Article

Power Line Monitoring through Data Integrity Analysis with Q-Learning Based Data Analysis Network

Rytis Maskeliūnas [1,*], Raimondas Pomarnacki [2], Van Khang Huynh [3], Robertas Damaševičius [4] and Darius Plonis [2]

1   Center of Excellence Forest 4.0, Department of Multimedia Engineering, Kaunas University of Technology, 51423 Kaunas, Lithuania
2   Department of Electronic Systems, Faculty of Electronics, Vilnius Gediminas Technical University, Sauletekio Ave. 11, 10223 Vilnius, Lithuania
3   Department of Engineering Sciences, University of Agder, Postboks 422, 4604 Kristiansand, Norway
4   Department of Applied Mathematics, Silesian University of Technology, 44-100 Gliwice, Poland
*   Correspondence: rytis.maskeliunas@ktu.lt

Abstract: To monitor and handle big data obtained from electrical, electronic, electro-mechanical, and other equipment linked to the power grid effectively and efficiently, it is important to monitor them continually to gather information on power line integrity. We propose that data transmission analysis and data collection from tools like digital power meters may be used to undertake predictive maintenance on power lines without the need for specialized hardware like power line modems and synthetic data streams. Neural network models such as deep learning may be used for power line integrity analysis systems effectively, safely, and reliably. We adopt Q-learning based data analysis network for analyzing and monitoring power line integrity. The results of experiments performed over 32 km long power line under different scenarios are presented. The proposed framework may be useful for monitoring traditional power lines as well as alternative energy source parks and large users like industries. We discovered that the quantity of data transferred changes based on the problem and the size of the planned data packet. When all phases were absent from all meters, we noted a significant decrease in the amount of data collected from the power line of interest. This implies that there is a power outage during the monitoring. When even one phase is reconnected, we only obtain a portion of the information and a solution to interpret this was necessary. Our Q-network was able to identify and classify simulated 190 entire power outages and 700 single phase outages. The mean square error (MSE) did not exceed 0.10% of the total number of instances, and the MSE of the smart meters for a complete disturbance was only 0.20%, resulting in an average number of conceivable cases of errors and disturbances of 0.12% for the whole operation.

Keywords: data integrity analysis; artificial neural network; Q-learning; power line; monitoring

1. Introduction

With the increasing expansion of society's need for energy in recent years, the power supply load pressure of the power system has also increased significantly. Transmission lines provide extensive coverage as a critical piece of transmission infrastructure. Air pollution, temperature conditions, and other environmental elements have a significant impact on high-voltage transmission lines. Conductor motion, insulation pollutant flashover, conductor freezing, wind skewed partial discharge, and other events occur on occasion, frequently resulting in arcing, material and insulation damage, conductor burning, disconnecting, tower collapsing, and other problems [1]. It generates significant costs and jeopardizes the safe functioning of high-voltage power lines. The omnipresent monitoring and control intended for objectives such as frequency regulation, demand response, asset management, and anomaly detection becomes a distinguishing and necessary feature of power grids [2]. Identifying possible concerns and anticipating defects in working cables

in a non-destructive way is an appropriate method for avoiding in-service failures [3]. It is also becoming more common to monitor and explore the integrity of power sources and manage distant systems, such as solar power plants [4,5], wind power plants [6–8], residential buildings [9], and smart homes [10,11]. On the other hand, there are an increasing number of automated production systems and factories, which necessitate comparable procedures [12–14]. One significant disadvantage of present cable diagnostics systems is the need for specialized equipment to conduct the tests [15]. This necessitates the development of remote data collecting and monitoring solutions. However, most of the time, such platform performance is compared to some type of distributed network or presently widespread Internet of Things (IoT) networks, where evaluation focuses on simplified data transfer capabilities [16]. For the smart power line management implementation process, the IoT edge agent device may perform the functions of a virtual switch, edge computing, interface adaptation, and protocol analysis, as well as adopting terminal security authentication techniques and data encryption designs [17], potentially benefiting from one of the major aspects in the deployment of software defined networking (SDN) technology in communications infrastructure—that is the balancing among real-time and flexibility [18]. As time granularity varies, active measuring methods in a ring network architecture were investigated, while specific measurement intervals may be attributable to changes in measurement frequencies, they discovered that the active approach was comparable with the Polling measuring method for evaluating the very same data flow trend [19].

One of the primary causes of power grid failures is the collapse of overhead power line towers, which expose electrical providers to significant, high-value fines. The proposed architecture for remote monitoring of mechanical stresses in guyed towers was suggestd in [20] work. The method includes a mesh network for data forwarding and artificial neural networks to enhance the performance of Low-Power and Lossy Networks, as well as the sensor fusion methodology, which takes into account the utilization of many sensors. The breakdown of power line insulators leads to the another common failure of power transmission systems; another frequently used approach is an insulator inspection system based on an aerial platform. Paper [21] examines the automated identification of insulator faults utilizing aerial photos, as well as properly localizing insulator problems occurring in input photographs acquired from real-world inspection situations. The authors present a unique deep convolutional neural network cascade architecture based on a region proposal network to turn defect inspection into a two-level object identification issue for performing localization and identifying flaws in insulators. Power grid frequency rises during periods of overproduction and falls during periods of underproduction. Deep learning algorithms can be used to anticipate grid stability eliminating constraining assumptions about input values [22]. Erdem et al. [23] propose utilizing Layer-Wise Relevance Propagation to determine the relevance scores of each input and make the system more understandable.

Solving the issues of inadequate transmission capacity at rush hour and power transmission breakdown is a big task. Conventional manual inspection procedures are labor- and material-intensive and cannot guarantee the correctness of the results. It does not offer real-time continuous monitoring and cannot detect high-voltage transmission line safety issues in a timely manner, and its efficiency is low. Because of the rapid advancement of information technology, suitable technological conditions for remote monitoring and evaluation of environmental factors and power transmission operating conditions are now available. Remote monitoring of the state of the power line can be implemented in the power supply system using sensor technology, network technology, and software development (deep learning and other ways of machine learning), providing a decision-making foundation for daily maintenance and line safety. To monitor and handle enormous volumes of data from electrical, electronic, electro-mechanical, and other equipment linked to the grid effectively and efficiently, it is important to monitor them continually [24,25] to gather information on power line integrity. It is feasible to foresee the operating possibilities and failures of the power line and execute timely maintenance and repair work using contemporary diagnostics and predictive maintenance approaches, as well as understanding the performance

characteristics of the linked sources. As a consequence, significant human and financial resources can be saved [26]. To build a platform based on industrial protocols [27] for monitoring and predicting the operation of power lines and connected equipment, one must first develop a system similar to IoT smart networks [28], then a complex distributed data analysis system, and finally a platform that combines both approaches with machine learning and deployment pipelines [29]. Even when constructed, such complicated systems are difficult to test due to the vast amount of hardware and software development required. Sometimes software emulators are employed as a partial solution, with the outcomes heavily reliant on the principles governing the system simulation. This does not always match to actual findings, but it is encouraged to do a primary validation in this manner [30]. Authors do similarly in the original work [31], except that the system is analyzed from the standpoint of data Quality of Service (QoS). The authors of [32] laid some foundations in the area by correctly observing that when simulating network flows, it is not always clear how they can precisely determine the performance of the entire system, so they attempted to investigate the issue of network load, with the clear result that the performance of the system truly depends on network load conditions. This is understandable given that the higher the network load, the greater the latency, data loss, and network adjustment to such factors [33]. Similar findings are obtained by the authors of other publications [34,35], although the authors were no longer able to give further study results previously indicated, suggesting merely that random data flows have an affect. Such studies show that there is always a danger since there may be an unlimited number of alternative circumstances in a distributed system, and it is impossible to simulate them all, thus we frequently focus only on the logical methods of how it can be done. It's worth noting that, in many cases, system scalability and manageability are also considered using QoS metrics [36], but in the case of scalability, we must consider system capacity, or how much data we can process with a fixed system size, and when to expand the system's capacity as the amount of data grows. To build a high availability and reliability platform pipeline, conditions must be provided under which the system can still operate successfully in the event of system malfunction or failure, and such cases must be reported to the user; in this case, the authors [37] recommend using backup devices and duplicating systems. Because nothing extra can be done in the case of a failure such a solution is only appropriate for simple systems. Certain analysis scenarios, however, may be transferred to some analytical engines for complicated cases [38]. In this scenario, having backup data for neural network models and outcomes, as provided in the works [39–41], or building a specific system maintenance mechanism that independently monitors the condition of the entire platform [42,43], is sufficient. In any case, using probability and mathematical models, such as graph networks and their solution methods, are employed for the examination of these factors [44].

In this paper, we propose a hypothesis, that it is possible to perform predictive maintenance on power lines through data transmission analysis and data gathering from devices such as digital power meters, thus avoiding dedicated equipment like power line modems and artificial data streams. The main goal of the work was to research and suggest viable model that may successfully, securely, and reliably function in a power line integrity analysis system, while potentially also usable with monitoring pools of alternative energy source parks and major users such as factories.

The paper is further structured as follows: the Section 1 of the paper serves as an introduction and presents the background context. Section 2 of the paper provides a brief overview of what has been done in the power line failure classification and problem formulation. Section 3 presents the methodology and proposed data analysis model. Section 4 present an experimental setup, while Section 5 presents the measured data and analysis of the results. Finally, Section 6 presents the discussion and conclusions.

2. Overview of Classification Approaches Based on Power Line Data Analysis

Methods for detecting, classifying, and locating faults in transmission lines and distribution systems have been extensively researched throughout the years through analysis of

the power line data [45]. With smart grid concepts gaining traction among academics, the need of developing a smart fault monitoring and diagnostic system capable of identifying and finding various sorts of faults cannot be stressed enough [46]. Even though the voltage and current signals include all of the information, fitting the raw signals into another sets of rules and criteria capable of intelligently deciphering the underlying messages provided by the signals is exceedingly difficult [47], yet there have been devised numerous approaches for building enhanced power line fault analysis and classification algorithms that might serve to increase the reliability of the grid [48] and which this section aims to familiarize the reader with.

Feature extraction techniques are valuable because they actively retrieve useful information while reducing the influence of variance inside the examined system [49]. They enable a better understanding of the nature of fault categorization or localization problems, allowing them to be solved in a more cohesive and efficient manner [50]. Reduced data dimensionality can occasionally improve the efficiency of certain methods used in classifiers or locators, resulting in more accurate and robust findings as quickly as feasible [51]. Failure classification is important in transmission line and power distribution system protection relays [52]. Significant part of classification approaches employ statistics based classifier models, while other studies use logic flows based on experience and data observation. Hidden Markov Models has frequently been applied in power system fault diagnostics, for example, to identify power problems that cause fault condition in a smart grid [53,54]. A categorization index was utilized to distinguish between double-phase to ground and single-phase to ground faults in [55]. The investigation concentrated on the waveform current and voltage sequences' zero and negative sequence values. Bayesian Networks have also been used to find faults in power line systems [56,57], as they are capable of dealing with unclear or partial data from a power failure analysis system [58]. Godse et al. [59] employed a combination of the major principle components and sequential component analysis to characterize and locate the flaws. The properties of current signals are retrieved and utilized to classify faults in [60] using Multi Resolution Analysis. The investigations depended on an immutable feature, and the thresholds were generally preset arbitrarily using a randomization procedure.

