energies

MDPI

Article

A Real-Time Digital Twin and Neural Net Cluster-Based Framework for Faults Identification in Power Converters of Microgrids, Self Organized Map Neural Network

Juan R. Lopez [1], Jose de Jesus Camacho [1], Pedro Ponce [1,*], Brian MacCleery [2] and Arturo Molina [1]

[1] Tecnologico de Monterrey, School of Engineering and Sciences, Puente 222, Tlalpan, Mexico City 14380, Mexico
[2] National Instruments Corporation, Austin, TX 78759, USA; brian.maccleery@ni.com
* Correspondence: pedro.ponce@tec.mx

Abstract: In developing distribution networks, the deployment of alternative generation sources is heavily motivated by the growing energy demand, as by environmental and political motives. Consequently, microgrids are implemented to coordinate the operation of these energy generation assets. Microgrids are systems that rely on power conversion technologies based on high-frequency switching devices to generate a stable distribution network. However, disrupting scenarios can occur in deployed systems, causing faults at the sub-component and the system level of microgrids where its identification is an economical and technological challenge. This paradigm can be addressed by having a digital twin of the low-level components to monitor and analyze their response and identify faults to take preventive or corrective actions. Nonetheless, accurate execution of digital twins of low-level components in traditional simulation systems is a difficult task to achieve due to the fast dynamics of the power converter devices, leading to inaccurate results and false identification of system faults. Therefore, this work proposes a fault identification framework for low-level components that includes the combination of Real-Time systems with the Digital Twin concept to guarantee the dynamic consistency of the low-level components. The proposed framework includes an offline trained Self Organized Map Neural Network in a hexagonal topology to identify such faults within a Real-Time system. As a case study, the proposed framework is applied to a three-phase two-level inverter connected to its digital model in a Real-Time simulator for open circuit faults identification.

Keywords: digital twin; real-time; microgrid; machine learning; fault identification

check for updates

Citation: 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. https://doi.org/10.3390/en15197306

Academic Editor: Umberto Berardi

Received: 17 June 2022
Accepted: 26 September 2022
Published: 4 October 2022

1. Introduction

Current environmental concerns in the energy generation sector have led to the rapid deployment of distributed energy sources, mainly in the form of alternative generation systems based on renewable resources. The increasing numbers in distributed generation have created paradigms in the control, coordination, management, and operation areas of such assets. This has motivated research interest towards microgrid systems as entities within distribution systems capable of coordinating the distributed generation with management and regulation capabilities. Since microgrids can be seen as a single entity within the electrical network, the quality control and management paradigms behind the distributed generation of electrical energy are simplified. In this sense, the how, when, and where the energy is consumed or generated, and the protection challenges that follow the physical and digital layers that compose its architecture can be handled in a simpler manner [1].

The right integration of solutions to the individual paradigms that fall into the many research categories that follow the microgrid topic can help to achieve a functional and reliable implementation of a microgrid system. Currently, microgrids are deployed in two formats, grid-connected and autonomous or island systems. The former functions as a complimentary service that contributes to the overall power demand while locally feeding a set of connected loads, the latter format being an independent system capable of self-managing and self-balancing the power generation and consumption rates to fully cover for electricity demand of isolated or energy-deprived communities while regulating electrical variables such as the microgrid voltage and frequency [2].

Microgrids are systems that require extensive studies and tests before their deployment to avoid any shortcomings in their performance due to unexpected scenarios that can even result in system faults that compromise the integrity of the generation and consumption assets connected to the distribution system [1]. These unexpected scenarios can be described as anomalies in the expected operating conditions of microgrid systems, such anomalies can be a miss prediction of weather conditions, overload conditions due to a higher power demand, and unexpected changes in the operation mode of the microgrid, causing difficulties in the decision-making process of Energy Management Systems (EMS) and in low-level power converter devices to correctly assess the derived problematics.

