

Article

Decision Tree Models and Machine Learning Algorithms in the Fault Recognition on Power Lines with Branches

Aleksandr Kulikov, Anton Loskutov ^{*}, Dmitriy Bezdushniy and Ilya Petrov

Department of Electric Power Engineering, Power Supply and Power Electronics, Nizhny Novgorod State Technical University n.a. R.E. Alekseev, Minin St., 24, 603115 Nizhny Novgorod, Russia; inventor61@mail.ru (A.K.); dimok.93@mail.ru (D.B.); ggosda@yandex.ru (I.P.)

* Correspondence: loskutov.nnov@gmail.com; Tel.: +7-908-158-6366

Abstract: The complication of the structure, topology and composition of the future electrical networks is characterized by difficult-to-recognize circuit-mode situations and requires modern methods for analyzing information parameters. The growing trend of digitizing signals in substations and the use of the IEC 61850 standard results in a huge amount of new data available at the nodes of the electrical network. The development and analysis of new methods for detecting and recognizing the modes of electrical networks (normal and emergency) are topical research issues. The article explores a new approach to recognizing a faulted section of an electrical network with branches by concurrently analyzing several information features and applying machine learning methods: decision tree, random forest, and gradient boosting. The application of this approach for decision-making by relay protection has not been previously implemented. Simulation modeling and the Monte Carlo method are at the heart of obtaining training samples. The results of testing the studied methods under review showed the required flexibility, the ability to use a large number of information parameters, as well as the best results of fault recognition in comparison with the distance protection relay.

Keywords: relay protection and automation (RPA); machine learning; simulation modeling; RPA algorithm; decision tree; random forest; gradient boosting; big data; short circuit recognition



Citation: Kulikov, A.; Loskutov, A.; Bezdushniy, D.; Petrov, I. Decision Tree Models and Machine Learning Algorithms in the Fault Recognition on Power Lines with Branches. *Energies* **2023**, *16*, 5563. <https://doi.org/10.3390/en16145563>

Academic Editors: Anna Pinnarelli and Mohsin Jamil

Received: 8 June 2023

Revised: 11 July 2023

Accepted: 20 July 2023

Published: 23 July 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Modern electric power systems are technically complexes consisting of a large number of interconnected elements. At the same time, both producers and consumers of electrical energy impose increasingly stringent requirements on the reliability of power supply. In the process of intellectualization of the electric power industry, the most important role is played by the improvement of relay protection and automation, the purpose of which is to disconnect faulted elements and sections of the electrical network. Modern relay protection and automation terminals, in addition to protection functions, perform the functions of control, registration, oscillography, etc., participating in data exchange with other devices located at the substation and beyond.

According to the results of the 49th Paris session of CIGRE, it is necessary to introduce and improve measures to improve the efficiency of the functioning of energy systems [1]. Moreover, this applies both to the operation of existing power systems, and promising ones, in which modern active elements can be installed (flexible power transmission lines, controlled shunt reactors, energy storage devices, etc.). Special attention is paid to mathematical modeling (simulation) of normal and emergency operating modes in the design, construction and reconstruction of electrical networks, as well as in the verification of engineering calculations and the choice of operation parameters for relay protection and automation devices (RPA) [2–5]. In addition, there is great interest in Big Data analysis technologies and artificial intelligence to improve the technical sophistication of electrical energy metering devices [6], automation and protection of electrical networks [7,8].

Reliable protection of power system elements from emergency modes, such as short circuits (SC), asynchronous operation, active power unbalance and others is the most important condition for its uninterrupted operation. In this regard, the development and analysis of methods for detecting and recognizing emergency modes is an urgent area for research. This becomes possible with the mass introduction of intelligent electronic devices (IEDs) of relay protection and automation into electrical networks that support the IEC 61850, which have direct access to a large amount of information about the protected object in real time [6–8]. Nevertheless, relay RPA algorithms have not fundamentally changed over the past decades and, in fact, are digital analogues of their electromechanical predecessors.

The complexity of future smart grids will require the use of modern methods of digital processing of current and voltage signals, systems for collecting information about current modes, identifying information parameters, as well as predictive models of the behavior of power systems in various modes. The problem of fault recognition in promising electrical systems, including those with distributed generation sources, will be even more difficult since transient processes occur very quickly. Also, there is a variability in the parameters of electrical signals at various generating installations and loads.

One of the promising directions for the qualitative development of RPA systems is the development of fundamentally new types of protection, acting in accordance with the operation rule, which is formed as a result of statistical processing of the results of simulation modeling of the protected object's operating modes. As a rule, these protections simultaneously control a large number of features (for example, currents, voltages, resistances, power, etc.) that are multi-parameter [2,4,5].

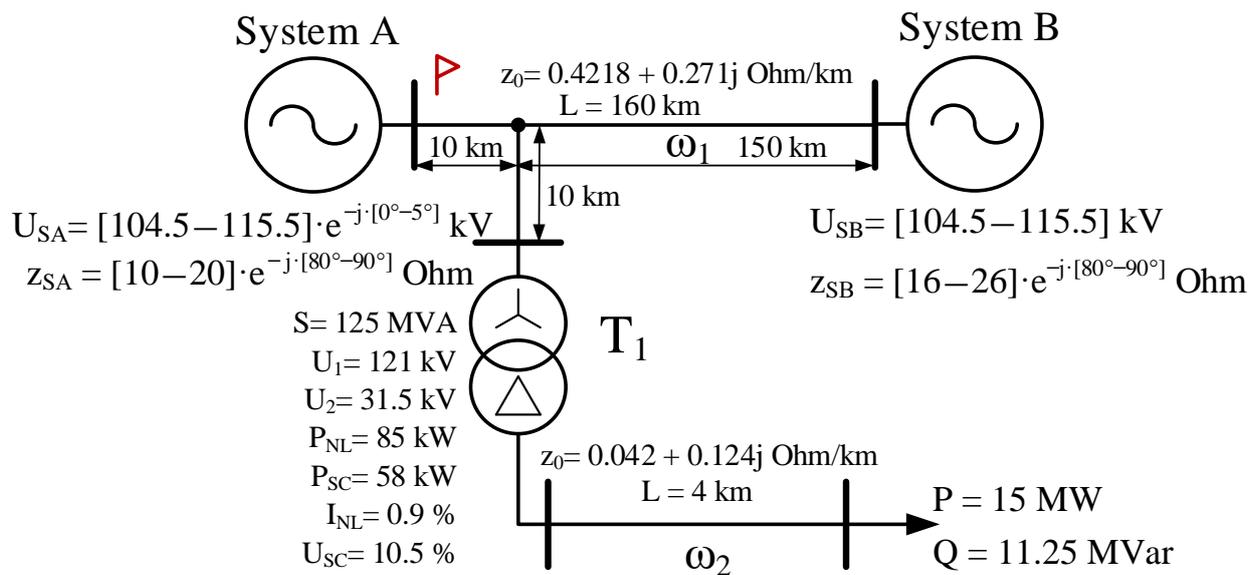
With the development of digitalization in the electric power industry, the recognition of emergency modes, in particular those associated with short circuits, can be implemented using machine learning algorithms [9–14]. These algorithms work in accordance with the triggering rule, which is formed after processing the results of model experiments (on simulation models). These multi-parameter algorithms simultaneously control a large number of characteristics or mode parameters (current, voltage, resistance, power, phase, etc.). Such an approach in relay protection and automation has become available since the computing power of modern processors has become sufficient to process the necessary amount of statistical data on the parameters of possible normal and emergency operation modes of electrical network sections. Such an approach may be most in demand in situations where, in order to ensure the selective operation of the RPA device, the protection algorithm must be able to distinguish short circuits in different sections of the electrical network which are characterized by close values of the mode parameters.

Previously, the authors conducted research on the use of machine learning methods to improve the functioning of relay protection and automation [3–5,15,16]. For example, the article [16] presents a classification of machine learning methods and examples of the application of some machine learning methods: k-nearest neighbor method; logistic regression method; support vector machine.

The aim of the article is to study an approach to fault recognition on power lines with branches by simultaneously analyzing several information features and applying such machine learning methods as the following: decision tree, random forest, and gradient boosting. Simulation and the Monte Carlo method are at the heart of obtaining training samples [17].

2. Simulation of an Electrical Network Section

Despite the introduction of digital technologies and measures to improve protection algorithms, there are still modes in electrical networks in which the typical use of relay protection and automation kits does not meet the requirements of sensitivity and selectivity. As a simplified example, let us consider the fragment of the electrical network shown in Figure 1.



▶ — installation location of the RPA device

Figure 1. Layout of a fragment of the electrical network (P_{NL} —no-load losses, P_{SC} —copper losses (short circuit loss), I_{NL} —no-load current, U_{SC} —short circuit voltage).