The advancement of research in this sector has been very significant to the advancement of pattern recognition and machine learning and its current popular branch—deep learning [61]. A fuzzy inference method was utilized by the authors of [62] to identify, categorize, and pinpoint a fault section in integrated power lines and subterranean power cables. Lopez et al. [63] employed a mix of self-organized maps and decision tree algorithms to find transmission line issues. Professor Coleman [64] centered on the concept that the deployment of supervised machine learning algorithms is dependent on the labeling of all data utilized for categorization. Two-stage classification was employed in [65] by adopting an unsupervised learning strategy to cluster the data, followed by a supervised learning algorithm to perform the training and classification. When using an independent technique for fault identification, the classifier and locator are activated once a secure fault is discovered [66]. Setting certain thresholds for the extracted characteristics makes this simple. Deep learning reduces the need to build additional fault detection methods if the classifier or locator can discriminate between faulty and non-faulty states [67]. In this scenario, one method for detecting faults is to utilize an individualized classifier to distinguish between faulty and non-faulty state. The alternate technique is to include the non-faulty condition in the output classes, and a fault is identified if the output is not in the non-faulty state. Kathari et al. [68] employed the support vector machine (SVM) technique for fault classification. The approach suggested in [69] employed a mix of this technique with fast fourje transformation to extract signal characteristics. Kamaracha et al. [70] employed a wavelet transform and an SVM classifier to identify power system issues, based on the wavelet energy conversion coefficients, while Ren et al. explored the alternative based on decision trees [71]. Fonseca combined this approach with neural networks, achieving better accuracy [72]. The authors of [73] devised a deep learning-based approach in which data is

taken from a power control center and precompiled prior to deep learning network training. The power line data can be processed using auto-encoders, and hidden characteristics are examined to determine the nature of the issue. If a defect is found, the next step is to categorize the fault type, which is done using learned stacking auto-encoders to train a deep learning network [74]. Li et al. suggested a system for detecting short-circuit problems in transmission lines based on a recurrent neural network with long-short term memory units. In the redevelopment process, a minimal neighborhood sample set is chosen from the huge samples based on their similarity, and the samples are subsequently trained using the back—propagation learning technique over time [75]. One of the key fields remaining to be addressed in power line classification scenarios is the coupling of sparse coding and dictionary learning approaches with discriminative models. Sparse models can aid deep learning algorithms reduce sample complexity and enable deep neural networks better deconstruct, compress, and rebuild input data [76]. Edge based processing is a another game-changing method that addresses security concerns about keeping sensitive data from power systems in the cloud while simultaneously reducing burden on cloud networks by processing data locally, providing real-time data processing [77]. It is also utilized to overcome problems with the poor capacity of discovering fault characteristics for line-to-ground or line-to-line faults, which have complex models due to their multiple modes [78], aiding the difficult deployment of recently established power line monitoring technologies to become more efficient [79].

Problem Definition

There are several hurdles to overcome when it comes to monitoring and troubleshooting the performance and integrity of electrical lines from data packet analysis. To begin with, the data can originate from equipment used to gather power data that are often positioned far away from their control centers, and sometimes even at large distances from each other [80]. Continuous data collection from these sites is not always practicable, as is the situation with offshore wind farms. However, monitoring and prompt diagnosis of equipment failures would give great advantages, including effective resource usage and timely maintenance, as well as a large decrease in maintenance costs and optimal material utilization in order to create a sustainable operation [81].

Data transmission can be either unidirectional or bidirectional [82]. All necessary monitoring and diagnostic work may already be performed in especially tough working circumstances with at least one-way data communication. Two-way data transmission would also allow for the control of emerging risks. For example, if the temperature of the power converter or generator increases, it would be possible to turn off the corresponding generator or voltage converter, stop the wind turbine by electromechanically breaking the shaft, or activate more generators whose resistance force would slow down the shaft rotation. Other management approaches described above enable for the efficient and economical utilization of resources. In the event of solar power plants that generate little electricity, automated cleaning and other operations may be necessary. To address the issue of monitoring and diagnostics, numerous activities must be completed as well as appropriate platform designs developed [83]. The following are the primary tasks that must be completed:

- Data. How much and what data will be collected.
- Transfer of data. What technologies and how often data will be transferred.
- Data flow broker. Will a data flow broker be used, as for example is the case with IoT power meters.
- Data collection. How data will be collected.
- Data pre-processing. How the data will be filtered and how often it will be submitted for analysis.
- Data monitoring. Real-time or batch presentation of incoming data.
- System training. Annotating the collected data and training the neural network model.

- Diagnostics of the situation. By applying the received data, diagnostics of the operation of systems and their individual devices and identification of potential risks are performed.
- Re-training of the system. Training with continuously augmented data and adjusting the previously trained network.

Data transmission may be accomplished in a variety of methods, each with its own cost and complexity. The most basic and least expensive method is to use ordinary power ethernet technology or suing a power line modem, which sends data over a power line [84]. When there is a well-developed network of relays nearby, this strategy is convenient [85]. A new branch of such an existing network would be inexpensive to establish and run, and it would allow substantial volumes of data to be transmitted at reasonable rates. Another practical method is to employ the wireless mobile network of many monitoring devices put at various places along the power line. This, like the wired enclosure, would be ideal in a developed location. However, in remote places, it may soon become difficult due to network availability or alternatives (e.g., satellite-based), and the cost of data transfer may be slightly greater. In practice, specialized data transmission technologies such as optical cables combined with power transmission cables, dedicated radio communication equipment, and satellite communication are employed in distant and difficult-to-reach places. Because technologies have been developed to carry data over great distances, running optical fiber alongside energy lines is typically the most optimal solution for both onshore and offshore power plants.

Another critical job is determining how we will manage the many data sources. A variety of industrial data transmission protocols that are not dependent on the data transmission interface have been developed for this purpose. Message Queuing Telemetry Transport (MQTT), which is utilized in IoT applications, is now one of the most popular protocols [86]. This strategy necessitates the use of a specialist broker who can manage several data sources at the same time and communicate telemetry data from a power meter [87]. If a MQTT broker is used for data gathering, it is sufficient to develop a MQTT client that connects to the broker and sends, receives all data. All data transfers must be encrypted using a specialized security protocol. All data transmission from one client to the data collector would be safeguarded in this manner.

The first processing of data before it is given to the automatization pipeline for the monitoring and artificial intelligence network is one of the most challenging parts of the complete power line monitoring pipeline. This entails a number of questions that must be addressed:

- Does the system work in real time;
- Does the system work with a certain time interval, accumulating data packets;
- If data is collected, how long the data must be collected;
- If data is collected, how much data needs to be collected;
- How to take into account if certain data will not be received for a certain period of time;
- Whether data filtering or normalization is necessary.
- Does data collection on the server include all other processes of the full neural network training and deployment pipeline, or is it more of web page front-end with secure access to the internal resources of neural network analysis similar to the REST based services?

Most significantly, consideration must be given to security. Security may be interpreted in several ways, such as whether the system is fault-tolerant if the system can be expanded at the correct moment without losing data, and whether third parties can use or harm the platform maintenance [88,89]. Hu et al. [90] stated in their work, that it is not feasible to assure security in a single approach, thus it is required to monitor how security may be achieved in each case of a new system. Decoupling of data devices from the network layer, secure data storage, and lastly, secure data processing are all of importance [91]. In actuality,

security strategy must be executed all of the time because each of the aforementioned levels operates on distinct principles and can be treated in entirely different ways, making it impossible to unilaterally specify what security must be provided in each area of the system [92]. It should be noted that there has not been much study on power integrity data pipeline themes to explain how the pipeline would appear and how the data would be handled. For example, Refs. [93,94] studied machine learning pipelines, but both do not consider the system's subsequent work in real-time. In addition, only little studies of the changing amount of data for the pipelines are frequently conducted [95]. During the evaluation of the problem, we feel it is advisable to isolate the system of data collecting and/or early data analysis with monitoring from the system of data diagnostics and accumulation also preventing false data injection [96,97]. It is more difficult to explain where the initial data integrity monitoring should be placed along the length of the power line, reducing the opportunity to employ specialized devices to infiltrate the network [98] or by evading MQTT security standards. For example, if we want the system to function in real time, we must consider that data may arrive at irregular intervals and without particular details. Furthermore, it is possible to state that no data has been received for a specific length of time or to utilize previously obtained data until new ones are received, as well as to inform users of a probable failure of a specific cable branch after a particular time.

More issues occur when we attempt to use deep learning for data analysis [99]. To begin with, it is always more convenient to utilize time-accumulated and annotated data for training, and we are not concerned with real time here. It is also critical to consider data submission to the artificial intelligence network for obtaining anomalies [100], predictive maintenance and diagnostic results [101], as well as investigate what the impact will be when the data is incomplete, as well as what size and interval of the data packet will allow the maximum realization of real-time execution, in order to obtain results in a timely manner when needed. The model's training must be ongoing. It may be implemented based on the amount of fresh data or the anticipated timetable, and if data is unavailable due to faults, none or previously gathered data can be utilized. The artificial neural network would be trained with fresh data, allowing it to adapt to the current condition of the acquired data and network, as well as any potential faults. Furthermore, power line parameters vary somewhat over time owing to changes in the environment or breakdowns, and measured parameters change slightly as well. The constant retraining of a network allows for some ongoing improvement of the previously developed diagnostic model.

3. Methodology

3.1. Proposed Data Analysis Model

First, we recommend employing the MQTT protocol broker to regulate and manage the data flow. Of course, there are other options, but the OASIS group recognizes MQTT as an industrial protocol for IoT devices that can operate on an untrusted and unreliable network and deliver brief informative messages. This is useful when we do not receive data and does not prevent the platform from functioning normally. Tasks may be balanced and dispersed based on complexity, platform server load, working time, and other similar characteristics, resulting in a completely devoted and intelligent data analysis platform (the idea is illustrated in Figure 1).
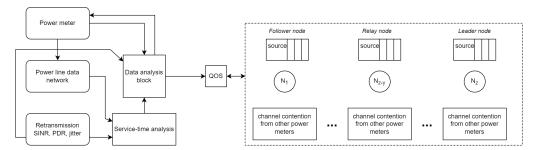

Figure 1. Data analysis flow model.

A separation feature between the internal analysis and diagnostics system and the power network is also required. It enables preliminary data processing and data filtering. Incorrect data can be discovered and erased, or, owing to the neural network's peculiarities, extra data can be introduced to ensure the neural network's proper operation. It is also feasible to synchronize data between various data processing servers. This could clearly have been done with the prior architecture as well. Our proposed data pipeline concepts are divided into two parts: deep neural network model training and model service deployment. These are important components that cannot be replaced with today's technology. The key distinction may be in how the resulting data is produced, as it can be processed outside of the analytics server. Furthermore, where and how the acquired data, trained neuron network models, and obtained diagnostic prediction findings are kept may differ. This process is illustrated in Figure 2.
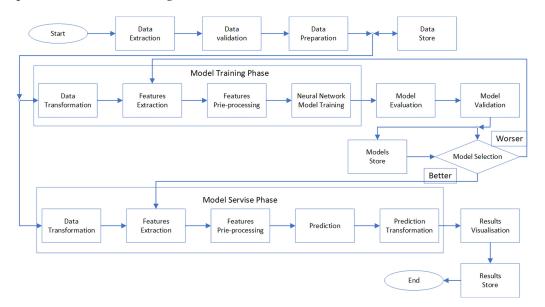

Figure 2. Proposed data analysis block.