These scenarios can trigger faults at any of the three operation levels of microgrids according to the hierarchical control architecture described in the literature [3,4]. Faults that occur at the tertiary level commonly deal with the management of resources and interrupted communication channels between the central controller and the governed assets. Faults at the secondary level affect the power-sharing capabilities among the generation units, causing a mismatch of delivered active and reactive power by the governed generation units and leading to harsh circulating reactive power by all of the generation agents and inducing instability in the distribution system [5]. Lastly, faults at the primary level are mainly found in power converter systems such as DC to DC, AC to DC, and DC to AC converters; these types of faults are technically more challenging since these can occur at the component and system levels where its identification or amendment can become costly and in a difficult process, especially in multi-level converter topologies. Component level faults deal with corrupt switching devices or other sub-components in the systems such as the associated diodes and DC link capacitors; component faults are mainly attributed to over-current and over-heating conditions that hinder the performance and have a negative impact on the expected lifespan of the semiconductors within the power converter system.

Open circuit faults can be classified in single phase, two-phase, or three phase. The basic type is the "single phase", and the rest of the open circuit faults are derived from it. This mainly occurs because one of the wires of the transmission line gets open (this can happen due to joint failure or broken conductor) and precludes the current from flowing through the wires. The effects that an open circuit fault can have are: (i) unbalanced load current; (ii) abnormal voltage and current behavior; (iii) danger to individuals surrounding the grid; and (iv) complete system shutdowns [6].

System shutdowns compromise the integrity of the microgrid and its assets; for that reason, the topics of fault detection and identification have gained popularity in the past decades. Mainly three categories for fault detection in switching power converters can be obtained from the literature: sub-component, system-based, and model-based. Sub-component techniques are known for being embedded in the power converter system and monitoring the gate signal of the switching devices for over-current and overheating conditions. The second category can be further divided into waveform analysis and algorithmic techniques [6]. System fault detection through waveform techniques operates by analyzing the voltage and current waveforms and comparing them against nominal system indicators to determine the fault type and location; however, this type of fault detection technique is known for being sensitive to unbalanced load conditions and large load switching scenarios, causing false tripping signals.

On the other hand, algorithmic strategies are implemented along with classifier algorithms such as neural networks, fuzzy logic, logistic regression, and decision trees, where their performance depends on the training procedure and the data obtained to train the classifiers. The latter model-based technique compares system measurement to estimated values according to a defined simulated model implemented in an offline system, the accuracy of model-based techniques depends highly on the simulated digital model and employed simulation tool to match the fast dynamics of the power systems, where an inaccurate or an inconsistent dynamic representation of the system would result in an erroneous fault diagnostic [7]. Thankfully, in recent years, the validation of microgrid and power systems technologies has been heavily supported by Real-Time (RT) simulation technologies that ensure the dynamic consistency of the proposed systems. According to Ibarra et al. [8], RT systems offer a safe path toward the simulation, test, and validation of complex and dangerous systems against novel hardware or software proposals, minimizing risks for the user and physical systems when dealing with high voltage or high current power systems. In addition, RT is an attractive tool for the expansion of cost-compelling technologies and solutions by closing the gap between the initial concept and its implementation or rejection.

As an example in the application of RT simulation in the power and energy sector, Lu et al. [9] proposes a low-cost RT Hardware-in-the-loop (HIL) system to test and validate the performance of power electronic controllers in applications that include a DC-DC boost converter and single-phase inverter systems; overall, the proposed RT system proposed by Lu et al. [9] achieves a good resolution to ensure the dynamic consistency of most power electronic devices. In a different power systems application, Ref. [10] uses an HIL setup to validate their proposal on a decentralized islanding detection algorithm for microgrid systems, the proposed HIL platform ensures the dynamic consistency of a microgrid equivalent model while the proposed decentralized island detection algorithm is embedded in a separate RT platform and running at its own step time. In general, the proposed RT validation setup ensures the dynamic consistency of the islanding detection proposal and the simulated microgrid equivalent model.

In the power systems fault detection landscape, the work presented by Poon et al. [11] uses an RT validation approach in their proposal of model-based fault detection and identification algorithm for switching power converters, such algorithm is based on a state estimator to identify faults related to components and system sensors, the fault identification process is carried out by comparing the measured voltage from the system against an estimated voltage. Overall, the state estimator proposal is validated through an HIL setup, demonstrating its capability to identify component level faults according to the programmed model of the power converter model, providing a generic approach toward fault identification. Particularly, the proposed model-based method identifies open circuit faults in inverter-based systems within 3.45 ms.