Let the protection of the line ω_1 be performed using a set of line differential protection (DPL). In order to ensure the blocking of the DPL in case of a short circuit behind the transformer T_1 , an additional permissive triggering element is required. As a rule, a distance relay is used as such a triggering element [18]. Its setting characteristic is chosen in such a way that the relay works with a SC on the line and does not work with a SC behind the transformer. Most often, the choice of a characteristic that meets the specified requirements is possible. However, short circuits on the line and behind the transformer become difficult to distinguish when the transformer has a sufficiently large, rated power (low through resistance) and the branch is located near the substation where the protection is installed.

For illustration, in accordance with the scheme (Figure 1), multiple simulation experiments of SC on the lines ω_1 and ω_2 were implemented in the MatLab software package. Two-phase and three-phase SCs were simulated at different distances from the protection installation site. In simulation the modes were divided into two categories: α -modes and β -modes [15,19]. α -modes are understood as SC and other emergency modes, in which the relay protection and protection devices should be triggered. β -modes are considered to be alternative modes from which protection must be adjusted.

The modeling methodology provides for multiple simulations of the operating modes of electrical networks in the MatLab program developed by the authors of the article [20]. This program is designed to generate data on the parameters of normal and emergency modes (complex values of currents and voltages) of electrical networks with a given configuration (for example, the configuration according to Figure 1). The network model can contain elements such as: voltage source, generating power, load, two-winding transformer, power line, etc. The result of the program is a set of .csv files inside a certain directory (/output). Similar results can be obtained using well-known simulation software packages MatLab/Simulink, PSCAD, RSCAD. Further, the results obtained for normal and emergency modes are processed in the Python language.

The developed MatLab program [20] passed the verification of the results of work with the Matlab/Simulink and PSCAD software packages and showed a discrepancy of less than 3%.

Figure 2 shows the measurements of the complex resistance (active and reactive) of the distance relay, with a short circuit at various points of the power transmission line ω_1 and behind the transformer T_1 (Figure 1). There is a significant degree of overlap of the mode parameters (Figure 2a). Figure 2b shows histograms of the effective (RMS) values of the currents flowing through the protection under the same conditions and types of SC.

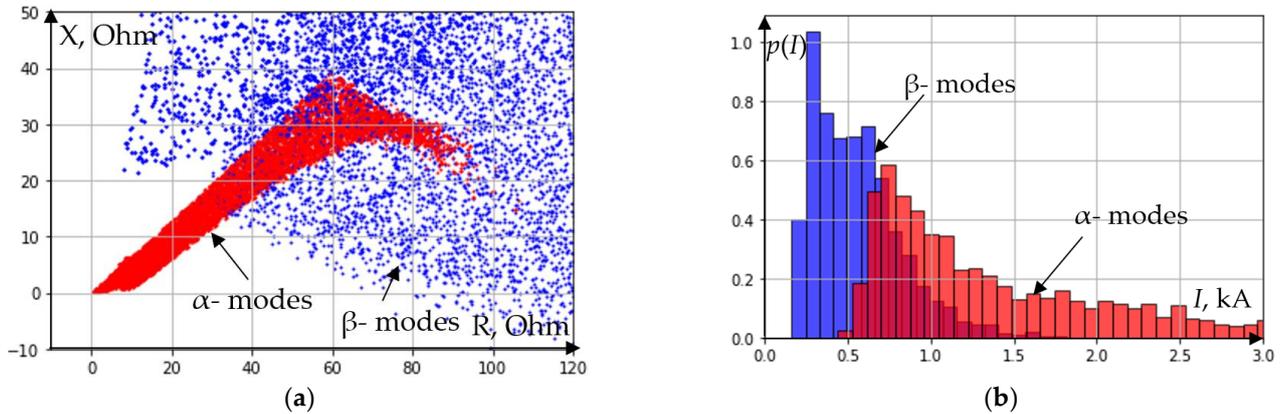


Figure 2. Results of simulation: (a) complex resistance (active and reactive) measurements; (b) histograms of the effective (RMS) value of currents.

It is obvious that it is not possible to select a triggering characteristic of a distance relay capable of ensuring reliable operation of the protection in case of a short circuit on the line and the absence of operation in the event of a short circuit behind the transformer. Any decision will involve a compromise between sensitivity and selectivity. Let us consider two boundary characteristics of operation of the distance relay (Figure 3a,b).

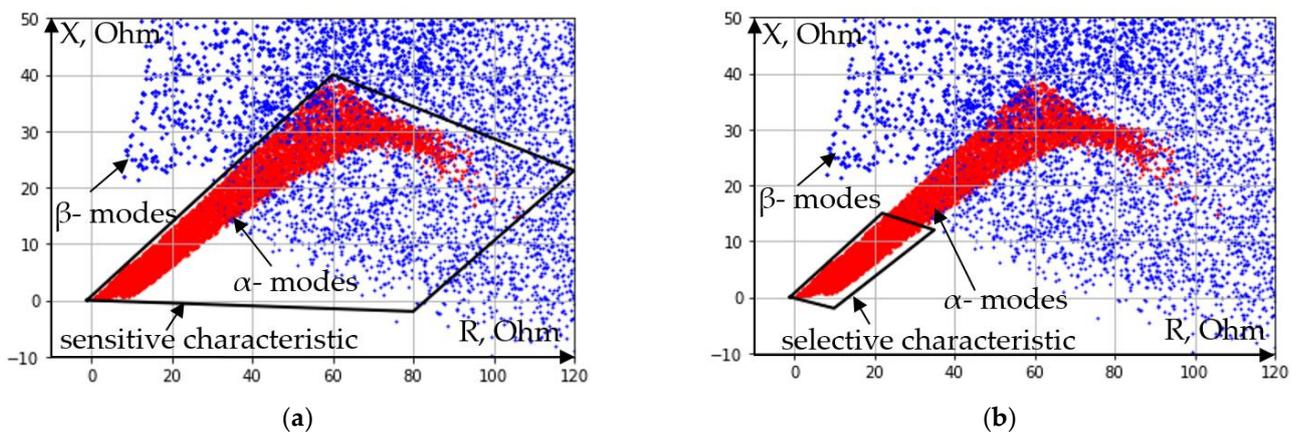


Figure 3. Boundary characteristics of remote relay operation: (a) sensitive; (b) selective.

The triggering area shown in Figure 3a is the most sensitive, since 100% SCs on the ω_1 power line fall into it. However, within this characteristic there is also 26.15% SC on the ω_2 power line located behind the transformer. The characteristic in Figure 3b is selective, does not allow false operation of the distance relay but provides only 35.25% SC on the power line ω_1 .

3. Statistical Efficiency Indicators of the Relay Protection Algorithm

To assess the reliability of the functioning of the RPA under operating conditions, an average statistical indicator h of its operation is used, called the percentage of correct actions [21]:

$$h = n_{c.t.} \times 100 / (n_{c.t.} + n_{t.f.} + n_{r.t.} + n_{f.t.}), \quad (1)$$

or

$$h = [n_{c.t.} - (n_{r.t.} + n_{f.t.})] \cdot 100 / (n_{c.t.} + n_{t.f.}), \quad (2)$$

where $n_{c.t.}$, $n_{r.t.}$, $n_{f.t.}$ —number of correct, redundant, and false triggering, respectively; $n_{t.f.}$ —number of triggering failures.

This indicator is statistical. It reflects the functioning of RPA devices under operating conditions and cannot be used to assess the recognition ability of protection algorithms according to simulation data (for example, Figure 2).

However, the use of a statistical approach, as well as methods that are widely used in machine learning problems (for example, [22–25]) and radio engineering applications (for example, [20]), allow us to introduce the required criteria for evaluating the effectiveness of RPA. It is assumed that the values of the variables $n_{c.t.}$, $n_{r.t.}$, $n_{t.f.}$ are obtained from simulation data.

Let us introduce the probabilistic indicators of recognition of the RPA algorithm under the condition of a SC (α -mode). Using the variables of expressions (1) and (2), such indicators include conditional probabilities:

- correct recognition of SC by the RPA algorithm

$$D = n_{c.t.} / (n_{c.t.} + n_{t.f.}); \quad (3)$$

- non-recognition of SC by the RPA algorithm

$$D' = n_{t.f.} / (n_{c.t.} + n_{t.f.}); \quad (4)$$

Since mutually exclusive solutions correspond to the same condition (presence of SC, α -mode), then

$$D + D' = 1. \quad (5)$$

Qualitative indicators of recognition by the RPA algorithm of the normal mode (no SC, β -mode) are conditional probabilities:

- false decision by the algorithm for triggering the RPA

$$F = n_{f.t.} / (n_{f.t.} + n_{c.f.}); \quad (6)$$

- the correct decision by the algorithm for the failure of the RPA

$$F' = n_{c.t.} / (n_{f.t.} + n_{c.f.}), \quad (7)$$

where $n_{c.f.}$ —the number of correct failures in the total number of simulation experiments, and

$$F + F' = 1. \quad (8)$$

It should be noted that in the calculations according to expressions (3)–(8) the indicator $n_{r.t.}$ is not used since it does not directly relate to the process of recognizing the studied circuit-mode situation, but it correlates with other modes of operation of the electrical network.