In this case, network training and subsequent model deployment are fairly comparable operations. During them, the data is modified so that it can be processed by the chosen neural network. In the case of online training, the data is also annotated throughout the data preparation process. Annotating the data is not required for basic model functioning. The examination and processing of data characteristics occurs during network training. Following validation and review of each result, the most correct outcome is chosen and validated against the trained network with the previously deployed model. If the outcome is superior, it is replaced, and the new model is employed in real time in the pipeline. If it is worse, it can be rejected, and retrained. Because the data is in basic form, this pipeline is incredibly quick in practice, with a measured latency of milliseconds (values from the power meters). The learning and data analysis operations take place concurrently and are not disrupted during regular operation.

3.2. Q-Learning Based Data Analysis Network

We describe a deep learning system for power line diagnostics that is inspired by [29,99] and also builds on signal quality measurements, reusing the communication channel state information, data integrity (packet loss, jitter) to gradually detect, analyze, and pinpoint probable cable degradation. Our technique begins by assessing the sort of deterioration that the cable is experiencing, whether it is limited to a segment of the cable or spreads uniformly over its length. We determine the degree of either form of deterioration to avoid a potential cable in-service failure that might occur if a cable beyond the usual service condition is left untreated. The network aims to identify the presence of a deterioration, analyze the type of damage, and estimate the degree and/or location of the degradation

based on whether it is homogenous or isolated along a part of the cable, while identifying and finding cable damage can aid in the deployment of targeted repair measures, measuring the level of deterioration is crucial in determining the cable's remaining life expectancy and anticipating an impending in-service breakdown.

To that purpose, we apply unsupervised machine learning (Huo's approach [29,99] used supervised) for both classification and regression tasks, retraining model for the job of each diagnostic ahead of time. We also employ channel transfer functions modeled using the bottom-up technique as part of the training procedure, suggested by [102]. This enables us to simulate signal travel over any part of the wire while introducing deterioration of adjustable degree. The total transfer function may then be calculated by concatenating these separate parts. When certain sparse restrictions are applied to self-encoding, the sparseness of learning features indicates that it will perform better in network learning as was determined in the work of [103], by altering the goal function and adding the penalty factor. If the sparse parameter is set to predetermined value v, the neuron's j sparse constraints in the hidden layer can be set after learning. The input layer may be rebuilt using the self-encoding principle and the hidden layer. If the dimension of the encoded signal n the limited hidden network layer becomes less than the dimension of the input signal (data packets from the power meter), it is discovered that the encoded signal may be represented by compressed raw data. When the number of hidden neurons is considerable, sparse restrictions are applied to the self-encoding neural network, where the specific structure of hidden input data may also be discovered.

A topology control system based on the Q-Learning algorithm is proposed, together with the self-adaptive window method. The combination of Q-Learning and self-adaptive windows is used to predict the robust topology of neighbor nodes (power meters) as was implied by [104]. The connection robustness of each neighbor node may be anticipated by real-time learning values from neighbor nodes. Each node in this suggested approach will store a Q-value matrix and a self-adaptive learning window. Each data node may be viewed as one Agent in the Q-learning algorithm-enabled topology control scheme, and each modification to the power line data network becomes a distributed multiple Agent corporation systems.

Finding appropriate sub-windows of different widths during incremental training is computationally difficult; it calls for a thorough comprehension of the complete data stream that has been received so far and simultaneous examination of underlying patterns and the intercorrelations of each sub-window. The size of the segmented sub-window can be divided into several intervals. Variable interval division is preferred over equal interval division in the first stage of our method because it can reveal a better distribution of underlying patterns than equal interval division does. This is because our proposed method is self-adaptive, which means that it can automatically find the best combination of all those sub-windows under a whole sliding window. The general formula for variable size sub-window division is as follows:

$$\sum_{i=1}^{n} sw_i = W,$$

where, in a sliding window, sw_i is the width of the i-th sub-window. Note that due to variable division, each sw_i is not always the same. W is the length of the sliding window, and n is the number of split sub-windows with a range of $1 \leq n \leq W$.

Over a full sliding window, we divide it into sub-windows. By generating both the length and the number of sub-windows beneath a single sliding window at random, it can provide a prospective and effective decomposition of the anticipated patterns. Additionally, by repeated random segmentation, a pool of candidates for different sub-window combinations is created that is totally sufficient. As a result, these candidates can serve as the search space's population as input for further processing. A variable sub-window has the benefit of offering a wider range of possible sub-window combinations. The optimal

sub-window lengths and numbers that fit the underlying pattern distribution are sought after from the combinations.

Here we adopted ADWIN [105] method for calculating adaptive sliding windows. The method maintains a sliding window W with the most recent instances read in it. ADWIN discards the older part of the window, if two big enough subwindows of W show separate enough averages, which indicates that the associated predicted values are different. This entails responding to a statistical question: Has the median "W with confidence" been constant? The definition of a cutting value ϵ_{cut} and the employed test are crucial components of the method. Allow n to represent the size of W, and n_0 and n_1 to represent the sizes of W_0 and W_1, respectively, so that $n = n_0 + n_1$. Let $\hat{W}_0$ and $\hat{W}_1$ represent the averages of W_0 and W_1's values. The value of ϵ_{cut} is set as follows:

$$\epsilon_{cut} = \sqrt{\frac{1}{2m} \cdot \ln \frac{4n}{\delta}}$$

where $m = \frac{2}{1/n_0 + 1/n_1}$ and $\delta \in (0,1)$ is a confidence parameter that is set to 0.2.

According to [105], a change is detected when:

$$\left| \mu^{\wedge}_{W_0} - \mu^{\wedge}_{W_1} \right| \geq \epsilon_{cut}$$

then the oldest parts of W are progressively dropped until obtaining:

$$\left| \mu^{\wedge}_{W_0} - \mu^{\wedge}_{W_1} \right| < \epsilon_{cut}$$

Once the self-adaptive sequence of sliding windows are constructed, we apply the Q-learning algorithm as follows.

The process involves specifying the 4-tuple, where $< S, A, P, R >$ is a transition matrix that may or may not be stochastic and S and A are sets of states and actions. The Markov decision process (MDP) defined by S, A, and P is $s_{i+1}\delta(s_i, a_i)$. The reward function that assigns a state transition value is called $R : S \times A \times S \to \mathbb{R}$. The rewards themselves may or may not be deterministic. A technique to compute a global reward V is also required since the rewards are gained from a single state change, yet the agent is expected to generate an efficient sequence of activities. Finding the ideal policy $\pi : S \to A$ is therefore a way to solve the Q-learning problem. Choosing the optimal course of action in a situation so as to maximize the total return $V^\pi : S \to \mathbb{R}$, where S is the beginning state for implementing the policy. Iteratively assessing using the available actions in each moment must result in the sequence of states assessed by V. The reward is calculated as the discounted cumulative benefit from the created series of states, beginning with a state s_0, as follows:

$$V^\pi(s_0) \to \sum_{i=0}^{\infty} \gamma^i r_i$$

Let $Q_x(d, y)$ is defined as the node x's estimation of the latency from node y to destination d. The method's procedure works as follows:

- Initialize $Q_x(d, y)$ arbitrary $(\forall d, y)$.
- For $t = 0, 1, 2, \ldots$ Choose the action y_t for the current state d_t. Take action y_t, observe $R(d_t)$, d_{t+1} $Q(d_t, y_t) \leftarrow (1 - \alpha_t)Q(d_t, y_t) + \alpha_t[R(d) + \gamma \max_y Q(d_{t+1}, y)]$

where α is a learning parameter controlling the convergence speed.

Up until the maximum number of iterations is achieved, this procedure is repeated. The algorithm updates the Q table after learning the Q value of carrying out various actions in various environmental situations through this iterative computation. The optimal policy, or the best scheduling plan may be found by looking up the final Q table once the iteration of the Q model is finished. can be written as:

$$\pi^* == \arg\max_{a \in A} Q(d, y).$$

The neural network's function is to implement supervised learning. The stochastic gradient descent method (SGD) is used to calculate the gradient, which updates the parameter W and the bias b. The general method builds two Q networks, in which the experience pool provides training samples, the target Q value and the calculated Q value determine the loss function, and the gradient is calculated. We build two networks with identical structures but different parameters. The Q value is calculated and predicted by one network using the most recent parameters, and updated by the other network using values from earlier in time. This guarantees the intended Q value's stability throughout time.

The experience pool provides learning examples in addition to finding a solution to the data correlation issue. At the start of the training and learning process, a memory bank is created. The experience pool stores the state, the action, the reward, and the state of the subsequent time slot once the current action has been completed. A specific quantity of memory data is randomly collected in batches from the experience pool each time a neural network is trained. The original data's order will be disturbed when the experience pool is full, further reducing the data's relevance. At the same time, the new memory will replace the old memory.

The proposed PowerQNet model uses two full connection layers and three convolutional layers in a convolution neural network (CNN) architecture (see Figure 3). Although the design is straightforward, convolution neural networks' characteristics significantly increase training time and computation requirements. Additionally, convolution kernels are unable to extract useful features at the early stages of the algorithm training, which significantly extends the training period of PowerQNet.

The model's hidden layer neurons behave as activation functions for the input data's nonlinear transformation. The nonlinear sigmoid function that we apply has a good impact on the feature space mapping of the signal since it has a strong signal gain in the center and relatively small signal gains on either side. Here *Loss* stands for the loss function during the training procedure (W, b).

$$Loss(W, b) = \frac{1}{N} \sum_{i=1}^{N} l(W, b)$$

where N denotes the overall number of input samples, $loss(W, b)$ denotes the loss function for a single sample, and the calculation expression for $loss(W, b)$ is as follows:

$$l(W, b) = \begin{cases} \frac{1}{2}(q_{eval} - q_t)^2, & \left| q_{eval} - q_t \right| \leq 1 \\ \left| q_{eval} - q_t \right| - \frac{1}{2}, & otherwise \end{cases}$$

where q_{eval} denotes the calculated Q value and q_t denotes the target Q value. The gradient descent method is used to update the parameter W and the bias b after forward.

The proposed architecture employs two neural networks with the same structures. The Target Q-parameter is θ^+, whereas the trained network's Q-parameter is θ^-. Only the Q-parameters are modified during each training session as follows:

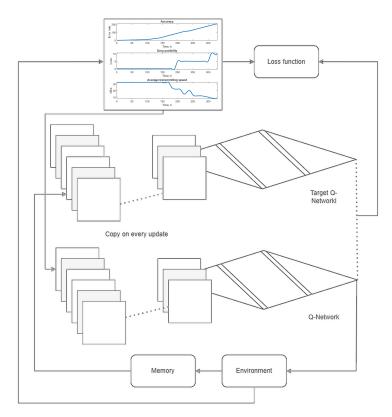$$Loss = \sum \left[\left(R + \gamma max_{a'} Q(S', A', \theta^-) - Q(S, A, \theta^+) \right)^2 \right]$$

Figure 3. The architecture of the proposed PowerQnet model.