In a different approach, the work reported by Estima and Marques Cardoso [12] successfully identifies open circuit faults in an inverter based system, the proposed system-based technique takes into consideration the average absolute values of the measured current to create a threshold that can identify the occurrence of open circuit faults with in 1.13 ms, additionally, Estima and Marques Cardoso [12] also integrates an HIL strategy to validate their proposal. Nonetheless, despite being validated in an HIL setup, such implementations are intended to be embedded within the control system of the power converters which can fall short of the required computational power to guarantee the dynamic consistency of the configured power converter model and fail as a fault identification distributed approach. Real-time simulation has been used to give a better approach when HIL is integrated because it is possible to obtain a real time simulation running with the same dynamic response in real time as the experimental system. Thus, several hardware elements could be evaluated in the same system. On the other hand, conventional simulation and co-simulation have been shown to provide a good response when comparing offline against the experimental system. In addition, conventional simulation is the first step to evaluating HIL in real-time simulation. As a result, both conventional and real time simulation could provide excellent results when offline simulations are evaluated.

Despite the efforts of having a proposal validated through RT simulation, model-based fault identification methods still require accurate models and platforms with the capacity to accurately replicate the real-life dynamics of these complex systems. To overcome such limitations, new digital representation technologies have emerged across the literature, where now the digital representation of a system is referred to as a Digital Twin (DT) [13,14]. Conventionally, the DT process is carried out in a digital simulation environment with real data as the input of the system; in this environment, the modeling, prediction, testing, and validation of a given scenario and problem are obtained rapidly to effectively send any correcting actions to the real system through a feedback channel. Nonetheless, little is known about the development of DTs and how the design process can affect the outcome of this digital counterpart. In such understanding, the work presented by Solman et al. [15] reviews the different actors that are included in the planning and deployment of DTs, the work concludes with a list of five key areas to take into consideration when planning a DT related to problematics in the energy generation sector: (i) energy generation system design, (ii) data to be used, (iii) define constraints of public engagement, (iv) simplification and selection of data according to energy-related challenges, and (v) select attributes to be digitized.

Within the energy landscape, DTs have emerged as a technological solution to energy management and operation paradigms. In the work presented by Jafari et al. [16], it is stated that DTs aid building owners and operators in predictive failure and fault diagnosis, execute what-if scenarios to analyze the system response to parametric or operational parameters, and carry out planning and resource analysis for performance optimization; overall, such work proposes a novel DT architecture for improved building energy management that results in a more sustainable system thanks to the intercommunication between business models, analytic engines, with the digital counterpart. A similar study is conducted by Onile et al. [17], where DT is explored as a tool for improving the smart management of energy generation according to demand side management strategies related to consumer behavior to predict and improve energy-related services. However, it is important to understand that this digital validation process is not always the best alternative in systems with a high degree of complexity, mainly because the conventional simulation environment can be a reason to have unmatched dynamics of the interfaced agents and result in unfeasible solutions.

An unfeasible solution could potentially lead to a failure of the real system even with the application of DTs in power systems. Therefore, this work proposes a framework for fault identification in a power system that encloses the DT in an interfaced distributed generation unit within an RT environment to avoid erroneous DT results due to inconsistent dynamics. In the proposed framework, fault identification is achieved by merging the operation of an algorithmic and model-based identification strategy. A Self-Organizing-Map (SOM) Neural Network is implemented and trained to detect and identify power system faults, while a DT model of the interfaced power converter is assembled for comparison between a real system and its digital replica. The proposed framework is validated in a case study for three-phase and single-phase open circuit faults in a three-phase inverter-based system.The main contribution of our work can be summarized as a fault identification framework for a three-phase inverter system, where the principles of DTs, RT simulation, and ML are combined to generate a simulation environment that can cope with the fast dynamics of power systems and generate an adequate platform for implementing online fault detection based on an ML clustering technique, resulting in a reliable fault identification tool that works in parallel with an experimental system.

The remainder of this work is structured as follows; Section 2 shows the DT concept in the microgrid scheme; Section 3 elaborates more on RT simulation as a validation strategy for microgrid technologies; Section 4 gives a brief introduction to the application of Machine Learning in the microgrid scope; Section 5 gives the description on the proposed fault identification framework; Section 6 provides details on the case study used to validate this work's proposal; Section 7 gives the experimental results and Section 8 opens the discussion and future trends of DT technologies in the power system landscape. Lastly, Section 9 gives the conclusions of this work.