With such a formulation of the problem in the theory of machine learning and radio engineering applications, to assess the recognition ability of trained algorithms [19–23], the “receiver operating characteristic” (ROC-curve) is used [22]. The ROC-curve in relation to the RPA algorithm can be built by plotting on the plane along the ordinate axis the values D , and along the abscissa axis the variable F for various thresholds (settings) of protection operation (Figure 4). Since the value of $n_{c.t.}$ characterizes the sensitivity of protection in a series of simulation experiments, the ROC-curve clearly shows the balance between the sensitivity of the RPA device and its failure in acceptable (normal) modes at different settings.

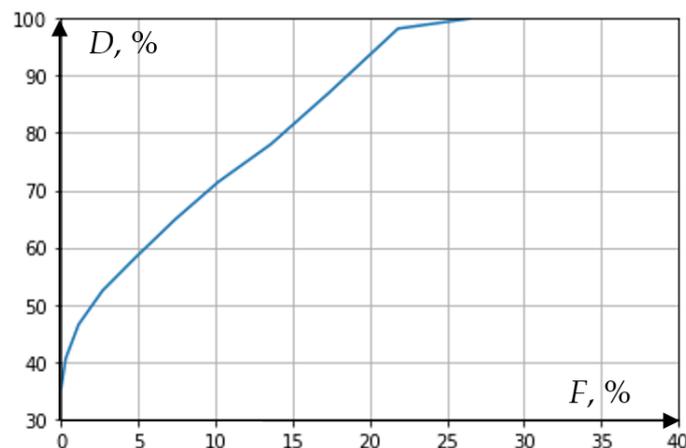


Figure 4. The ratio of correct and false triggering for different characteristics of the distance measuring element (ROC-curve).

Using the terminology of machine learning, we can say that the dependence (Figure 4) is a characteristic of the classifier, which is the distance relay [22,26,27], and gives a visual representation of the quality of the binary (relay) classification algorithm.

4. Application of the Decision Tree Algorithm for Recognizing the Modes of the Electrical Network

We use simulation and machine learning methods in the task of recognizing emergency modes, and we will choose the “decision tree” as the SC recognition algorithm. In this case, the computational procedure is a certain set of the simplest decision rules, united in a hierarchical structure (tree). An example of such a structure is shown in Figure 5.

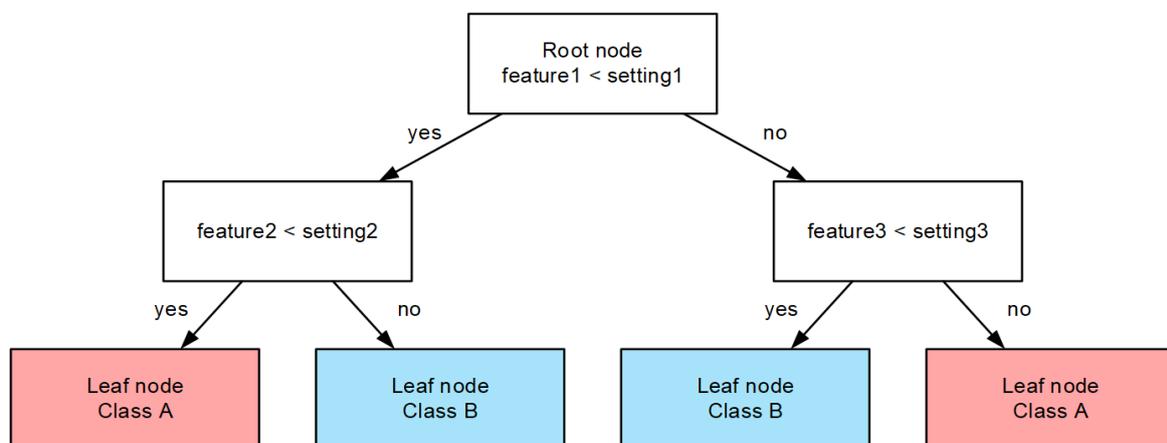


Figure 5. An example of the decision tree structure.

Each decision rule is a comparison of some information feature with a given response threshold (setting). The leaf nodes of the tree are associated with one of the possible classes. The task of recognizing the mode of an electrical network by its parameters using a trained decision tree requires movement from the root of the tree to one of its leaf nodes, along a trajectory that depends on the fulfillment or non-fulfillment of conditions at the nodes.

Learning the decision tree algorithm involves the following recursive procedure:

1. Finding an information feature and a trigger threshold for it, which will ensure the division of the training sample in such a way that the total heterogeneity of classes in the child nodes will decrease as much as possible compared to the heterogeneity in the initial node. The most commonly used measure of heterogeneity is:

- Entropy. The entropy calculation before and after splitting is performed according to expressions (9) and (10). The entropy difference is equal to the amount of information received from a given node of the tree [28]:

$$H(X) = - \sum_{i=1}^n p(x_i) \log_2 p(x_i). \quad (9)$$

$$H(X|Y) = - \sum_{i=1}^m p(y_i) \sum_{j=1}^n p(x_j|y_i) \log_2 p(x_j|y_i). \quad (10)$$

where $H(X)$ —the initial entropy of the node; $H(X|Y)$ —total entropy after splitting according to condition Y ; $p(x_i)$ —the proportion of objects in the training sample that have class i ; $p(y_0)$ —the proportion of objects in the training sample that do not satisfy the condition in the node; $p(y_1)$ —the proportion of objects in the training sample that satisfy the condition in the node; $p(x_i|y_0)$ —the proportion of objects that have class i among objects that do not satisfy the condition in the node; $p(x_i|y_1)$ —the proportion of objects that have class i among the objects that satisfy the condition in the node.

- Gini index [29,30]:

$$Gini(X) = 1 - \sum_{i=1}^n p(x_i). \quad (11)$$

$$Gini(X|Y) = \sum_{i=0}^1 p(y_i) \cdot \left(1 - \sum_{j=1}^n p(x_j|y_i) \right). \quad (12)$$

where $Gini(X)$ —the initial inhomogeneity of the node; $Gini(X|Y)$ —total inhomogeneity after splitting according to condition Y .

2. After dividing the training sample into subsets according to the selected criterion, each such subset is again divided in accordance with paragraph 1. The procedure is performed recursively until the nodes contain representatives of only one class or the specified maximum depth of the tree is reached. The maximum depth of a decision tree is the maximum possible length from root to leaf.

The features of the decision tree algorithm include:

1. Interpretability;
2. Ability to use a large number of information features;
3. No need to scale the data before training the algorithm;
4. The impossibility of changing the decision threshold of the trained model in order to increase its sensitivity/coarsen it.

Also, a feature of such a machine learning model as a decision tree is the absence of the need for preprocessing of the initial data, in contrast to, for example, logistic regression or the support vector machine [16], the use of which requires data normalization and elimination of correlation between various features. Therefore, as a training sample for the machine learning model, the real and imaginary parts of the complex values of the mode parameters obtained directly as a result of the simulation model were used.

4.1. Formation of Feature Space

As a result of each iteration of the simulation model, the complex values of the current and voltage of the three phases are formed at the installation site of the developed protection (Figure 1). Based on the received data, the mode parameters are calculated, which are used as information features in the process of learning the decision tree algorithm. The list of mode parameters, as well as their calculated expressions, are summarized in Table 1.

Table 1. Information features used for mode recognition.