4. Experimental Setup

In the VILNIUS TECH area in the city center, 6 smart Sagem T211 3 Phase Power Meter power meters were [106] installed as data sources. Data was collected on a 32-kilometer distance to our primary facility in the Sauletekis neighborhood. This style of power meters is also one of the more common meters put in Lithuanian user houses, demonstrating the practicality of our study. Several tests were carried out in order to determine the dependability of the offered solutions. We analyzed the continual transmission of messages consisting of power meter information.

An MQTT server was built using an ASUS Tinker Board 2 embedded microcontroller running Linux and the open source Mosquito MQTT software [107]. The data collecting computer was linked to the power ethernet router and the 6 analysis servers, which were outfitted with NVIDIA GTX 1080 graphics cards for data analysis and continuous network retraining. The open software Mosquitto MQTT equipment we used can transport up to 268435455 bytes (about 260 MB) of data in a single message, which is more than adequate for our configuration. In the experiment standard RS485 to Ethernet converters such as EPROMA PKZ397WEB are used for conversion from RS485 to Ethernet, and DEVOLO Magic 2 LAN DINrail module is used for transmission over 3-phase electrical networks. Transmission possible up to 500 m and in any case, it is necessary to connect to an internet network because it is not possible to transmit data up to a distance of 32 km without a repeater. In our case it is not a problem, because we are in the urban area and each transformer station now has internet equipment. The block diagram is presented in Figure 4. The experiment made no use of elaborate and expensive automation sensors or monitored electrical energy converters since it makes no sense in light of the data acquired.
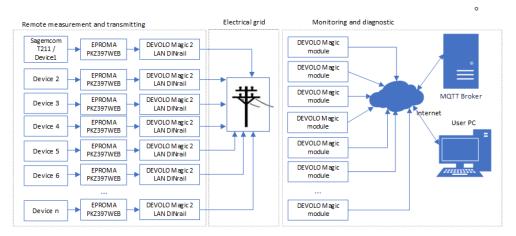
Figure 4. Block diagram of the deployed equipment and communication network.

5. Results

The results were obtained by using the 6 available power meters and by creating several scenarios and simulating certain cases of operation and data transmission. Additional fault simulation was necessary because during the experiments only a couple of power failures occurred due to ongoing repairs and upgrades of the power grid with prior notification from the grid management company. The experiments were designed to investigate the impact on the transmission of data packets of the load on the data network or simply on the Internet and the resulting delay and possible loss of data packets, as well as the impact on the delay in receiving packets of different sizes of data packets. The authors aimed to test the reception of a large number of meters by simulating the transmission of data over a similar time interval, identical to a real city, and to determine how long it would take to receive the data and the amount of traffic that would be generated in the network. Accordingly, we aimed to determine how long it would take to receive the data at the collection server and how much computing resources would be required to receive the data as efficiently as possible for further analysis in relation to the quality of the power line integrity.

5.1. Analysis of Real Data in Small Scale Scenario

128 bytes, 16 kB, 32 kB, and 64 kB data chunks (in different circumstances) were delivered every 1 s during from the power meters Each power meter device also created random blank data, although the beginning of each message array is similar. The array message had an ID number, sensor data NTP time values, and random data. The experiment lasted three hundred thirty hours. The goal of this experiment was to establish a realistic data flow for simulation in experiment number two. The results of analysis are presented in Figures 5–8, respectively, at 128 bytes, 16 kB, 32 kB, and 64 kB.

During the ongoing experiment, the time of all data transfer, regardless of size, did not surpass 300 ms using the available equipment. It should be mentioned, however, that the experiment was carried out in a reasonably controlled context with no extra data flows. Because of the lack of NTP precision, the authors were unable to catch less than 100 ms. Such a time is perfectly appropriate because, in actuality, it is a period of time that has no bearing on the user. It would be fascinating to test a high number of client connections in real time, but this is difficult to achieve and is only doable through simulation.
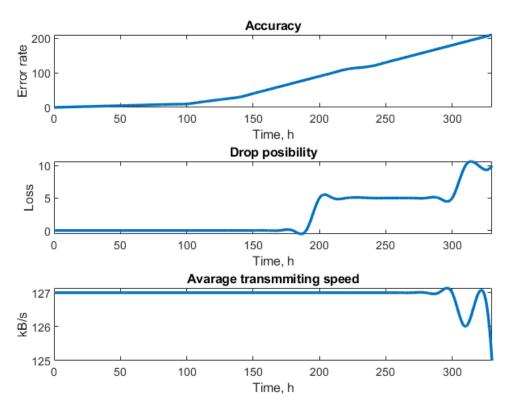
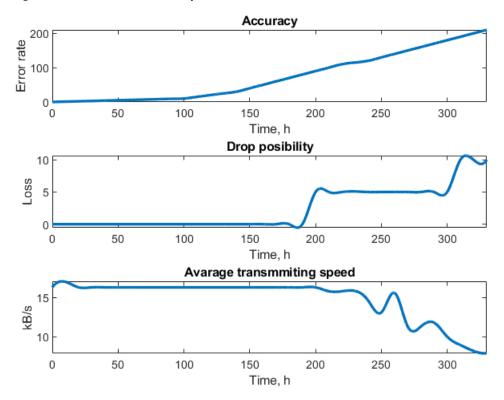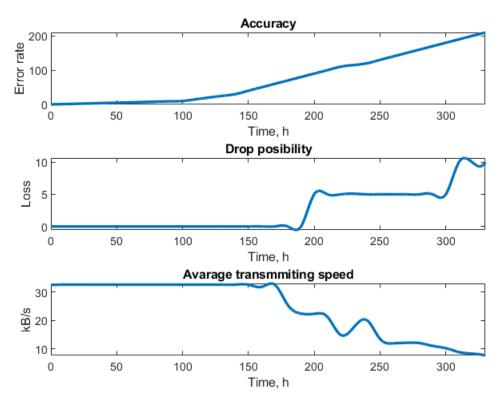**Figure 5.** The circumstances 128 bytes.

Figure 6. The circumstances 16 kB.
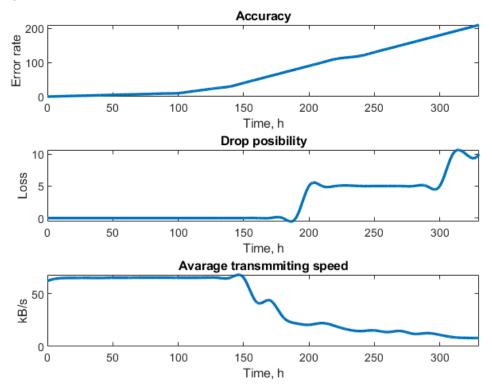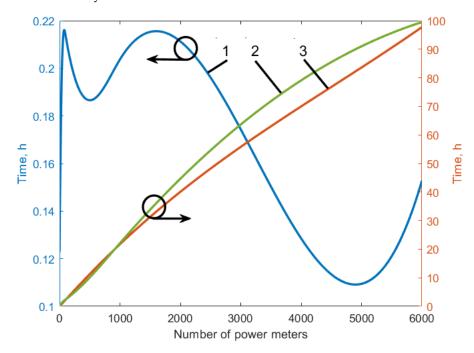
Figure 7. The circumstances 32 kB.

Figure 8. The circumstances 64 kB.

5.2. Analysis of Simulated Full Data Flow of a Whole District

In experiment two, we produced artificial data traffic to imitate from 10 to 6000 power meters, in order to realistically load the data transmission network with as much of a typical power line branch as feasible (potential data of a whole city district power meters). In this scenario, 10 GB and 50 GB data stream was produced and repeatedly broadcast during the trial. Data had to pass over the same power network during testing. There were no delays

or any changes in the transfer of our data, indicating that its integrity was not jeopardized, and the system also responded during high peak hours.
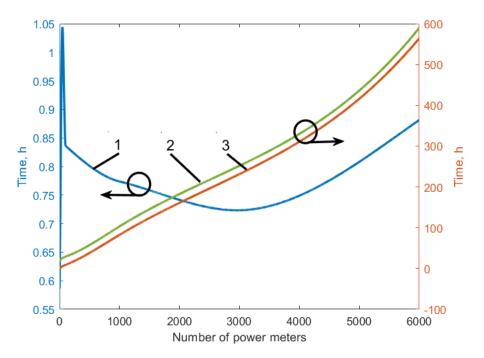
The experiment presented in Figure 9 illustrates a continuous download of 10 GB of data from the Smart meter devices. The resulting data (Graph 1) shows how long it took to transmit a data set from one device and from all devices. The results show that it takes approximately up to 22 min to receive data from a single sender. We see a ripple in the time when sending one data packet. This ripple indicates that a large amount of instantaneous time is required to send the first data packet, but the time decreases slightly as the number of smart meters increases. A larger drop in the data sending time occurs when the data from the first smart meter devices is finished being received, but the sessions are not yet fully freed up to initialize new connections. Then the time starts to increase again and a similar kind of surge occurs until all devices have transmitted data. As the number of Devices increases, even though the data transfer happens at the same time, memory space is required to store the data and for this reason we observe the time taken by the receiver to collect all the data increases (Graph 2 and Graph 3 in Figure 9). This occurs because it is not possible to collect the amount of data used in the study at the same time and to manage all the connections to the senders. Therefore, requests for new connections are put on hold in a buffer until the existing sessions are completed and the server has freed up the necessary memory to create a new session. It should be mentioned that a multi-session/multi-threaded TCP/IP server written in the Python programming language was used for the test year. Comparing the time of receipt (graph 3) with the time of sending (graph 2), we see that the sender finishes the tasks slightly earlier than the receiver. This is natural, since even after the sender has sent an acknowledgement that the transmission is complete, the server software can still process the received data in order to allocate it in the server memory.

Figure 9. The produced artificial data traffic, when 10 GB data stream was produced and repeatedly broadcast during the trial, when 1—transmission time of one power meter, 2—transmission time of all power meters selected, 3—receiver time in system of all power meters selected. Black circles shows y axis.

Another data analysis experiment was carried out with 50 GB of data (see Figure 10) to analyze a case of more smart meters connected. In this case, it can be seen that the download time per data packet approaches 1.02 h. Afterwards, the sending time decreases slightly in time as the data volume is spread out and the experiment settles down to a

slightly steady rate as the number of IoT devices increases. Wave fading here indicates that a very large amount of data is being used and when one data transmission session is over, other sessions are still processing a very large amount of data, which does not allow for a faster initialization of a new pending session in the queue. As was noticed in the case of previous scenario (Figure 9), in this case (Figure 10) the sender and server data transfer times are slightly different. As the number of smart meter devices increases, the sending and receiving times change linearly and the results are similar to those seen previously, with only a slight difference in the slope of the curves, either to the left or to the right.

