were able to maintain their predictive performance under challenging future scenarios, two cases were analysed, the current network configuration and a scenario with high low carbon technologies penetration.

All the classification models achieved an accuracy above 99% across both scenarios, which confirms that ML-based methods are viable and reliable complements to traditional protection schemes and will be able to be used under more complex operational conditions introduced by DER penetration.

After analysing the performance of the classification models, the one that stands out is the SVM since even if the three models achieved similar accuracy levels when tested with simulated data, the computational cost of the SVM was significantly lower, with training times up to seven times shorter than the other models. In addition, despite the non-ideal conditions for testing due to the low amount of real data, when testing GBDT and SVM with SPENs real data, only the SVM model correctly predicted the majority of the cases, reinforcing its suitability for its deployment.

The GNN fault location models demonstrated strong performance, with a top-1 prediction accuracy exceeding 66%, which increases to over 90% with the top-5 predictions and almost 99% when taking the top-10 predictions. Implementing this model can help DSOs to increase reliability by clearing faults much faster thanks to knowing the exact bus were the fault occurred and it highlights the benefits of leveraging the network's graph structure.

Overall, the integration of these ML approaches into distribution network fault management systems could bring substantial benefits: faster fault detection, better classification for targeted response and increased location accuracy, especially in meshed LV networks where traditional methods face inherent challenges. Moreover, the demonstrated robustness under high LCT penetration suggests that these techniques can support the ongoing energy transition, ensuring protection systems remain effective as networks evolve. However, despite the promising results of many research studies about the use of AI in fault management, there are no known commercial AI-based protection devices available today that would replace conventional relays such as IDMT, differential or distance protection. This is due to the high standards for safety, reliability and predictability required for grid protection systems, which highlights the further need for research in this domain to develop solutions that can take advantage of the AI/ML fault classification and location models.

Future work should focus on improving the models with more input features, like current or weather conditions; extending the target fault types to cover more fault scenarios; implementing data pre-processing techniques to detect wrong measurements and handle missing data; and improving the smart meter's data acquisition strategies to ensure higher temporal resolution, enabling a more accurate fault analysis.

APPENDIX. ALIGNMENT WITH THE SUSTAINABLE DEVELOPMENT GOALS

This project aligns with several of these goals, which are closely related to energy and climate action:

SDG 7: Affordable and Clean Energy. This project helps to increase stability and reliability in energy systems that integrate renewable energy sources by improving fault detection with the use of ML. This facilitates the integration of LCTs such as solar photovoltaic panels, electric vehicles and heat pumps, which makes these technologies more accessible for everyone while ensuring that the grid remains stable and secure during its operation.

SDG 9: Industry, Innovation and Infrastructure. The use of AI methods to detect, classify and locate faults is an innovative way of fault management in distribution networks, which if it is implemented by DSOs it can help increase the reliability and advance in the use of AI technologies. This project is also closely aligned with the need of resilient infrastructure, since it supports the automation and digitalization of distribution network infrastructure, by leveraging the readings of smart meters that are being installed in all customer loads for predicting the state of the network.

SDG 11: Sustainable Cities and Communities. This project supports this objective, whose aim is to make more safe, resilient and sustainable cities. The use of meshed topologies for distribution networks help to improve resilience in the electricity supply and this project enables the use of these kind of topology.

SDG 13: Climate Action. One of the central challenges in achieving decarbonization targets is the safe and reliable integration of LCTs into the distribution grid. Without effective fault management strategies, increased DER penetration can destabilize the network, limiting the scale at which clean technologies can be deployed. This project addresses that challenge by providing a methodology that facilitates the secure and stable operation of grids under high levels of LCT integration. Thanks to this, DSOs can control and ensure that the maximum capacity of renewable technologies is installed on the grid safely, directly supporting the energy transition.

In summary, this project contributes to the access to affordable and clean energy for everyone, to more resilient and automated infrastructure, to a more sustainable urban development and to climate change mitigations, all by using AI to analyse faults in meshed networks with LCT penetration

REFERENCES

[1] S. Javadian, M.-R. Haghifam, S. Bathaee and M. F. Firoozabad, "Adaptive Centralized Protection Scheme for Distribution Systems with DG Using Risk Analysis for Protective Devices Placement," International Journal of Electrical Power & Energy Systems, vol. 44, pp. 337-345, 2013.

[2] M. R. M. Cruz, D. Z. Fitiwi, S. F. Santos, S. J. P. S. Mariano and J. P. S. Catalão, "Prospects of a Meshed Electrical Distribution System Featuring Large-Scale Variable Renewable Power," Energies, vol. 11, no. 3399, 2018.

[3] Y. Su, Z. Dong and Y. Zhang, "Active distribution network management system for the new power system," in 2025 2nd International Conference on Smart Grid and Artificial Intelligence (SGAI), 2025.

[4] Gobierno de España, "Plan Nacional Integrado de Energía y Clima (PNIEC) 2023-2030," Ministerio para la Transición Ecológica y el Reto Demográfico, Madrid, Spain, 2023..

[5] Electricity North West Limited, "Distribution Future Electricity Scenarios 2024," 2025. [Online]. Available: https://www.enwl.co.uk/about-us/future-forecasting/dfes/.

[6] N. Damianakis, G. R. C. Mouli and P. Bauer, "Grid impact of photovoltaics, electric vehicles and heat pumps on distribution grids — An overview," Applied Energy, vol. 380, no. 125000, 2025.

[7] H. Rezapour, S. Jamali and A. Bahmanyar, "Review on Artificial Intelligence-Based Fault Location Methods in Power Distribution Networks," Energies, vol. 16, no. 4636, 2023.

[8] M. M. Zaben, M. Y. Worku, M. A. Hassan and M. A. Abido, "Machine Learning Methods for Fault Diagnosis in AC Microgrids: A Systematic Review," IEEE Access, vol. 12, pp. 20260-20298, 2024.

[9] Pacific Gas and Electric Company (PG&E), "Electric Program Investment Charge (EPIC) 3.20 Data Analytics for Predictive Maintenance," 2023

[10] Fundamentals, "China uses AI to swiftly and successfully address grid outages," 30 05 2023. [Online]. Available: https://fundamentals.tech/china-uses-ai-to-swiftly-and-successfully-address-grid-outages/. [Accessed 7 2025].

[11] Power Technology, "Artificial intelligence: who are the leaders in AI-assisted power fault monitoring for the power industry?," 5 01 2024. [Online]. Available: https://www.power-technology.com/data-insights/innovators-ai-assisted-power-fault-monitoring-power/?cf-view.

[12] ecoexperts, «Which Countries Are Winning the European Heat Pump Race?,» 21 04 2023. [En línea]. Available: https://www.theecoexperts.co.uk/heat-pumps/top-countries. [Último acceso: 7 2025].

[13] H. D, «Decision Trees Splitting Criteria For Classification And Regression,» [En línea]. Available: https://machinelearning-basics.com/decision-trees-splitting-criteria-for-classification-and-regression/#:~:text=Friedman%20MSE%20criteria%20computes%20also,left%20side%20of%20the%20node. [Último acceso: 07 2025].

[14] Wikipedia, «Radial basis function kernel,» 3 06 2025. [En línea]. Available: https://en.wikipedia.org/wiki/Radial_basis_function_kernel. [Último acceso: 07 2025].

[15] L. Joel, W. Doorsamy y B. Paul, «On the Performance of Imputation Techniques for Missing Values on Healthcare Datasets,» 13 03 2024. [En línea]. Available: arXiv:2403.14687. [Último acceso: 08 2025].