2. Digital-Twin Concept in the Microgrid Scheme

In the electric sector, as in different industries, many decision-making actions are taken thanks to the acquired result from a digital model. Commonly, the data fed into the digital model are obtained via extensive pre-studies that analyze the past behavior of consumers, utilities, industries, or environments. For the microgrid, these data points are represented by the fluctuation analysis of the energy market, time-of-use tariff profiles, historical weather information, just to mention a few. These data are later used to formulate a model that can help in the decision-making task of managing the microgrid system more efficiently.

Nonetheless, this approach relies heavily on the recorded data during past events. This means that the recorded data are directly correlated to the system for which it was obtained. This statement indicates that any parametric changes in the real system can affect the predicted outcome of the model. For that matter, emerging technologies such as the DT make use of running system data to predict the behavior of the real system. The DT approach minimizes the dependency on past modeling and prediction methods. Allow a tailored management experience according to the ongoing scenarios.

Essentially, the objective of a DT is to develop a digital reproduction of a physical system. DTs are capable of mimicking the dynamics and functionality of their physical counterparts. NASA defines the DT as a multiphysics, multiscale, probabilistic simulation of a physical system that makes use of accurate physical models, sensor data, and performance history to resemble the outcome of its real equivalent [18].

According to the literature review, a DT can be constituted from three to five main areas. For example, the work presented in [19] defines the DT as a constitution of five elements: (i) physical system, (ii) virtual environment, (iii) provided service, (iv) acquired data, and (v) communication link between the real and the virtual environment. On the other hand, Wang et al. [20] clarify that, by following the NASA definition, the DT concept can be formed in only three dimensions, the physical product, the virtual model, and the bi-directional communication link in between the two. In the end, the convolution of these key areas helps to define a monitoring scheme capable of providing valuable data regarding a real system; this contributes to the early detection of anomalies and complexities, while improving the forecast of operational conditions.

In another study presented in [21], it is stated that the DT can have several definitions depending on the industry of application. The DT can be segregated in: (i) conventional; (ii) prototype; (iii) instance; and (iv) environment. Regardless of these partitions, the essential objective of the DT remains the same, only changing in the area of application, the DT prototype definition is suited for application in the energy industry. The DT prototype concept is defined as a digital representation of a real artifact, where all the information required to mimic the real implementation is self-contained in the virtual system. By following this definition, the microgrid can be comprehended as the real artifact, all the information of the many components is self-contained within the model and only the running data are fed into the DT to predict the microgrid's behavior. Figure 1 shows a scheme of how the DT definition can be applied to the microgrid.

In the first partition of the microgrid's DT scheme, the active participation of the real system is defined by the collection and sent data, these actions are achieved by an established AMI and the microgrid's EMS, capturing raw sensor data and allowing a seamless flow of information from the many assets to the microgrid's main operator. The sent data contain information regarding the state of the many participating agents, including their consumption and generation readings. Subsequently, the running data are used to create and update the system's model through AI techniques.

AI has been a key factor driving the current development of DT technologies. With the integration of the virtualization process and AI, the structure, context, and dynamics of a real system allow not just the monitoring of past and present operations, but also enable prediction capabilities about future events, leading to self-healing models that adapt to their real twins [22]. Once the data-driven analytics stage is complete, the output data are then simulated in a physics-based model of the real system. In this stage, computational simulation is commonly the main platform for which a given scenario is validated. This leads to the final stage, where the validated outcome is fed back into the real system, closing the communication loop between the real implementation and its digital counterpart.

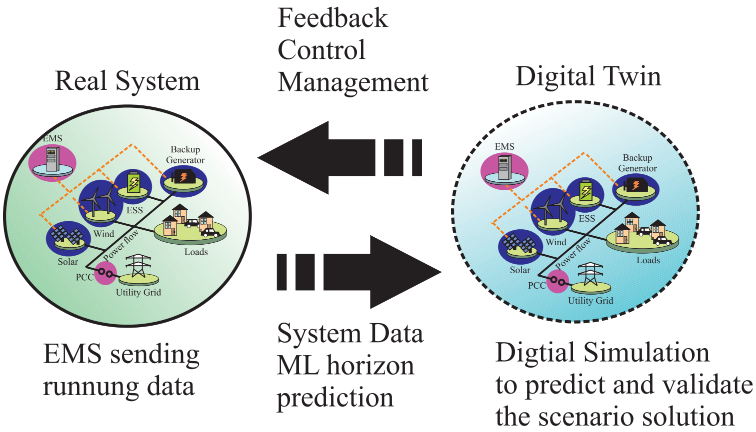

Figure 1. Digital twin microgrid scheme.