Figure 10. The produced artificial data traffic, when 50 GB data stream was produced and repeatedly broadcast during the trial, when 1—transmission time of one power meter, 2—transmission time of all power meters selected, 3—receiver time in system of all power meters selected.

5.3. Analysis of Real Data in Power Line Failure Scenarios

We have used a T211 3 Phase Power Meter within the input installation of our building as well as adding power consumers to get real data, when one phase was disconnected, as well as when the network failed due to high load. The goal of this experiment was to establish realistic reference data for simulation in experiment number four. The analysis results of experiment three are presented in Figure 11.

During our investigation we found, that data was not obtained from certain meters during the duration of experiment because they were "unplugged" by turning off sections of the input line (scheduled maintenance or installations). In this situation, the MQTT broker and server client ceased receiving data for an extended period of time, if at all. The developed system can alert users to possible difficulties as long as the MQTT broker is active and the server may listen for data but not accept it. There are two types of situations based on the projected waiting time:

1. Due to network issues, data from the meter is arriving with a larger delay than intended and at irregular intervals.
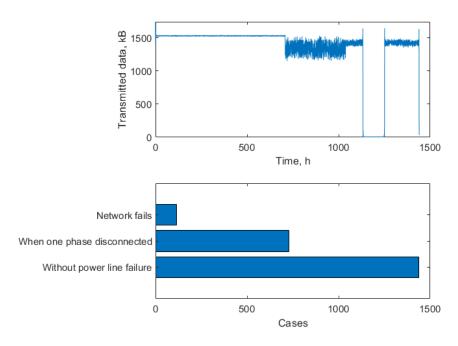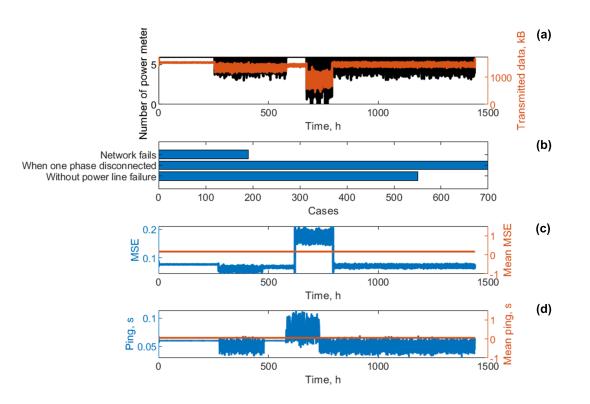2. The data does not reach the broker for an extended length of time, indicating a malfunction in the system.

Figure 11. The results of size of transmitted data related to the transmissions failures.

This experiment was carried out to collect data every 15 min for 60 days from 6 three-phase smart voltage and power meters. It simulated the frequency of power grid failures. The nature of the disturbances was assumed to be as follows: when one of the three phases of the meter goes down, or the meter stops working altogether, due to a lack of power supply, and the data transmission fails. Such results were obtained 116 times during the simulation (Figure 11). This can occur due to lightning discharges during storms, short emergency and power grid failures or repairs and upgrades, and careless behaviour of consumers that triggers the protection of the power transmission grid. For similar problems, but only for one phase, a simulated disconnection occurred 728 times, while normal operation was foreseen 1439 times. According to this distribution, data collection was carried out using the MQTT protocol and the transmission of data from 6 m was monitored.

5.4. Classification of of Power Line Failures

Finally we have validated how the developed and built platform reacts to data loss from our T211 3 Phase meters, classifying potential power line failures. In other words, if data is not received at a specific time or if it is received incompletely, as well as tolerance for jitter and ping. Such an experiment informs us about the reliability, fault-tolerance capabilities, and flexibility of power lines in various living conditions, and what is crucial is what it reveals about our pipeline dependability and fault-tolerance. The analysis results of experiment are presented in Figure 12.

The results of the experiment are plotted in Figure 12a, which shows how the number of meters (left side) and the number of transmitted data (right side) vary over time. As very few disturbances were noticed in the real environment monitoring as Lithuanian grid is one of the most stable in the Baltic sea region, therefore we manually introduced failure simulations, operating the circuit breaker boxes at university facilities throughout the city. Figure 12b. Figure 12a shows the results varying and showing a clear decrease in the number of meters and the amount of data flow if any meter experiences a power failure or a meter malfunction. Figure 12a clearly shows a small drop of 200 kB if a single phase failure occurs in one of the meters, resulting in a minimum value of 530 for the data rate. In turn, if a meter fails to send data for a longer period of time, the number of meters "visible" and the data throughput were reduced, as for example can be seen in Figure 12a when 4 m were down between 690 and 787 h.
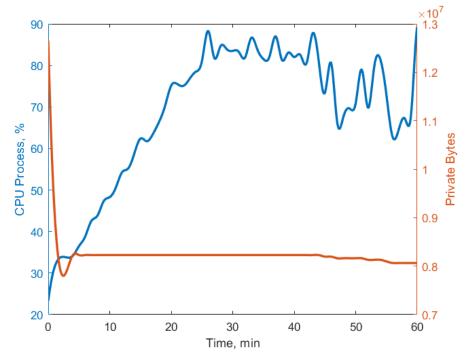
Figure 12. The analysis of network integrity by (**a**)—number of power meters related to their data transmission and (**b**)—number of failures, (**c**)—Mean square error rate of failures and (**d**)—data packets delay time through the experiment period

As was already mentioned, the power supply to the meters in Figure 12b was additionally disrupted as there were only a couple of outages during the monitoring period, making it difficult to predict how the system would work. In this case, an additional 190 cases of total power loss and 700 cases of single phase loss were created. Observing the number of possible disturbances from the classification results shown in Figure 12c, it can be seen that the mean square error (MSE) of the 6 monitored meters did not exceed 0.1% of the total number of cases, and that the MSE of the meters for a complete disturbance was only 0.2%, which is why the average number of possible cases of errors and disturbances during the entire operation did not exceed 0.12%.

Figure 12d shows how the data delay varies with time. From Figure 12d we can see that a power line outage affects data transmission, as the meters are unable to provide data in the event of a complete loss of power, and in the event of a connection failure and a glitch, the receiver reserves resources and waits for the data that does not arrive. This results in data being received after the scheduled time or in bursts. The ping used indicates how long it can take to transmit data in a system with interference. In the general case, with additional simulated meter failures or power failures, it did not exceed 0.06 s for the whole test.

5.5. Computational Performance

The graph in Figure 13 shows that the recipient, the server software that collects the data of all devices, requires about 88 MB of private memory at initialization, but this is reduced and settles. This is related to the monitoring of modern operating system resources and the reduction of memory, provided that memory is not used for an extended period of time. The demand on the private memory of the process being used does not increase any more when the data being sent is 10 GB in size. In addition, the transfer is spread over time, so that the actual size of the data transfer at a given point in time is smaller and does not

affect or increase the amount of memory space occupied. This result is highly dependent on the available server hardware and processing speed.

Figure 13. Computational performance, when 10 GB data stream was produced.

In the case of Figure 14 the situation changes slightly with memory allocations as in this case 5 times more data is transferred. In this case, the memory provided in the server is no longer sufficient and it is necessary to provide considerably more memory to be able to receive a considerably larger amount of data. In this case, the server software uses as much memory as the operating system allows per process (during the tests, the server's 32 RAM was used to its maximum and additional SWAP memory was used).
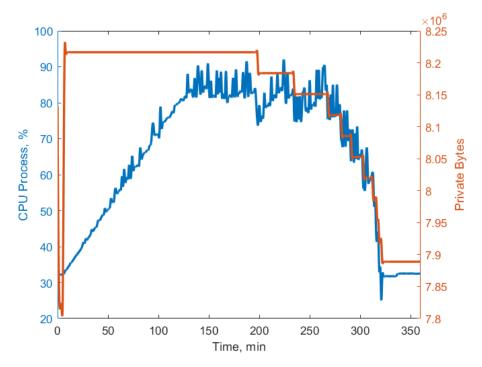

Figure 14. Computational performance, when 50 GB data stream was produced.

The data transfer is never identical and therefore the amount of memory used decreases by a small amount of about 10 GB after the first data packets are transferred. The decrease is in the form of a staircase, where the duration of the staircase is continuously decreasing as the amount of data remaining to be received decreases and the amount of RAM used decreases accordingly.

Looking at the microprocessor load from the first graph, it can be seen that the microprocessor is heavily involved in the data fetch and distribution operations and even a small amount of data transfer and multiple Power Meter devices, almost to a maximum of 90% load on the Intel i7-12700K microprocessor. A similar characteristic was obtained in the second test with a significantly higher data load. However, the second test showed a significantly more pronounced variation in the microprocessor load and a more uniform decrease in the same way as in the case of memory utilisation. This can be explained by the fact that modern microprocessors try to distribute the tasks over several microkernels, so that when some cores become free, a change of about 10% occurs. In the general case, the microprocessor used has 8 fast computing cores and 4 efficient cores, which are reserved for smaller tasks and the variation is similar to the number of cores in the microprocessor.

In the first case, the significantly smaller amount of data did not require the use of a larger number of microkernels, so that in practice the end-of-run graph variation was abrupt, as in the second test, from 60 to 30% at 320 min.

After concurrent connections to the senders, and completion of the operations, the resources used by the server software stabilize, as seen in test 2 at 320 min.

6. Discussion and Conclusions

The developed approach was successful in indicating potential power line breakdowns. Essentially, this would only effect the real-time monitoring system because the data would be less frequently updated. Because the data is often provided on a time scale, the subsequent result can simply be displayed in another period with some information about other probable faults. Because we could potentially simulate any stationary device using the IoT paradigm installed at different points over power line for data management and seamlessly adapt solutions developed for this area to data transmission and large data management, the proposed architectures have the potential to make excellent reuse of IoT solutions already developed.

During the experimental evaluation, the distance of the devices was varied by simulating the delay of data packets (as the distance increases, more routers appear that delay the packets), and at high interference and delay, data losses (max up to 20%) are possible. With a small packet of data, as can be seen from the results, the 128 B Internet TCP/IP protocol is able to successfully transmit data on time within a 1 s interval, and delays do not occur that exceed 1 s. Even with high packet losses, data is received on time. In the general case, 100 data packets are sent with an interval of 1 s during the test. As the size of the data packets increases, it can be seen that the total delay of the data packets starts to increase at lower latency. At a packet size of 16 kB, somewhere around 210 ms, a delay of more than 1 s starts to appear, resulting in an increase in the transmission time of all data packets. In turn, for a 32 kB packet, the latency is at 180 ms and for a 64 kB packet at 150 ms and increases very significantly.

Our Q-network properly detected and classified simulated 190 incidents of total power loss and 700 cases of single phase loss. The mean square error (MSE) did not exceed 0.10% of the total number of instances, and the MSE of the smart meters for a complete disturbance was only 0.20%, resulting in an average number of conceivable cases of errors and disturbances of 0.12% for the whole operation.