No.	Feature	Formula
1	Active resistance. Phases AB	$R_{AB} = \text{Re}[(\underline{U}_A - \underline{U}_B)/(\underline{I}_A - \underline{I}_B)]$
2	Active resistance. Phases BC	$R_{BC} = \text{Re}[(\underline{U}_B - \underline{U}_C)/(\underline{I}_B - \underline{I}_C)]$
3	Active resistance. Phases CA	$R_{CA} = \text{Re}[(\underline{U}_C - \underline{U}_A)/(\underline{I}_C - \underline{I}_A)]$
4	Reactance. Phases AB	$X_{AB} = \text{Im}[(\underline{U}_A - \underline{U}_B)/(\underline{I}_A - \underline{I}_B)]$
5	Reactance. Phases BC	$X_{BC} = \text{Im}[(\underline{U}_B - \underline{U}_C)/(\underline{I}_B - \underline{I}_C)]$
6	Reactance. Phases CA	$X_{CA} = \text{Im}[(\underline{U}_C - \underline{U}_A)/(\underline{I}_C - \underline{I}_A)]$
7	Active power. Phase A	$P_A = \text{Re}(\underline{U}_A \cdot \underline{I}_A^*)$
8	Active power. Phase B	$P_B = \text{Re}(\underline{U}_B \cdot \underline{I}_B^*)$
9	Active power. Phase C	$P_C = \text{Re}(\underline{U}_C \cdot \underline{I}_C^*)$
10	Reactive power. Phase A	$Q_A = \text{Im}(\underline{U}_A \cdot \underline{I}_A^*)$
11	Reactive power. Phase B	$Q_B = \text{Im}(\underline{U}_B \cdot \underline{I}_B^*)$
12	Reactive power. Phase C	$Q_C = \text{Im}(\underline{U}_C \cdot \underline{I}_C^*)$
13	RMS current. Phase A	$I_{\text{rms}A} = (1/\sqrt{2}) \cdot \underline{I}_A $
14	RMS current. Phase B	$I_{\text{rms}B} = (1/\sqrt{2}) \cdot \underline{I}_B $
15	RMS current. Phase C	$I_{\text{rms}C} = (1/\sqrt{2}) \cdot \underline{I}_C $
16	RMS voltage. Phase A	$U_{\text{rms}A} = (1/\sqrt{2}) \cdot \underline{U}_A $
17	RMS voltage. Phase B	$U_{\text{rms}B} = (1/\sqrt{2}) \cdot \underline{U}_B $
18	RMS voltage. Phase C	$U_{\text{rms}C} = (1/\sqrt{2}) \cdot \underline{U}_C $
19	RMS positive sequence current	$I_{\text{rms}1} = (1/3) \cdot \underline{I}_A + \underline{I}_B \cdot e^{j120^\circ} + \underline{I}_C \cdot e^{j240^\circ} $
20	RMS negative sequence current	$I_{\text{rms}2} = (1/3) \cdot \underline{I}_A + \underline{I}_B \cdot e^{j240^\circ} + \underline{I}_C \cdot e^{j120^\circ} $
21	RMS positive sequence voltage	$U_{\text{rms}1} = (1/3) \cdot \underline{U}_A + \underline{U}_B \cdot e^{j120^\circ} + \underline{U}_C \cdot e^{j240^\circ} $
22	RMS negative sequence voltage	$U_{\text{rms}2} = (1/3) \cdot \underline{U}_A + \underline{U}_B \cdot e^{j240^\circ} + \underline{U}_C \cdot e^{j120^\circ} $

4.2. An Example of Training a Decision Tree Algorithm

Let us train and analyze the results of the decision tree algorithm. We set the requirements for the structure of the algorithm: the depth of the decision tree is 3, the minimum value of the number of sample values of the information feature that fall into the leaf node is 1. As a measure of heterogeneity, we use the Gini index.

We divide the set of simulated emergency and normal modes into training and test samples in the proportion of 67% to 33%. The training sample will be used to train the decision tree algorithm, and the test sample will be used to evaluate the quality of its functioning.

As a result of multiple simulation experiments aimed at training the protection algorithm, we obtain a decision tree with a structure (Figure 6).

At the root node (Figure 6), the decision tree implements the division of normal and emergency modes by the value of reactance, and the probabilistic distributions of this information feature are shown in Figure 7.

Although the probabilistic distributions of the information feature for α and β modes overlap each other (Figure 7), the setting value (threshold) makes it possible to separate the characteristic area of a short circuit on the ω_1 line from the characteristic area of a short circuit on ω_2 . As a result of such a division, only modes that satisfy the condition of node 0 get into node 1 of the decision tree. As can be seen from the analysis of Figures 6 and 7, modes in the proportions of 82% (α -mode) and 18% (β -mode) fall into node 1.

As a result of learning the decision tree algorithm, the feature that ensures the separation of modes at node 1 also turned out to be reactance, but this time X_{BC} . Figure 8 shows the probabilistic distributions of the information feature X_{BC} in node 1 and the setting (threshold) value of the feature.

The modes located to the left of the threshold value in Figure 8 fall into node 3, while the modes on the right—into node 4. Figure 8 shows that only β -modes fall into node 4; therefore, their further analysis does not make sense.

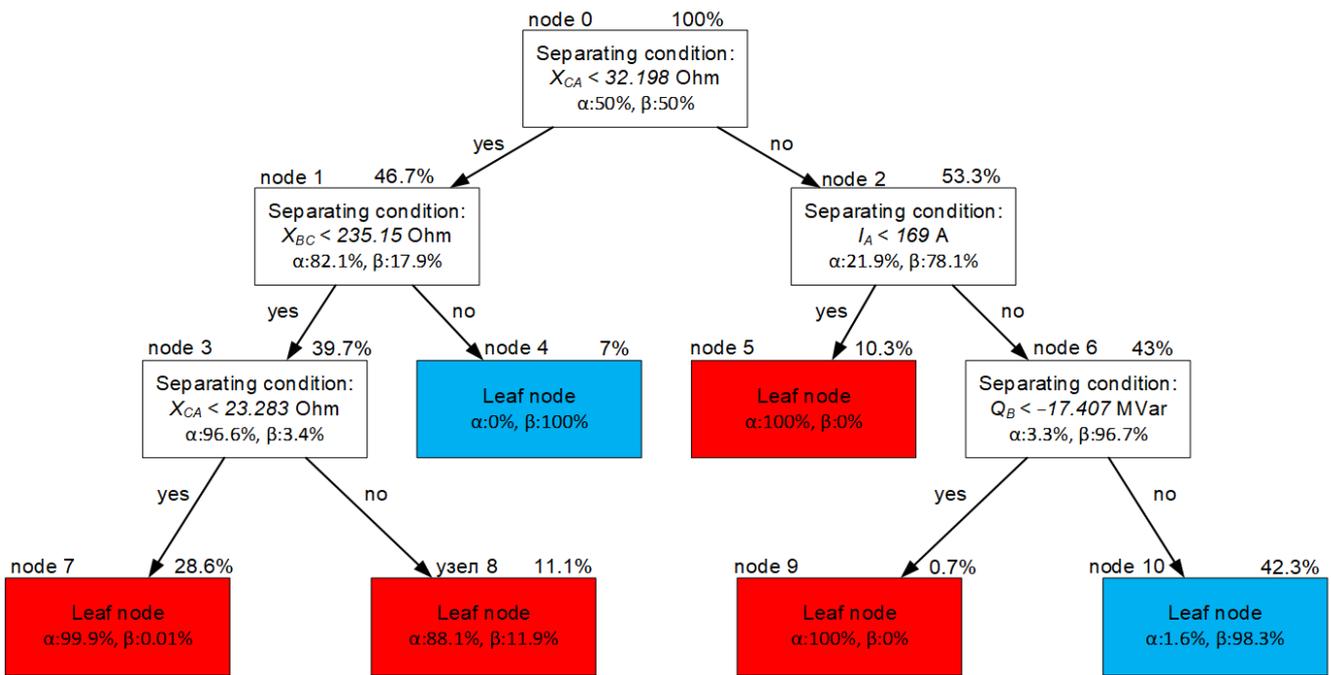


Figure 6. The decision tree structure.

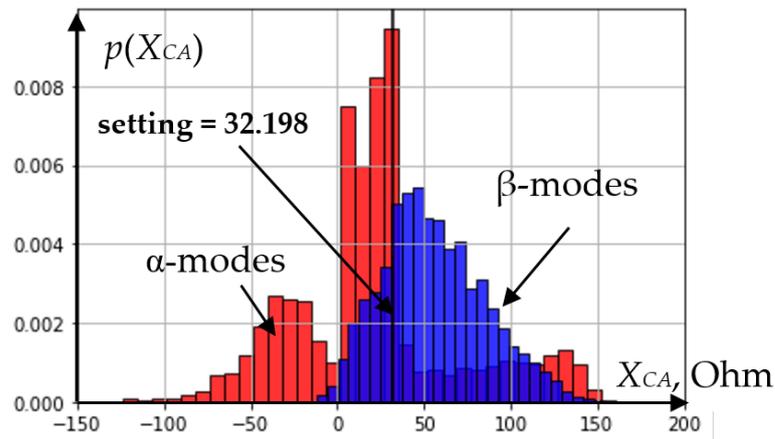


Figure 7. Probability distributions of feature X_{CA} in the root node of the decision tree.