Overall, the application of DT technologies to the microgrid's operation opens the door for a tailored management experience in key opportunity areas that help in the performance and sustainability of the small-scale distributed electrical network: the optimization of its performance, prediction of preventive and corrective maintenance, planning the expansion development in generation and consumption capacity, and optimization of urban sustainability [22]. Indeed, all of these features are enabled through DT technology. However, due to the dynamic complexity and constant expansion of power systems, traditional validation methods such as computational simulation can hinder DT performance. For that matter, other validation tools can be adopted to improve the validation stage of the DT scheme in complex systems such as the microgrid.

3. Real-Time Simulation as a Validation Structure

Traditionally, the validation process within the DT concept is carried out in an online digital simulation environment. Although this validation strategy can be sufficient for other processes in the manufacturing manufacturing area, the challenge that comes with the complex dynamics of power systems requires more effective validation strategies. RT simulation has been gaining popularity as a validation tool to overcome the dynamic complexity of power systems and offering a high degree of fidelity as studied by Sekine et al. [23] in 1994.

A central characteristic of RT simulation is the capability of offering a resembling approximation to the real system dynamics by virtue of guaranteeing the dynamic consistency of the simulated model. Dynamic consistency is an attribute formed by the task to be done and the time-step (Ts) of the simulation, essentially, to achieve the dynamic consistency, the tasks to be done need to start at a deterministic Ts and finish before the next Ts begins [24]. However, in traditional simulation environments, where the TS and the task to be done do not follow this deterministic behaviour. In example, discrete simulation operates with a fixed TS; however, the task of the simulation can start and/or end at any giving time of the simulation, meaning that these can be completed after or before the next fixed Ts. Similarly, continuous simulation adapts the Ts according to the task at hand; this results in a simulation that does not have a deterministic Ts, therefore the dynamic consistency of the simulated model can not be guaranteed. A comparison of these simulation environments is illustrated in Figure 2.

As mentioned in a previews section, RT simulation is a tool used in many engineering fields, where many industries and research groups take good advantage of this simulation technology by testing and simulating control, software, and high-power level management strategies without compromising the integrity of the personnel and/or real equipment. In addition, RT opens the door to new validation strategies such as Co-simulation, HIL, Hardware-Under-test (HUT), and Power-Hardware-in-the-Loop (PHIL) simulation methods, each with its unique characteristics and favored areas of application.

To begin, the co-simulation strategy aids to reduce the complexity of simulations that require the contribution of multiple systems for the resolution of a given model. This strategy divides core parts of given problematic and combines multiple simulation platforms to generate a holistic test environment. Essentially, the system of interest is segregated into a finite number of core modules, then each module is simulated in single but interconnected, simulation environments. As an example, Zhang et al. [25] presents a co-simulation environment that validates the right interaction of multiple power systems before the actual implementation; such proposal divides the entire system in three modules, a hybrid power grid, a 500 KV transformer, and the electromagnetic transients and finite analysis calculation-model. In such arrangement, the validation of such systems is achieved without the direct interaction with real and industrial equipment.

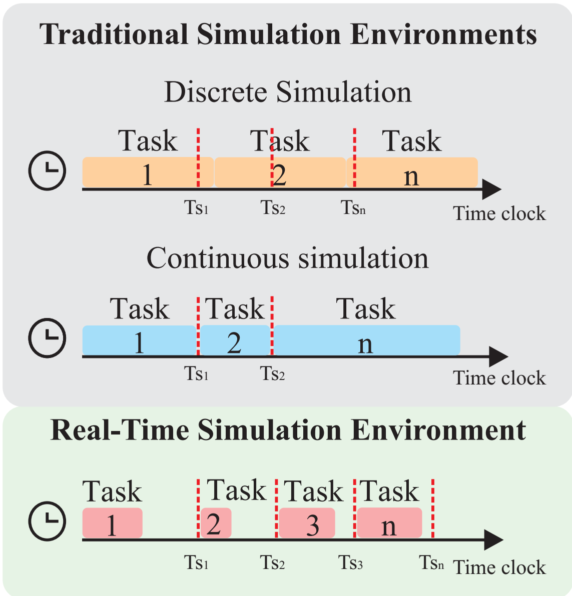

Figure 2. Types of simulation environment.