We see that the amount of data sent varies depending on the fault and the intended data packet size. The maximum number of data sent in a packet is 256 kB. When one phase is disconnected, the number of data sent decreases in time slightly. There are two possible cases. The first one is when the meter sees that there is no data and therefore sends slightly less information than when all 3 phases are active. The second case is when it is possible to

control programmatically how much data should be transmitted in the case where only one phase is used. Since we know that this is not the case in our experiment, we do not discuss the programmatic disconnection. The reduction of the single phase monitoring data reduces the data size from 1523 kB to 1123 kB and in this case the 3 m did not have 1 phase. In the case where all phases are missing from all meters we see a sharp drop in the amount of data received to 0 kB. During the monitoring this indicates that there is a loss of power supply to all 6 m. When even one phase appears we again get only part of the information. Of course, such situation does not provide information when all phases occur, because at that point we would see the data volume return to normal operation. Therefore we did not consider the possibility to control each source individually, because in this case the data would change for an individual sensor and at then end we would see a differently graded result.

In future works, the authors will try to evaluate how monitoring and diagnostic by using artificial intelligence models will work taking into account expected data delays or losses. Would the loss or delay of a small amount of data meet the possibilities of real-time monitoring and diagnostics, and how would the situation be improved by creating a digital twin of a 3-phase power supply grid, combining the received telemetry data with a digital twin.

In this work, unlike others, we implement the data transmission using the part of real electrical network. In this case, we have an advantage because we no longer need to use a more expensive mobile connection. Of course, when transmitting data over longer distances, it is necessary to use digital signal amplifiers or repeaters, but it would be difficult to integrate them without the intervention of the power grid operator. Currently, there is no way to get those that work with 3-phase networks by conventional means, so in any case you have to connect to a wired or mobile Internet network only significantly later than directly from the distribution box of the electricity supply. Of course, broadband over power lines has already been used, but it still uses signal amplifiers. Electricity network operators also always try to connect data to the Internet network as early as possible. Another important aspect is that many of the discussed studies conduct experiments only through simulation or based on mathematical logic. In this work, a large part of the experiments was performed with real equipment, and in order to test a large amount of equipment, corresponding computer equipment was connected to the measuring equipment and additionally the data of new simulated devices was sent using real physical electrical networks.

Author Contributions: Conceptualization, R.M. and D.P.; Data curation, R.P. and D.P.; Formal analysis, R.M., R.P., R.D. and D.P.; Funding acquisition, R.M. and D.P.; Investigation, R.M., R.P., V.K.H. and D.P.; Methodology, R.P., R.D. and D.P.; Project administration, D.P.; Resources, R.P.; Software, R.P.; Supervision, R.M. and D.P.; Validation, R.M., R.P., V.K.H. and D.P.; Visualization, D.P.; Writing—original draft, R.M., R.P. and D.P.; Writing—review & editing, R.M. and D.P. All authors have read and agreed to the published version of the manuscript.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

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

Abbreviations

The following abbreviations are used in this manuscript:

SDN	software defined networking
MQTT	Message Queuing Telemetry Transport
SVM	Support Vector Machine
DL	Deep learning
QOS	Quality of Service
ADWIN	Adaptive Windowing approach
MDP	Markov decision process
Q-learning	model-free reinforcement learning algorithm
Sagemcom T211	Model of Smart Power meter
TCP/IP	Transmission Control Protocol/Internet Protocol
Mosquitto MQTT	Open source (EPL/EDL licensed) message broker software
NTP	Network Time Protocol
MSE	Mean Square Error

References

1. Hou, J.; Guo, H.; Wang, S.; Zeng, C.; Hu, H.; Wang, F. Design of a Power Transmission Line Monitoring System Based upon Edge Computing and Zigbee Wireless Communication. *Mob. Inf. Syst.* **2022**, *2022*, 9379789. [CrossRef]
2. Bin, X.; Qing, C.; Jun, M.; Yan, Y.; Zhixia, Z. Research on a Kind of Ubiquitous Power Internet of Things System for Strong Smart Power Grid. In Proceedings of the 2019 IEEE Innovative Smart Grid Technologies—Asia (ISGT Asia), Chengdu, China, 21–24 May 2019. [CrossRef]
3. Jabir, H.; Teh, J.; Ishak, D.; Abunima, H. Impacts of Demand-Side Management on Electrical Power Systems: A Review. *Energies* **2018**, *11*, 1050. [CrossRef]
4. Tejwani, R.; Kumar, G.; Solanki, C. Remote Monitoring for Solar Photovoltaic Systems in Rural Application Using GSM Voice Channel. *Energy Procedia* **2014**, *57*, 1526–1535. [CrossRef]
5. Meligy, R.; Lopez-Iturri, P.; Astrain, J.J.; Picallo, I.; Klaina, H.; Rady, M.; Paredes, F.; Montagnino, F.; Alejos, A.V.; Falcone, F. Low-Cost Cloud-Enabled Wireless Monitoring System for Linear Fresnel Solar Plants. *Eng. Proc.* **2020**, *2*, 6. [CrossRef]
6. Zhang, H.; Zhang, D.; Zhou, Y.; Cutler, M.E.J.; Cui, D.; Zhang, Z. Quantitative Analysis of the Interaction between Wind Turbines and Topography Change in Intertidal Wind Farms by Remote Sensing. *J. Mar. Sci. Eng.* **2022**, *10*, 504. [CrossRef]
7. Sayed, M.; Nemitz, M.; Aracri, S.; McConnell, A.; McKenzie, R.; Stokes, A. The Limpet: A ROS-Enabled Multi-Sensing Platform for the ORCA Hub. *Sensors* **2018**, *18*, 3487. [CrossRef]
8. Ahmed, M.; Kim, Y.C. Hierarchical Communication Network Architectures for Offshore Wind Power Farms. *Energies* **2014**, *7*, 3420–3437. [CrossRef]
9. Ogunleye, O.; Alabi, A.; Misra, S.; Adewumi, A.; Ahuja, R.; Damasevicius, R. *Comparative Study of the Electrical Energy Consumption and Cost for a Residential Building on Fully ac Loads Vis-a-Vis One on Fully dc Loads*; Lecture Notes in Electrical Engineering; Springer: Singapore, 2020; Volume 612, pp. 395–405.
10. Makhadmeh, S.N.; Al-Betar, M.A.; Alyasseri, Z.A.A.; Abasi, A.K.; Khader, A.T.; Damaševičius, R.; Mohammed, M.A.; Abdulka-reem, K.H. Smart home battery for the multi-objective power scheduling problem in a smart home using grey wolf optimizer. *Electronics* **2021**, *10*, 447. [CrossRef]
11. Makhadmeh, S.N.; Al-Betar, M.A.; Awadallah, M.A.; Abasi, A.K.; Alyasseri, Z.A.A.; Doush, I.A.; Alomari, O.A.; Damaševičius, R.; Zajančkauskas, A.; Mohammed, M.A. A Modified Coronavirus Herd Immunity Optimizer for the Power Scheduling Problem. *Mathematics* **2022**, *10*, 315. [CrossRef]
12. Tao, F.; Qi, Q. New IT Driven Service-Oriented Smart Manufacturing: Framework and Characteristics. *IEEE Trans. Syst. Man Cybern. Syst.* **2019**, *49*, 81–91. [CrossRef]
13. Chen, W.L.; Huang, C.T.; Chen, K.S.; Chang, L.K.; Tsai, M.C. Design and Realization of Microcontroller-Based Remote Status Monitoring System for Smart Factory Task Planning Applications. In Proceedings of the 2021 60th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE), Tokyo, Japan, 8–10 September 2021; pp. 1500–1505.
14. Tamang, D.; Pozzebon, A.; Parri, L.; Fort, A.; Abrardo, A. Designing a Reliable and Low-Latency LoRaWAN Solution for Environmental Monitoring in Factories at Major Accident Risk. *Sensors* **2022**, *22*, 2372. [CrossRef]
15. Yang, L.; Fan, J.; Liu, Y.; Li, E.; Peng, J.; Liang, Z. A Review on State-of-the-Art Power Line Inspection Techniques. *IEEE Trans. Instrum. Meas.* **2020**, *69*, 9350–9365. [CrossRef]
16. Mudaliar, M.D.; Sivakumar, N. IoT based real time energy monitoring system using Raspberry Pi. *Internet Things* **2020**, *12*, 100292. [CrossRef]
17. Bu, X.; Liu, S.; Yu, Q. Framework Design of Edge IoT Agent Used in State Grid Electrical Internet of Things. In Proceedings of the 2019 IEEE International Conference on Energy Internet (ICEI), Nanjing, China, 27–31 May 2019; pp. 185–190. [CrossRef]

18. Liu, C.; Hao, H.; Li, B.; Wang, C. SDN Mixed Mode Flow Table Release Mechanism Based on Network Topology. In Proceedings of the 2017 6th International Conference on Measurement, Instrumentation and Automation (ICMIA 2017), Zhuhai, China, 29–30 June 2017; Atlantis Press: Zhengzhou, China 2017. [CrossRef]

19. Liu, C.; Zhang, G.; Tao, J.; Jiang, D.; Qi, S.; Lu, J. Active Measurement Approach to Traffic QoS Sensing for Smart Network Access in SDN. In Proceedings of the 2019 IEEE International Conference on Industrial Internet (ICII), Orlando, FL, USA, 10–12 November 2019; pp. 103–108. [CrossRef]

20. Dias, C.F.; de Oliveira, J.R.; de Mendonça, L.D.; de Almeida, L.M.; de Lima, E.R.; Wanner, L. An IoT-Based System for Monitoring the Health of Guyed Towers in Overhead Power Lines. *Sensors* **2021**, *21*, 6173. [CrossRef]

21. Tao, X.; Zhang, D.; Wang, Z.; Liu, X.; Zhang, H.; Xu, D. Detection of Power Line Insulator Defects Using Aerial Images Analyzed With Convolutional Neural Networks. *IEEE Trans. Syst. Man Cybern. Syst.* **2020**, *50*, 1486–1498. [CrossRef]

22. Breviglieri, P.; Erdem, T.; Eken, S. Predicting Smart Grid Stability with Optimized Deep Models. *SN Comput. Sci.* **2021**, *2*, 73. [CrossRef]

23. Erdem, T.; Eken, S. Layer-Wise Relevance Propagation for Smart-Grid Stability Prediction. In *Pattern Recognition and Artificial Intelligence*; Springer International Publishing: Berlin/Heidelberg, Germany, 2022; pp. 315–328. [CrossRef]

24. Moosavi, S.S.; Djerdir, A.; Ait-Amirat, Y.; Khaburi, D.A.; N'Diaye, A. Artificial neural network-based fault diagnosis in the AC–DC converter of the power supply of series hybrid electric vehicle. *IET Electr. Syst. Transp.* **2016**, *6*, 96–106. [CrossRef]

25. Shah, F.M.; Maqsood, S.; Damaševičius, R.; Blažauskas, T. Disturbance rejection and control design of mvdc converter with evaluation of power loss and efficiency comparison of sic and si based power devices. *Electronics* **2020**, *9*, 1878. [CrossRef]