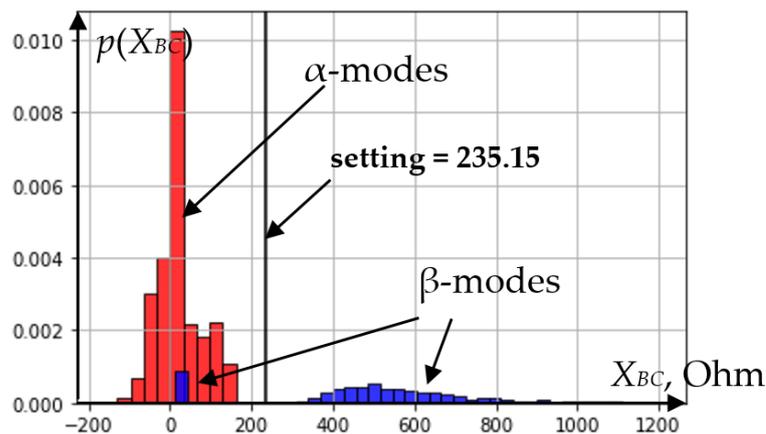


Figure 8. Probability distributions of feature X_{BC} in node 1 of the decision tree.

At the input of node 3, sample values typical for α - and β -modes are received in a percentage ratio of 96.6% to 3.4%. Here, according to the results of learning the decision tree algorithm, the most informative feature for separating modes is reactance X_{CA} again, the probability distributions of which for α - and β -modes are shown in Figure 9.

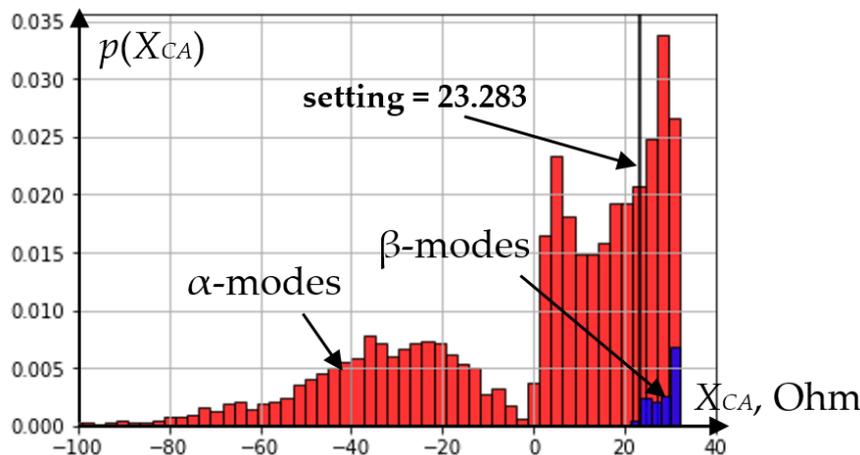


Figure 9. Probability distributions of feature X_{CA} in node 3 of the decision tree.

The setting (threshold) value for node 3 divided the modes in such a way that almost 100% of α -modes turned out to the left of it, and the modes turned out to be in the proportion of 88% to 12% to the right. Due to the maximum depth limit, the decision tree is not built further.

The analysis of the functioning of the decision tree algorithm for the modes located to the right of the threshold value of the root node 0 is implemented in a similar way. Probabilistic distributions of information features of nodes, as well as setting values are shown in Figure 10a,b.

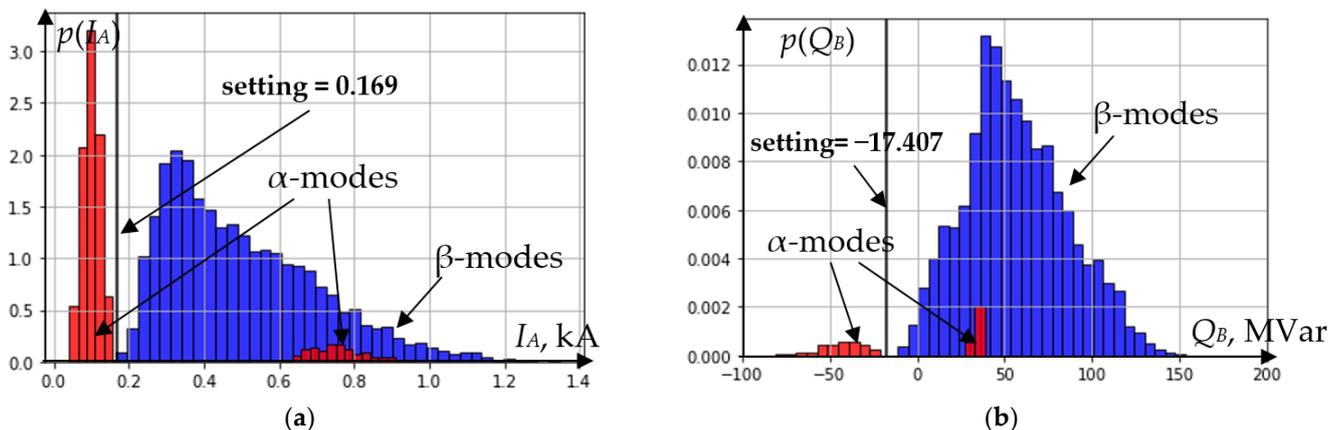


Figure 10. Probability distributions of features in tree nodes: (a) node 2; (b) node 6.

4.3. Discussion: Analysis of the Efficiency of the Decision Tree Algorithm

To estimate the performance of the trained decision tree algorithm, a test sample was used that did not participate in the learning process of the algorithm. The assignment of the tested mode to one of the two classes in the presence of a trained decision tree was performed from the root of the tree with progress along the branches. The required parameters of the classified mode (information features) are compared with the threshold values of the nodes. Once in a leaf node, a decision was made in favor of the class whose probability is the highest according to the training results.

When conducting simulation experiments with a test sample, an error matrix was obtained (Table 2), which characterizes the process of recognizing modes by the RPA algorithm, which includes a decision tree (Figure 6). In Table 2 SCs on the line ω_2 (Figure 1) are designated as class 0, while SCs on the line ω_1 —as class 1.

Table 2. Decision tree error matrix (Figure 6).

Class Characteristics		Predicted Class	
		0	1
True class	0	97.995	2.005
	1	1.292	98.708

As can be seen from Table 2, the decision tree algorithm in the RPA device (Figure 1) ensures the detection of 98.7% of SCs on the power line ω_1 , while the probability of false operation of the RPA device in case of a SC behind the transformer is about 2%.

Let us compare the result (Table 2) with the result of the distance relay by placing D and F of the decision tree and the previously obtained ROC-curve (Figure 4) of the distance measuring element on the same graph (point 1 in Figure 11).

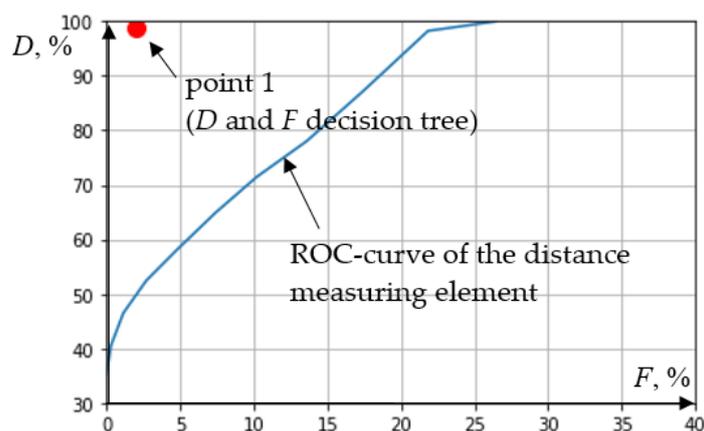


Figure 11. Correlation between the ROC-curve of the decision tree algorithm and the distance measuring element.

Analysis of Figure 11 clearly shows the advantage of the decision tree algorithm in comparison with the distance relay when recognizing emergency and normal modes by the RPA device (Figure 1). It should be noted that the disadvantage of the decision tree algorithm is the inability to optimize the setting values (Figure 6), thereby changing the sensitivity/selectivity ratio.

4.4. Structural Optimization of Decision Tree Algorithm Parameters

Above, we considered a decision tree algorithm with fixed, predetermined parameters: tree depth (*max depth*) and the minimum value of the number of objects in a leaf node (*min_samples_leaf*). However, in the practice of building recognition systems based on machine learning, the selection of parameters is usually performed iteratively in order to ensure the best quality of the classification algorithm. The simplest way to select the optimal parameters of the classification algorithm is the *Grid Search* method [29,30]. The method consists of sequential enumeration of all combinations of parameters from the given ranges. For each such combination, the quality of the classification algorithm is evaluated according to a pre-selected target metric, after which the best combination is selected. In this case, it is advisable to use the cross-validation technique, which consists of repeatedly reproducing the following sequence of steps:

1. Randomly, a part of the data used for training is divided into training and validation sets.
2. The classification algorithm is learning on the training set.
3. The target metric is estimated using the validation set.

The final result is defined as the average value of the target metric over all iterations.

During the simulation, the parameters of the classification algorithm were changed (Table 3), which were used to best separate the modes of the electrical network (Figure 1).