As a different approach, HIL adopts embedded systems to test and validate novel control or software proposals. In an HIL strategy, the end product is downloaded to an embedded system and interfaced via bidirectional communication channels with the RT simulation environment where the plant or physical system is being simulated. In doing so, the embedded system is assured to act over the real-dynamics of their intended target, thus the validation of its operation is achieved before it actual implementation. According to Bélanger and Venne [24], HIL allows for more repeatable testing results; since the dynamics of the simulated system remain the same across the entire simulation, unlike in physical systems, the dynamic response could change. HIL strategy helps to accelerate the integration and verification stages within the development process of an idea, something that is hardly achieved by offline simulation [24].

Lastly, PHIL is defined by Omar Faruque et al. [26] as any HIL setup where power transfer is done to or from a hardware unit, this is achieved by the implementation of power amplifiers or sensors. For other authors, a PHIL strategy requires a power amplifier stage and a sensing stage, to properly close the loop between the hardware and simulation interface [8]. PHIL is used more in power systems since this strategy includes a power exchange layer between tested hardware and the RT simulation environment. This strategy is preferred to test and validate novel control or management proposal in high-power systems, as it offers a safe simulation space for the users and the equipment. Overall, these strategies contribute to the early detection of faults and reduce the need for extensive testing.

In this work, command signals are sent to the inverter, and a model is running in real time simulation, so the real time model and the power electronics stage are interacting in real time. This is considered an HIL. For instance, sending command signals to the inverter through a DSP is not considered an HIL since no real-time digital model is running into the real time simulation. In addition, real time simulation solves the model according to the real time dynamics of the experimental system to be connected with the hardware. Thus, the model running in the real time simulation can maintain the same dynamic response so that repeatable testing can be conducted. In the experimental system, parameters and initial conditions could change, so repeatable testing is difficult to accomplish. In any case, the HIL strategy fulfills the requirement of having an RT simulation that can emulate and validate such test cases with high fidelity. This presents simulation models that can be monitored as DT. In this work, RT simulation creates an adequate environment for the generation of DT of complex power systems.

4. Machine Learning in the Microgrid Landscape

We are now at a stage where technology is able to take advantage of the advances made in artificial intelligence algorithms which allow faster convergence, manipulating massive amounts of data, and solving increasingly more complex problems. One of the methods of artificial intelligence employed on the microgrid is Machine Learning. According to Dangeti [27], Machine Learning (ML) is defined as the branch of computer science that utilizes experience to learn from and use its knowledge to make future decisions. The goal of ML is to generalize a detectable pattern or to create an unknown rule from given examples.

ML has been used several times at the academic and enterprise level given that it has shown its usefulness by extracting meaningful data from raw environmental interaction. One of the keys to using ML is selecting an appropriate algorithm for the task at hand (classification, detection, prediction, optimization, etc.); this affects the ability to predict and accurately divide into groups the data fed to the system. Figure 3 shows different categories for training machine learning systems and some of the most commonly used algorithms.

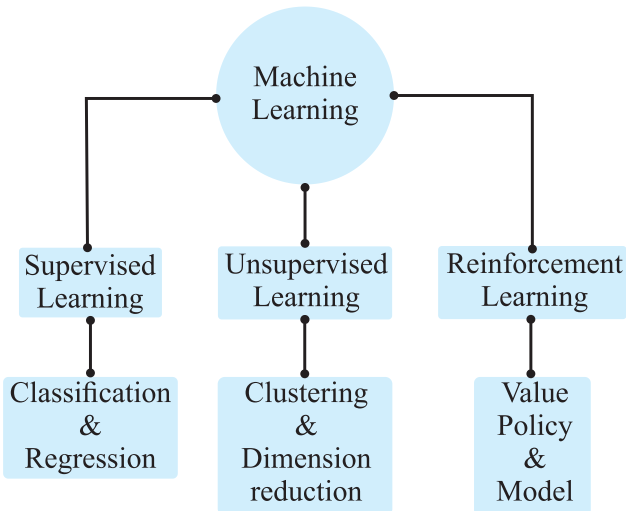

Figure 3. Categories of machine learning and commonly used classification & clustering algorithms.

Each algorithm has its own application scope and can not be used to solve every problem; Tables 1 and 2 show some of the advantages and disadvantages of said algorithms.