26. Alseiari, A.Y.; Farrel, P.; Osman, Y. Notice of Removal: The Impact of Artificial Intelligence Applications on the Participation of Autonomous Maintenance and Assets Management Optimisation within Power Industry: A Review. In Proceedings of the 2020 IEEE 7th International Conference on Industrial Engineering and Applications (ICIEA), Bangkok, Thailand, 16–21 April 2020. [CrossRef]

27. Morgan, J.; Halton, M.; Qiao, Y.; Breslin, J.G. Industry 4.0 smart reconfigurable manufacturing machines. *J. Manuf. Syst.* **2021**, *59*, 481–506. [CrossRef]

28. Kim, H.S.; Park, Y.J.; Kang, S.J. Secured and Deterministic Closed-Loop IoT System Architecture for Sensor and Actuator Networks. *Sensors* **2022**, *22*, 3843. [CrossRef]

29. Huo, Y.; Prasad, G.; Atanackovic, L.; Lampe, L.; Leung, V.C.M. Cable Diagnostics With Power Line Modems for Smart Grid Monitoring. *IEEE Access* **2019**, *7*, 60206–60220. [CrossRef]

30. Frincu, M.; Irimie, B.; Selea, T.; Spataru, A.; Vulpe, A. Evaluating Distributed Systems and Applications through Accurate Models and Simulations. In *Studies in Big Data*; Springer International Publishing: Berlin/Heidelberg, Germany, 2018; pp. 1–18. [CrossRef]

31. Hill, J.H.; Schmidt, D.C.; Edmondson, J.R.; Gokhale, A.S. Tools for Continuously Evaluating Distributed System Qualities. *IEEE Softw.* **2010**, *27*, 65–71. [CrossRef]

32. Jogalekar, P.; Woodside, M. Evaluating the scalability of distributed systems. *IEEE Trans. Parallel Distrib. Syst.* **2000**, *11*, 589–603. [CrossRef]

33. Mesbahi, M.R.; Rahmani, A.M.; Hosseinzadeh, M. Reliability and high availability in cloud computing environments: A reference roadmap. *Hum.-Centric Comput. Inf. Sci.* **2018**, *8*, 20. [CrossRef]

34. Yang, B.; Liu, D. Research on Network Traffic Identification based on Machine Learning and Deep Packet Inspection. In Proceedings of the 2019 IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), Chengdu, China, 15–17 March 2019. [CrossRef]

35. Vieira, T.; Soares, P.; Machado, M.; Assad, R.; Garcia, V. Measuring Distributed Applications through MapReduce and Traffic Analysis. In Proceedings of the 2012 IEEE 18th International Conference on Parallel and Distributed Systems, Singapore, 17–19 December 2012. [CrossRef]

36. Khan, R.H.; Heegaard, P.E.; Ullah, K.W. Performance Evaluation of Distributed System Using SPN. In *Petri Nets—Manufacturing and Computer Science*; InTech: London, UK 2012. [CrossRef]

37. PHADKE, A.G.; WALL, P.; DING, L.; TERZIJA, V. Improving the performance of power system protection using wide area monitoring systems. *J. Mod. Power Syst. Clean Energy* **2016**, *4*, 319–331. [CrossRef]

38. Kefalakis, N.; Roukounaki, A.; Soldatos, J. A Configurable Distributed Data Analytics Infrastructure for the Industrial Internet of things. In Proceedings of the 2019 15th International Conference on Distributed Computing in Sensor Systems (DCOSS), Santorini Island, Greece, 29–31 May 2019. [CrossRef]

39. Yan, Z.; Zhang, L.; Ding, W.; Zheng, Q. Heterogeneous Data Storage Management with Deduplication in Cloud Computing. *IEEE Trans. Big Data* **2019**, *5*, 393–407. [CrossRef]

40. Das, C.K.; Bass, O.; Kothapalli, G.; Mahmoud, T.S.; Habibi, D. Overview of energy storage systems in distribution networks: Placement, sizing, operation, and power quality. *Renew. Sustain. Energy Rev.* **2018**, *91*, 1205–1230. [CrossRef]

41. Kermarrec, A.M.; Merrer, E.L.; Straub, G.; van Kempen, A. Availability-Based Methods for Distributed Storage Systems. In Proceedings of the 2012 IEEE 31st Symposium on Reliable Distributed Systems, Irvine, CA, USA, 8–11 October 2012. [CrossRef]

42. Wang, N.; Li, M.; Xiao, B.; Ma, L. Availability analysis of a general time distribution system with the consideration of maintenance and spares. *Reliab. Eng. Syst. Saf.* **2019**, *192*, 106197. [CrossRef]

43. Fang, J.; Wan, S.; Huang, P.; Xie, C.; He, X. Early Identification of Critical Blocks: Making Replicated Distributed Storage Systems Reliable Against Node Failures. *IEEE Trans. Parallel Distrib. Syst.* **2018**, *29*, 2446–2459. [CrossRef]

44. Wang, X.; Zhao, X.; Wu, C.; Wang, S. Mixed shock model for multi-state weighted k-out-of-n: F systems with degraded resistance against shocks. *Reliab. Eng. Syst. Saf.* **2022**, *217*, 108098. [CrossRef]

45. Shakiba, F.M.; Azizi, S.M.; Zhou, M.; Abusorrah, A. Application of machine learning methods in fault detection and classification of power transmission lines: A survey. *Artif. Intell. Rev.* **2022**. [CrossRef]

46. Chen, K.; Huang, C.; He, J. Fault detection, classification and location for transmission lines and distribution systems: A review on the methods. *High Volt.* **2016**, *1*, 25–33. [CrossRef]

47. Goswami, T.; Roy, U.B. Predictive Model for Classification of Power System Faults using Machine Learning. In Proceedings of the TENCON 2019—2019 IEEE Region 10 Conference (TENCON), Kochi, India, 17–20 October 2019; pp. 1881–1885. [CrossRef]

48. Mukherjee, A.; Kundu, P.K.; Das, A. Transmission Line Faults in Power System and the Different Algorithms for Identification, Classification and Localization: A Brief Review of Methods. *J. Inst. Eng. (India) Ser. B* **2021**, *102*, 855–877. [CrossRef]

49. Zin, A.A.M.; Saini, M.; Mustafa, M.W.; Sultan, A.R.; Rahimuddin. New algorithm for detection and fault classification on parallel transmission line using DWT and BPNN based on Clarke's transformation. *Neurocomputing* **2015**, *168*, 983–993. [CrossRef]

50. Moravej, Z.; Pazoki, M.; Khederzadeh, M. New Pattern-Recognition Method for Fault Analysis in Transmission Line With UPFC. *IEEE Trans. Power Deliv.* **2015**, *30*, 1231–1242. [CrossRef]

51. Jiang, J.A.; Chuang, C.L.; Wang, Y.C.; Hung, C.H.; Wang, J.Y.; Lee, C.H.; Hsiao, Y.T. A Hybrid Framework for Fault Detection, Classification, and Location—Part I: Concept, Structure, and Methodology. *IEEE Trans. Power Deliv.* **2011**, *26*, 1988–1998. [CrossRef]

52. Subashini, A.; Claret, S.P.A. A literature survey on fault identification and classification system using machine learning. In Proceedings of the 2nd International Conference on Mathematical Techniques and Applications: ICMTA 2021; AIP Publishing: Melville, NY, USA 2022. [CrossRef]

53. Hasheminejad, S. A new high-frequency-based method for the very fast differential protection of power transformers. *Electr. Power Syst. Res.* **2022**, *209*, 108032. [CrossRef]

54. Kumar, D.; Bhowmik, P.S. Hidden Markov Model Based Islanding Prediction in Smart Grids. *IEEE Syst. J.* **2019**, *13*, 4181–4189. [CrossRef]

55. Chen, K.; Hu, J.; He, J. Detection and Classification of Transmission Line Faults Based on Unsupervised Feature Learning and Convolutional Sparse Autoencoder. *IEEE Trans. Smart Grid* **2018**, *9*, 1748–1758. [CrossRef]

56. Sahu, A.R.; Palei, S.K. Fault analysis of dragline subsystem using Bayesian network model. *Reliab. Eng. Syst. Saf.* **2022**, *225*, 108579. [CrossRef]

57. Zhang, J.; Bian, H.; Zhao, H.; Wang, X.; Zhang, L.; Bai, Y. Bayesian Network-Based Risk Assessment of Single-Phase Grounding Accidents of Power Transmission Lines. *Int. J. Environ. Res. Public Health* **2020**, *17*, 1841. [CrossRef] [PubMed]

58. Sun, B.; Li, Y.; Wang, Z.; Yang, D.; Ren, Y.; Feng, Q. A combined physics of failure and Bayesian network reliability analysis method for complex electronic systems. *Process Saf. Environ. Prot.* **2021**, *148*, 698–710. [CrossRef]

59. Godse, R.; Bhat, S. Mathematical Morphology-Based Feature-Extraction Technique for Detection and Classification of Faults on Power Transmission Line. *IEEE Access* **2020**, *8*, 38459–38471. [CrossRef]

60. Mishra, D.P.; Ray, P. Fault detection, location and classification of a transmission line. *Neural Comput. Appl.* **2017**, *30*, 1377–1424. [CrossRef]

61. Prasad, A.; Edward, J.B.; Ravi, K. A review on fault classification methodologies in power transmission systems: Part—I. *J. Electr. Syst. Inf. Technol.* **2018**, *5*, 48–60. [CrossRef]

62. Sadeh, J.; Afradi, H. A new and accurate fault location algorithm for combined transmission lines using Adaptive Network-Based Fuzzy Inference System. *Electr. Power Syst. Res.* **2009**, *79*, 1538–1545. [CrossRef]

63. Lopez, J.R.; de Jesus Camacho, J.; Ponce, P.; MacCleery, B.; Molina, A. A Real-Time Digital Twin and Neural Net Cluster-Based Framework for Faults Identification in Power Converters of Microgrids, Self Organized Map Neural Network. *Energies* **2022**, *15*, 7306. [CrossRef]

64. Coleman, N.S.; Schegan, C.; Miu, K.N. A study of power distribution system fault classification with machine learning techniques. In Proceedings of the 2015 North American Power Symposium (NAPS), Charlotte, NC, USA, 4–6 October 2015. [CrossRef]

65. Li, L.; Che, R.; Zang, H. A fault cause identification methodology for transmission lines based on support vector machines. In Proceedings of the 2016 IEEE PES Asia-Pacific Power and Energy Engineering Conference (APPEEC), Xi'an, China, 25–28 October 2016. [CrossRef]

66. Mtwa, Y.A.; Haque, A.; Halabi, T. A Review and Taxonomy on Fault Analysis in Transmission Power Systems. *Computation* **2022**, *10*, 144. [CrossRef]

67. Teng, S.; Li, J.; He, S.; Fan, B.; Hu, S. On-line Fault Diagnosis Technology and Application Based on Deep Learning of Fault Characteristic of Power Grid. *J. Phys. Conf. Ser.* **2021**, *2023*, 012023. [CrossRef]

68. Kothari, N.H.; Tripathi, P.; Bhalja, B.R.; Pandya, V. Support Vector Machine Based Fault Classification and Faulty Section identification Scheme in Thyristor Controlled Series Compensated Transmission Lines. In Proceedings of the 2020 IEEE-HYDCON, Hyderabad, India, 11–12 September 2020. [CrossRef]