Table 3. Ranges of decision tree parameters used in the grid search method.

Parameter	Values
Decision Tree Depth	3, 5, 7, 9, 11, 13, 15
Number of objects in a leaf node	1, 3, 5, 10

The value M was used as the target metric, which characterizes the ratio of the number of correct recognitions and mode classification errors [30]. Taking into account the use of variable expressions (1)–(7), the target metric takes the form:

$$M = 2 \cdot [n_{c,t}/(n_{c,t} + n_{f,t})] \cdot [n_{c,t}/(n_{c,t} + n_{t,f})] / [n_{c,t}/(n_{c,t} + n_{f,t}) + n_{c,t}/(n_{c,t} + n_{t,f})]. \quad (13)$$

As a result of applying the grid search method, the parameters of the decision tree algorithm were found to provide the best value of the target metric: maximum tree depth = 9; the minimum number of objects in the leaf node = 3. This algorithm for classifying the modes of the electrical network corresponds to the error matrix (Table 4).

Table 4. Decision tree error matrix as a result of parameter optimization.

Class Characteristics	Predicted Class		
	0	1	
True class	0	98.619	1.381
	1	0.758	99.242

A comparative analysis of Tables 2 and 4 shows that the optimization of the parameters of the decision tree algorithm in relation to the RPA device (Figure 1) made it possible not only to increase the proportion of detected SCs on the line, but also to reduce the probability of false triggering of protection in case of a SC behind the transformer.

4.5. Structural Diagram of the Relay Protection Device That Implements the Decision Tree Algorithm

Figure 12 shows a block diagram of the relay protection device that implements decision making using the decision tree algorithm.

Before the RPA device starts functioning, the decision tree algorithm is subject to training on a sample of information parameters (for example, Table 1), which is obtained from the results of preliminary simulation modeling. During the simulation, all possible modes typical for the selected fragment of the electrical network are simulated. For each of the modes, which has its own identifier number, the learning results in the form of setting values of features, as well as options (parameters) for connecting informational features to nodal comparison circuits, are stored in the memory block.

During the operation of the RPA device, its input continuously receives instantaneous values (complexes) of currents and voltages. In the block for estimating the mode parameters, the parameters (information features) involved in the implementation of the decision tree algorithm are calculated (for example, in Table 1). In accordance with the current mode of the electrical network, a control signal (for example, in the form of a mode number) is supplied to the input of the memory block, which ensures the output

of information from the memory block to the input of the block for estimating the mode parameters about the switching parameters for transferring information features to the required comparison circuits ((CC) (the nodes of the decisive tree)). Additionally, according to the control signal at the input of the memory block, the setting values are output to the comparison circuits to implement the decision-making by the decision tree algorithm (RPA device). The connection of comparison circuits is carried out in accordance with the chosen structure of the decision tree algorithm (for example, Figure 6).

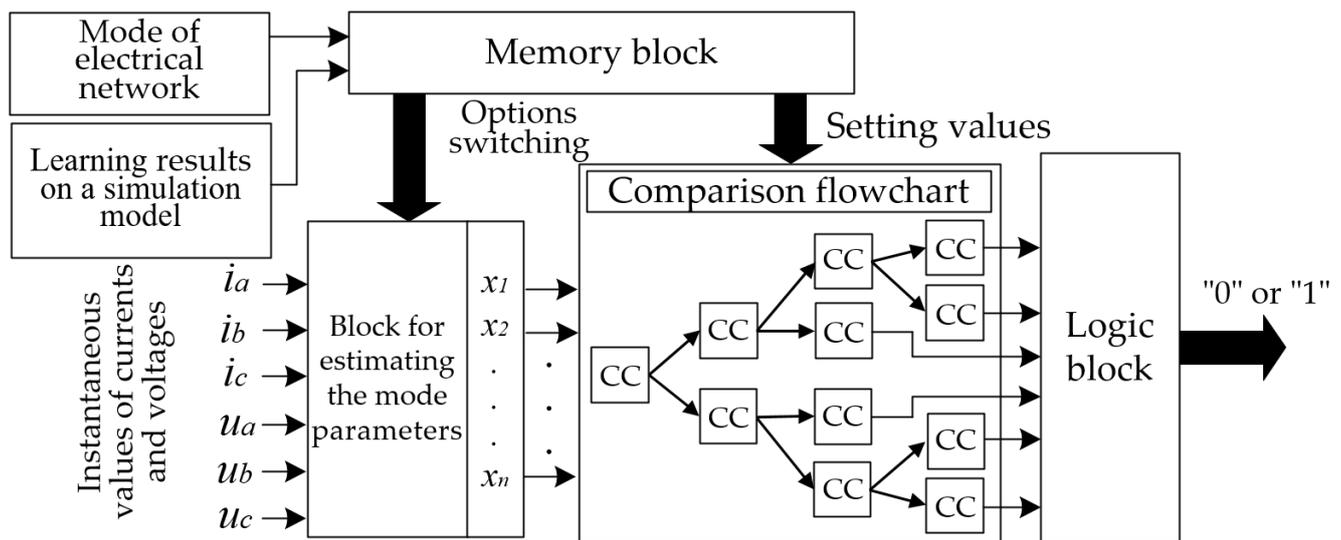


Figure 12. Structural diagram of the relay protection device that implements the decision tree algorithm.

At the outputs of the comparison circuits corresponding to the leaf nodes of the decision tree algorithm, signals are generated, according to which a decision is made regarding the recognizable mode (α - or β -mode) and which are fed to the logic block. In the logic block, the final decision regarding the recognized mode is formed (for example, by the majority principle) and is fed to the output of the RPA device.

It should be noted that the implementation of the decision tree algorithm, for example, for RPA microprocessor devices will not require hardware upgrades, but is associated with the refinement of special software.

5. Recognition of Electrical Network Modes Using Group Machine Learning Algorithms

An effective way to increase the probability of correct classification of electrical network modes is the joint use of several machine learning algorithms. This effect is especially evident when the group algorithm includes algorithms that differ significantly in the principles of operation. When using many different computational procedures, situations are possible when their calculation errors cancel each other out.

To form group algorithms for classifying the modes of an electric network, the application of two machine learning algorithms is analyzed below: random forest and gradient boosting [29–32]. In both cases, the decision trees considered above are used as the basic algorithms that are part of the group algorithms.

5.1. Random Forest

Random forest [29,30] is a group machine learning algorithm consisting of a certain number of decision trees. Each of them is trained independently on different subsets of the input data. When classifying the modes of an electrical network, the probability of each class is determined as the proportion of decision trees included in the group algorithm that point to a given class.

The main parameters of the random forest algorithm are:

- the number of decision trees in the group algorithm ($n_{estimators}$);
- the amount of the training sample that will be used to train each tree ($max_samples$). In this case, the formation of a subsample for each decision tree is implemented randomly according to a scheme with repetitions;
- the number of information features of the electrical network modes which will be used to train each tree ($max_features$);
- parameters characterizing the decision trees included in the group algorithm.

Let us determine the parameters of the random forest group algorithm in relation to the problem of classifying the modes of the electrical network (Figure 1). Let us take the maximum number of information features ($max_features$) in the group algorithm equal to the total number of information features (Table 1), and the maximum number of sample samples ($max_samples$)—equal to the number of samples in the training sample. With this scheme, due to sampling with repetitions, an average of 63% of previously unused values from the training sample will be used to train each tree. The number of separate algorithms ($n_{estimators}$) will be determined as a result of selecting the parameters of the group algorithm.

As in the case of training the decision tree algorithm, we will use the grid search method to determine the parameters of the group random forest algorithm that provide the maximum probabilities of correctly recognizing the modes of the electrical network. The list of changeable parameters is given in Table 5.

Table 5. Parameter values of the group random forest algorithm used in the implementation of the grid search method.

Parameter	List of Values
Maximum decision tree depth	3, 5, 7, 9, 11, 13, 15
Minimum number of objects in a leaf node	1, 3, 5, 10, 15
Number of trees in the group algorithm	50, 100, 150

Unlike decision trees, the group random forest algorithm allows you to build the ROC-curve (Figure 13), and the area under the ROC-curve can be used as a metric when selecting the parameters of the group classification algorithm.

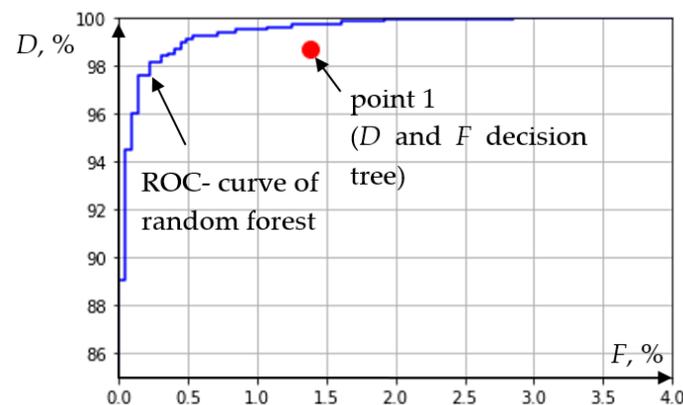


Figure 13. Recognition indicators of decision tree and random forest algorithms.

As a result of the implementation of the grid search method, the parameters of the group random forest algorithm were found, which provide the best recognition of the modes of the electric network (Figure 1): maximum depth of the decision tree = 9; minimum number of objects in a leaf node = 1; number of trees in the group algorithm = 100.

The error matrix for the trained random forest group algorithm with a threshold (setting) value of 0.5 corresponds to Table 6.

Table 6. Error matrix of group random forest algorithm.

Class Characteristics		Predicted Class	
		0	1
True class	0	99.153	0.847
	1	0.535	99.465

Figure 13 shows the ROC-curve of the group random forest algorithm, as well as the values of D and F , which are specific to the decision tree algorithm.

As can be seen from Figure 13 and analysis Table 6, the use of a group random forest algorithm reduces the probability of errors in the classification of electrical network modes compared to a single decision tree algorithm, and also makes it possible to adjust the setting (threshold) of operation to increase the sensitivity or coarsen the relay protection. Since the obtained group random forest algorithm includes 100 single decision tree algorithms, it becomes obvious that the increase in the recognition ability of the relay protection device is achieved by its significant complication.

5.2. Gradient Boosting

Boosting is an alternative way to form group algorithms for classifying electric network modes [29,31,32]. The idea of boosting is that each subsequent separate recognition algorithm included in the group is trained in such a way as to compensate for the errors of all previous ones. Then implementing the procedure for classifying the electrical mode using a group algorithm, it is necessary to add up the solutions of all algorithms included in the group.

When applying gradient boosting, the basic algorithms are regression algorithms. Modifications of decision trees for the regression problem (*Regressor decision tree*) are most often used [29]. However, regression algorithms may have different values of output parameters and the sum of their final solutions may range from $-\infty$ to $+\infty$. In order to obtain the probabilities of classes lying in the range $[0, 1]$, a limiting function is used, usually a sigmoidal one (Figure 14), corresponding to the expression:

$$\text{sigm}(x) = \frac{1}{1 + e^{-x}}. \quad (14)$$

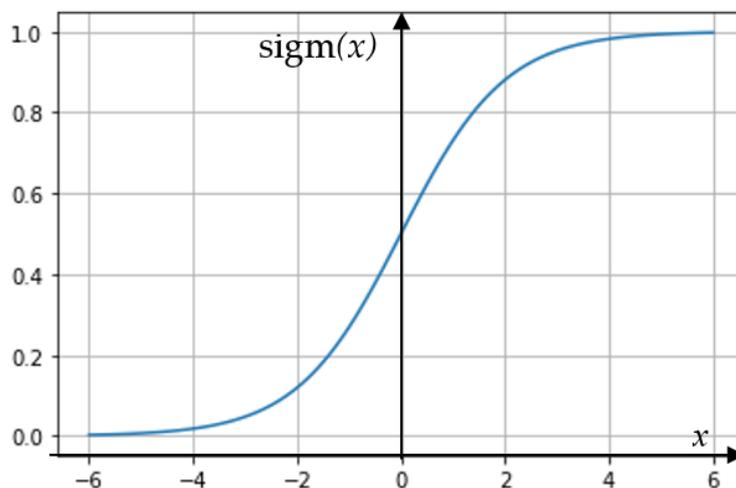


Figure 14. Graph of the sigmoid function.

When training a group classification algorithm based on gradient boosting, a loss function is set, depending on the ratio of decisions made by the group algorithm and the true values of the classes. During the training procedure of the group gradient boosting algorithm, the loss function is minimized, which is chosen as the log loss(x) dependence corresponding to the equality:

$$\log \text{loss}(x) = - \sum_{i=0}^{N-1} y_i \cdot \log(p_i) + (1 - y_i) \cdot (1 - p_i). \quad (15)$$

where p_i —the probability predicted by the model that the group algorithm will attribute the i -th mode of the electric network to class “1”; y_i —true class of the i -th object (0 or 1); N —the sample size.

The training of the group gradient boosting algorithm is also conducted using an iterative procedure and the grid search method [29]. At each iteration, the value of the loss function (15) and its gradient are calculated. Additionally, the *learning rate* parameter is introduced. When using an adjustable learning rate coefficient, a positive effect is achieved, aimed at preventing any skipping of the local minimum of the loss function.

In addition to the learning rate coefficient, the main parameters of the gradient boosting group algorithm include: the number of trees in the group algorithm ($n_estimators$), as well as parameters describing each decision tree separately.

The values of the parameters of the group gradient boosting algorithm, which were used in training using the grid search method, are given in Table 7.

Table 7. Ranges of gradient boosting parameters.

Parameter	List of Values
Maximum decision tree depth	3, 5, 7, 9, 11, 13, 15
Minimum number of objects in a leaf node	1, 3, 5, 10, 15
Number of trees in the group algorithm	50, 100, 150
Learning rate coefficient	0.01, 0.05, 0.1

As a result of the optimization selection of gradient boosting parameters with the target metric, which was the area under the performance curve, the following values were obtained: the maximum depth of the decision tree = 7, the minimum number of objects in a leaf node = 15, the number of trees in the group algorithm = 100, learning rate coefficient = 0.1.

The error matrix formed as a result of testing the group gradient boosting algorithm corresponds to Table 8.

Table 8. Error matrix of the gradient boosting group algorithm.

Class Characteristics	Predicted Class		
	0	1	
True class	0	98.797	1.203
	1	0.49	99.51

5.3. Discussion

Figure 15 shows the ROC-curves of group algorithms for gradient boosting and random forest.

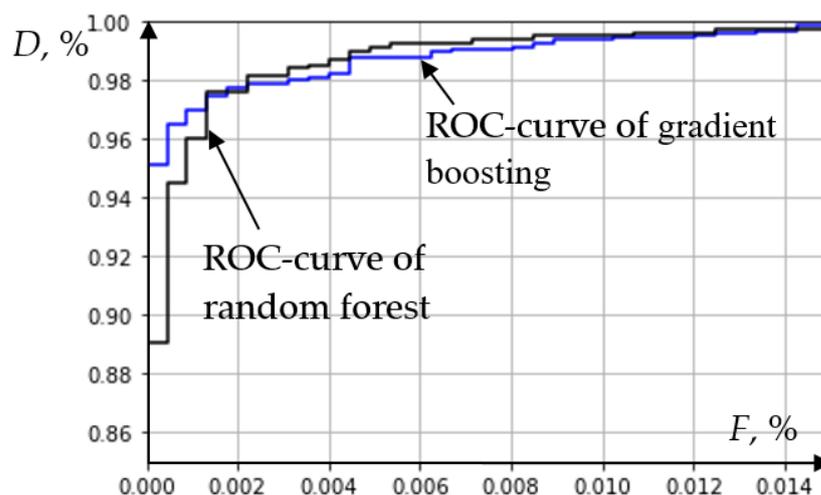


Figure 15. Recognition indicators of group algorithms for gradient boosting and random forest.

An analysis of the characteristics (Figure 15) of group algorithms for gradient boosting and random forest allows us to conclude that they are comparable in efficiency. Gradient boosting is somewhat more sensitive at low values of F . However, with an acceptable false positive probability of more than 0.2% more, random forest is preferable. Due to the closeness of the ROC-curves (Figure 15), it can be assumed that the combined use of various group machine learning algorithms makes it possible to obtain the limiting values of the parameters characterizing the potential capabilities of the relay protection device for recognizing the modes of the electrical network.

It is important to note that despite the high recognizing ability of group algorithms for gradient boosting and random forest, their use in relay protection devices is very doubtful. This is due to the fact that the simultaneous implementation of hundreds of separate decision tree algorithms during the response time of the RPA (tens of milliseconds) will require extremely high performance and cost of the corresponding devices. However, for solving problems of recognizing electrical network modes that do not require high performance, the use of group machine learning algorithms is undoubtedly promising.

6. Conclusions

1. When organizing RPA on overhead power lines with branches, the required sensitivity of protection in case of short circuit behind high-power transformers is not always provided. It is required to develop new relay protection algorithms for recognizing difficult-to-distinguish modes.

2. Multiple simulation modeling based on the Monte Carlo method allows not only for obtaining the distributions of current and voltage parameters in various modes of operation of the electrical network, but also for creating the basis for the synthesis of statistical relay protection algorithms based on machine learning methods.