69. Phong, B.H.; Hoang, T.M.; Le, T.L. A new method for displayed mathematical expression detection based on FFT and SVM. In Proceedings of the 2017 4th NAFOSTED Conference on Information and Computer Science, Hanoi, Vietnam, 24–25 November 2017. [CrossRef]

70. Karmacharya, I.M.; Gokaraju, R. Fault Location in Ungrounded Photovoltaic System Using Wavelets and ANN. *IEEE Trans. Power Deliv.* **2018**, *33*, 549–559. [CrossRef]

71. Ren, Y.; Yu, D.; Li, Y. Research on Causes of Transmission Line Fault Based on Decision Tree Classification. In Proceedings of the 2020 IEEE/IAS Industrial and Commercial Power System Asia (I&CPS Asia), Weihai, China, 13–15 July 2020; pp. 1066–1070. [CrossRef]

72. Fonseca, G.A.; Ferreira, D.D.; Costa, F.B.; Almeida, A.R. Fault Classification in Transmission Lines Using Random Forest and Notch Filter. *J. Control Autom. Electr. Syst.* **2021**, *33*, 598–609. [CrossRef]

73. Wang, Y.; Liu, M.; Bao, Z. Deep learning neural network for power system fault diagnosis. In Proceedings of the 2016 35th Chinese Control Conference (CCC), Chengdu, China, 27–29 July 2016; pp. 6678–6683. [CrossRef]

74. Cordoni, F.; Bacchiega, G.; Bondani, G.; Radu, R.; Muradore, R. A multi–modal unsupervised fault detection system based on power signals and thermal imaging via deep AutoEncoder neural network. *Eng. Appl. Artif. Intell.* **2022**, *110*, 104729. [CrossRef]

75. Li, T.; Zhao, H.; Zhou, X.; Zhu, S.; Yang, Z.; Yang, H.; Liu, W.; Zhou, Z. Method of Short-Circuit Fault Diagnosis in Transmission Line Based on Deep Learning. *Int. J. Pattern Recognit. Artif. Intell.* **2022**, *36*, 2252009. [CrossRef]

76. Khodayar, M.; Liu, G.; Wang, J.; Khodayar, M.E. Deep learning in power systems research: A review. *CSEE J. Power Energy Syst.* **2021**, *7*, 209–220. [CrossRef]

77. Guo, Y.; Pang, Z.; Du, J.; Jiang, F.; Hu, Q. An Improved AlexNet for Power Edge Transmission Line Anomaly Detection. *IEEE Access* **2020**, *8*, 97830–97838. [CrossRef]

78. Wang, M.; Chai, W.; Xu, C.; Dong, L.; Li, Y.; Wang, P.; Qin, X. An edge computing method using a novel mode component for power transmission line fault diagnosis in distribution network. *J. Ambient. Intell. Humaniz. Comput.* **2020**, *13*, 5163–5176. [CrossRef]

79. Faheem, M.; Shah, S.; Butt, R.; Raza, B.; Anwar, M.; Ashraf, M.; Ngadi, M.; Gungor, V. Smart grid communication and information technologies in the perspective of Industry 4.0: Opportunities and challenges. *Comput. Sci. Rev.* **2018**, *30*, 1–30. [CrossRef]

80. Kotur, D.; Stefanov, P. Optimal power flow control in the system with offshore wind power plants connected to the MTDC network. *Int. J. Electr. Power Energy Syst.* **2019**, *105*, 142–150. [CrossRef]

81. Forstel, L.; Lampe, L. Grid diagnostics: Monitoring cable aging using power line transmission. In Proceedings of the 2017 IEEE International Symposium on Power Line Communications and its Applications (ISPLC), Madrid, Spain, 3–5 April 2017. [CrossRef]

82. Prasad, G.; Lampe, L. Full-Duplex Power Line Communications: Design and Applications from Multimedia to Smart Grid. *IEEE Commun. Mag.* **2020**, *58*, 106–112. [CrossRef]

83. Malik, H.; Iqbal, A.; Yadav, A.K. (Eds.) *Soft Computing in Condition Monitoring and Diagnostics of Electrical and Mechanical Systems*; Springer: Singapore, 2020. [CrossRef]

84. Shagiev, R.I.; Karpov, A.V.; Kalabanov, S.A. The Diagnostics System for Power Transmission Lines of 6/10 kV. *Appl. Mech. Mater.* **2014**, *666*, 138–143. [CrossRef]

85. Passerini, F.; Tonello, A.M. Analog Full-Duplex Amplify-and-Forward Relay for Power Line Communication Networks. *IEEE Commun. Lett.* **2019**, *23*, 676–679. [CrossRef]

86. Djordjevic, M.D.; Vracar, J.M.; Stojkovic, A.S. Supervision and Monitoring System of the Power Line Poles Using IIoT Technology. In Proceedings of the 2020 55th International Scientific Conference on Information, Communication and Energy Systems and Technologies (ICEST), Niš, Serbia, 10–12 September 2020. [CrossRef]

87. van de Kaa, G.; Fens, T.; Rezaei, J.; Kaynak, D.; Hatun, Z.; Tsilimeni-Archangelidi, A. Realizing smart meter connectivity: Analyzing the competing technologies Power line communication, mobile telephony, and radio frequency using the best worst method. *Renew. Sustain. Energy Rev.* **2019**, *103*, 320–327. [CrossRef]

88. Petchrompo, S.; Li, H.; Erguido, A.; Riches, C.; Parlikad, A.K. A value-based approach to optimizing long-term maintenance plans for a multi-asset k-out-of-N system. *Reliab. Eng. Syst. Saf.* **2020**, *200*, 106924. [CrossRef]

89. Fernández, P.M.G.; López, A.J.G.; Márquez, A.C.; Fernández, J.F.G.; Marcos, J.A. Dynamic Risk Assessment for CBM-based adaptation of maintenance planning. *Reliab. Eng. Syst. Saf.* **2022**, *223*, 108359. [CrossRef]

90. Hu, V.C.; Kuhn, D.R.; Ferraiolo, D.F. Access Control for Emerging Distributed Systems. *Computer* **2018**, *51*, 100–103. [CrossRef]

91. Eze, K.G.; Akujuobi, C.M. Design and Evaluation of a Distributed Security Framework for the Internet of Things. *J. Signal Inf. Process.* **2022**, *13*, 1–23. [CrossRef]

92. Duan, L.; Sun, C.A.; Zhang, Y.; Ni, W.; Chen, J. A Comprehensive Security Framework for Publish/Subscribe-Based IoT Services Communication. *IEEE Access* **2019**, *7*, 25989–26001. [CrossRef]

93. Krishnadas, G.; Kiprakis, A. A Machine Learning Pipeline for Demand Response Capacity Scheduling. *Energies* **2020**, *13*, 1848. [CrossRef]

94. Berre, A.J.; Tsalgatidou, A.; Francalanci, C.; Ivanov, T.; Pariente-Lobo, T.; Ruiz-Saiz, R.; Novalija, I.; Grobelnik, M. Big Data and AI Pipeline Framework: Technology Analysis from a Benchmarking Perspective. In *Technologies and Applications for Big Data Value*; Springer International Publishing: Berlin/Heidelberg, Germany, 2022; pp. 63–88. [CrossRef]

95. Giordano, D.; Pastor, E.; Giobergia, F.; Cerquitelli, T.; Baralis, E.; Mellia, M.; Neri, A.; Tricarico, D. Dissecting a data-driven prognostic pipeline: A powertrain use case. *Expert Syst. Appl.* **2021**, *180*, 115109. [CrossRef]
96. Aoufi, S.; Derhab, A.; Guerroumi, M. Survey of false data injection in smart power grid: Attacks, countermeasures and challenges. *J. Inf. Secur. Appl.* **2020**, *54*, 102518. [CrossRef]
97. Yaacoub, J.P.A.; Salman, O.; Noura, H.N.; Kaaniche, N.; Chehab, A.; Malli, M. Cyber-physical systems security: Limitations, issues and future trends. *Microprocess. Microsyst.* **2020**, *77*, 103201. [CrossRef] [PubMed]
98. Omerovic, A.; Vefsnmo, H.; Erdogan, G.; Gjerde, O.; Gramme, E.; Simonsen, S. A Feasibility Study of a Method for Identification and Modelling of Cybersecurity Risks in the Context of Smart Power Grids. In Proceedings of the 4th International Conference on Complexity, Future Information Systems and Risk, Heraklion, Greece, 2–4 May 2019; SCITEPRESS—Science and Technology Publications: Setúbal, Portugal, 2019. [CrossRef]
99. Huo, Y.; Prasad, G.; Lampe, L.; Leung, V.C.M. Advanced Smart Grid Monitoring: Intelligent Cable Diagnostics using Neural Networks. In Proceedings of the 2020 IEEE International Symposium on Power Line Communications and its Applications (ISPLC), Malaga, Spain, 11–13 May 2020. [CrossRef]
100. Aligholian, A.; Farajollahi, M.; Mohsenian-Rad, H. Unsupervised Learning for Online Abnormality Detection in Smart Meter Data. In Proceedings of the 2019 IEEE Power & Energy Society General Meeting (PESGM), Atlanta, GA, USA, 4–8 August 2019. [CrossRef]
101. Serradilla, O.; Zugasti, E.; Rodriguez, J.; Zurutuza, U. Deep learning models for predictive maintenance: A survey, comparison, challenges and prospects. *Appl. Intell.* **2022**, *52*, 10934–10964. [CrossRef]
102. Huo, Y.; Prasad, G.; Lampe, L.; Leung, V.C.M. Cable Health Monitoring in Distribution Networks using Power Line Communications. In Proceedings of the 2018 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm), Aalborg, Denmark, 29–31 October 2018. [CrossRef]
103. Xi, P.; Feilai, P.; Yongchao, L.; Zhiping, L.; Long, L. Fault Detection Algorithm for Power Distribution Network Based on Sparse Self-Encoding Neural Network. In Proceedings of the 2017 International Conference on Smart Grid and Electrical Automation (ICSGEA), Changsha, China, 27–28 May 2017. [CrossRef]
104. Chen, W.; Zheng, L. Q-Learning Algorithm Based Topology Control of Power Line Communication Networks. In Proceedings of the 2020 IEEE 11th International Conference on Software Engineering and Service Science (ICSESS), Beijing, China, 16–18 October 2020. [CrossRef]
105. Bifet, A.; Gavaldà, R. Learning from Time-Changing Data with Adaptive Windowing. In Proceedings of the 2007 SIAM International Conference on Data Mining, Minneapolis, MN, USA, 26–28 April 2007; Society for Industrial and Applied Mathematics: Philadelphia, PA, USA, 2007. [CrossRef]
106. Specification of Sagemcom T211 Smart Power Meter. Available online: https://www.sagemcom.com/sites/default/files/datasheet/Datasheet%20T211_1121.pdf (accessed on 24 September 2022).
107. Mosquitto Message Broker Software. Available online: https://mosquitto.org/ (accessed on 24 September 2022).