3. The use of the decision tree method for the organization of relay protection and protection on power lines with branches made it possible to form a multi-parameter protection algorithm, which is characterized by increased sensitivity in case of short circuit on a branch behind the transformer compared to a distance triggering protection element.

4. The combined use of group algorithms of gradient boosting and random forest based on machine learning makes it possible to obtain the limiting values of the parameters characterizing the potential capabilities of the relay protection device for recognizing the modes of the electrical network. However, their hardware implementation is very doubtful due to the very high-performance requirements of the computing components of the RPA devices.

Author Contributions: Conceptualization, A.K., A.L. and D.B.; methodology, A.K.; software, D.B. and I.P.; validation, A.L. and D.B.; formal analysis, A.L.; investigation, A.L., D.B. and I.P.; resources, D.B. and I.P.; data curation, D.B.; writing—original draft preparation, A.L. and I.P.; writing—review and editing, A.K. and A.L.; visualization, D.B. and I.P.; supervision, A.K.; project administration, A.K. and A.L.; funding acquisition, A.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Ministry of Science and Higher Education of the Russian Federation (state task No. FSWE-2022-0005).

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Mcguinness, S.; Bi, T.; Kreutzer, P. *Special Report for SC B5. Protection and Automation*; CIGRE: Paris, France, 2022.
2. Sharygin, M.V.; Kulikov, A.L. Statistical Methods of Mode Recognition in Relay Protection and Automation of Power Supply Networks. *Power Technol. Eng.* **2018**, *52*, 235–241. [[CrossRef](#)]
3. Loskutov, A.A.; Pelevin, P.S.; Vukolov, V.Y. Improving the recognition of operating modes in intelligent electrical networks based on machine learning methods. In Proceedings of the E3S Web of Conferences, Kazan, Russia, 21–25 September 2020; Volume 216, p. 1034. [[CrossRef](#)]
4. Loskutov, A.A.; Pelevin, P.S.; Mitrovic, M. Development of the logical part of the intellectual multi-parameter relay protection. In Proceedings of the E3S Web of Conferences, Tashkent, Uzbekistan, 23–27 September 2019; Volume 139, p. 1060. [[CrossRef](#)]
5. Kulikov, A.; Loskutov, A.; Sovina, A. The Use of Machine Learning and Artificial Neural Networks to Recognition of Turning Faults in Power Transformers. In Proceedings of the 49th CIGRE Session, Paris, France, 29 August–2 September 2022; p. 11162.
6. Ribeiro, P.F.; Duque, C.A.; Ribeiro, P.M.; Cerqueira, S.C. *Power Systems Signal Processing for Smart Grids*; Wiley: New York, NY, USA, 2013; 448p.
7. Rebizant, W.; Szafran, J.; Wiszniewski, A. *Digital Signal Processing in Power System Protection and Control*; Springer: London, UK, 2011; 316p.
8. Ghiasi, M.; Wang, Z.; Mehrandezh, M.; Alhelou, H.H.; Ghadimi, N. A New Fast Bus Tripping System Design of Protection Relay in an AC Power Network. In Proceedings of the GlobConET 2023, London, UK, 19–21 May 2023. [[CrossRef](#)]
9. Hasan, A.N.; Pouabe, P.S.; Twala, B. The Use of Machine Learning Techniques to Classify Power Transmission Line Fault Types and Locations. In Proceedings of the International Conference on Optimization of Electrical and Electronic Equipment, Fundata, Romania, 21–24 May 2017; pp. 221–226. [[CrossRef](#)]
10. Witten, I.H.; Frank, E. *Data Mining: Practical Machine Learning Tools and Techniques*, 2nd ed.; Elsevier: Amsterdam, The Netherlands, 2005; 525p.
11. Michie, D.; Spiegelhalter, D.; Taylor, C. *Machine Learning, Neural and Statistical Classification*; Ellis Horwood: Chichester, UK, 1994; 290p.
12. Bishop, C.M. *Pattern Recognition and Machine Learning*; Springer: Berlin, Germany, 2006; 738p.
13. Leo, B. *Classification and Regression Trees*, 1st ed.; Routledge: New York, NY, USA, 1984; 368p.
14. Yongli, Z.; Limin, H.; Jinling, L. Bayesian Networks-Based Approach for Power Systems Fault Diagnosis. In *IEEE Transactions on Power Delivery*; IEEE: Piscataway, NJ, USA, 2006; No. 21; pp. 636–669. [[CrossRef](#)]
15. Kulikov, A.; Ilyushin, P.; Loskutov, A. Enhanced Readability of Electrical Network Complex Emergency Modes Provided by Data Compression Methods. *Information* **2023**, *14*, 230. [[CrossRef](#)]
16. Kulikov, A.; Loskutov, A.; Bezdushniy, D. Relay Protection and Automation Algorithms of Electrical Networks Based on Simulation and Machine Learning Methods. *Energies* **2022**, *15*, 6525. [[CrossRef](#)]
17. Abud, T.P.; Augusto, A.A.; Fortes, M.Z.; Maciel, R.S.; Borba, B.S.M.C. State of the Art Monte Carlo Method Applied to Power System Analysis with Distributed Generation. *Energies* **2023**, *16*, 394. [[CrossRef](#)]
18. Vakili, R.; Khorsand, M. A Machine Learning-Based Method for Identifying Critical Distance Relays for Transient Stability Studies. *Energies* **2022**, *15*, 8841. [[CrossRef](#)]
19. Qawaqzeh, M.Z.; Miroshnyk, O.; Shchur, T.; Kasner, R.; Idzikowski, A.; Kruszelnicka, W.; Tomporowski, A.; Bałdowska-Witos, P.; Flizikowski, J.; Zawada, M.; et al. Research of Emergency Modes of Wind Power Plants Using Computer Simulation. *Energies* **2021**, *14*, 4780. [[CrossRef](#)]
20. Computer Program “Program for Multiple Simulation Modeling of Operating Modes of Electrical Networks”/Loskutov, A.A., Bezdushny, D.I.; Certificate of State Registration No. 2022666666; dec. No. 2022665769 of 08/26/2022; publ. 09/06/2022.—Bull. No. 9. Available online: <https://www.fips.ru/iiss/document.xhtml?faces-redirect=true&id=57a7e1ed5b1caab9dcd4fce11a9eb130> (accessed on 19 July 2023).
21. Wang, K.; Li, W.Q. Research on Reliability of Relaying Protection in Smart Substation. *World J. Eng. Technol.* **2019**, *7*, 333–338. [[CrossRef](#)]
22. Fawcett, T. An introduction to ROC analysis. *Pattern Recognit. Lett.* **2006**, *27*, 861–874. [[CrossRef](#)]

23. Pendrill, L.R.; Melin, J.; Stavelin, A.; Nordin, G. Modernising Receiver Operating Characteristic (ROC) Curves. *Algorithms* **2023**, *16*, 253. [[CrossRef](#)]
24. Ciocan, A.; Hajjar, N.A.; Graur, F.; Oprea, V.C.; Ciocan, R.A.; Bolboaca, S.D. Receiver Operating Characteristic Prediction for Classification: Performances in Cross-Validation by Example. *Mathematics* **2020**, *8*, 1741. [[CrossRef](#)]
25. Hastie, T.; Tibshirani, R.; Friedman, J. *The Elements of Statistical Learning*, 2nd ed.; Springer: Berlin/Heidelberg, Germany, 2001; 745p.
26. Qi, W.; Swift, G.; McLaren, P. Distance protection using an artificial neural network. In Proceedings of the Sixth International Conference on Developments in Power System Protection, Nottingham, UK, 25–27 March 1997; No. 434, pp. 286–290.
27. Hutter, F.; Kotthoff, L.; Vanschoren, J. *Automated Machine Learning*; Springer: Cham, Switzerland, 2019; 219p.
28. Xia, Y.Q.; Li, K.K.; David, A.K. Adaptive Relay Setting for Stand-Alone Digital Distance Protection. *IEEE Trans. Power Deliv.* **1994**, *9*, 480–491. [[CrossRef](#)]
29. Guido, S.; Mueller, A. *Introduction to Machine Learning with Python*; O'Reilly: Sebastopol, CA, USA, 2016; 338p.
30. Zhang, C.; Wang, W.; Liu, L.; Ren, J.; Wang, L. Three-Branch Random Forest Intrusion Detection Model. *Mathematics* **2022**, *10*, 4460. [[CrossRef](#)]
31. Zhao, L.; Lee, S.; Jeong, S.-P. Decision Tree Application to Classification Problems with Boosting Algorithm. *Electronics* **2021**, *10*, 1903. [[CrossRef](#)]
32. Di Persio, L.; Fraccarolo, N. Energy Consumption Forecasts by Gradient Boosting Regression Trees. *Mathematics* **2023**, *11*, 1068. [[CrossRef](#)]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.