For the specific case study of this work (microgrids), some examples of the use of ML are reported by Almutairy and Alluhaidan [28] and Ali et al. [29]. In the former case [28], a fault diagnosis framework becomes feasible with the inclusion of ML. In such an article, a DC microgrid is introduced as a case study where an AC grid is also connected through a rectifier; in addition, photovoltaic panels, battery storage, and loads are all connected to the main DC bus. An artificial Neural Network (ANN) is trained with data for the line changing current by analyzing the current slope angle, thus determining if a short circuit occurred. The ANN is trained to recognize the angle at which a fault is produced (near 90°. This ANN is then connected to a circuit breaker to isolate the segment of the Microgrid producing the fault, avoiding further damage to the grid.

In the latter example of ML application [29], it is demonstrated that microgrid problematics such as islanding events identification can also be addressed through the use of machine learning as proposed by Ali et al. [29]. Islanding detection is vital on the microgrid for efficiency rising and according to *IEEE 1547 standard*; this scenario should be detected within 2 s. This study shows an ANN trained offline with data gathered by multiple simulations of islanding scenarios in *MATLAB Simulink*. ML is used in this scenario given that the threshold has a sweet spot that this algorithm was able to learn, if this threshold is not accurately spotted, it can produce a non-detection zone for islanding scenarios. The goal of achieving islanding detection using ML was achieved within 100 ms and without any nondetection zones.

Self Organizing Map (SOM) has multiple applications as shown in [30] for profiling and forecasting at the microgrid level and [31] for load forecasting. SOM is a powerful clustering algorithm, where surrounding neighborhood data can belong to one neighboring cluster or a close one. It was introduced by Kohonen [32] inspired by biological processes. It models the plasticity of the connections in the brain, where the neural connections either get strengthened or disappear while learning.

In an SOM or Kohonen network, for a map that has I units, each unit compares its weight vector to the input vector $x(t)$. Through an iterative process, each weight gets altered every epoch, and these changes are dependent on the similarities between the neighborhoods (how close the input pattern is to the map pattern) [33]. This is measured by calculating the Euclidean distance from x to all weight vectors. The neuron with values closer to the input vector ones is called the Best Matching Unit (BMU). As listed in the Advantage for Self Organizing Map in Table 2, real-time implementation is possible with this algorithm. As the BMU is found, the update process for the weights vectors gets started, and the BMU is re-arranged in a way that gets closer to the input vector. In the following sections, it will be shown how it can be implemented for fault detection on a DT scheme for the microgrid application.

Table 1. Supervised methods in ML [34].

Supervised Method	Advantages	Disadvantages
Artificial Neural Network	• Parallel processing capability. • Can be used on noisy data and data-sets on which they have not been trained. • Well tested on real scenarios.	• It is better to train them offline as long training time is required. • Difficulty to interpret the results, as the learning phase is represented by connection weights.
Decision Trees	• Easy to comprehend. • Fast decision-making. • Good accuracy.	• Long training time. • When dealing with large amounts of data, memory availability can be an issue. • Most of the decision trees algorithms do not deal well with diagonal partition problem.
Naïve Bayes	• Unrelated features get discarded, so classification performance increases. • Does not require big computational time to solve.	• Large amount of data are required to train this network. • The accuracy for this method is average.

Table 2. Unsupervised methods in ML [33,35,36].

Un-Supervised Method	Advantages	Disadvantages
Hierarchical Classification	• Particularly useful when data intrinsically has hierarchical relations.	• Not suitable for large data groups, execution time is linearly proportional to the number of input sample. • High sensitivity to atypical values.
K-means	• Easy to implement. • Even with a large quantity of variables, it can be computed faster compared to other algorithms.	• High sensitivity to initial seeds. • Ordering of data fed can have an impact on the results.
Gaussian Mixture Model	• Provides the probability of a sample to be part of a group.	• Large execution time. • Initial seeds can have a large impact on the results. • Data ordering can have an impact on the results.
Self Organizing Map	• Can be implemented on Real-Time Machines. • Less sensitivity to noise present on training samples. • Easy to construe, as results are shown on a map.	• Initial seeds can have a large impact on the results. • Close together samples can be hard to differentiate from a group or the neighboring group.

5. Real-Time Neural Network and Digital Twin-Based Fault Identification Framework

In this section, we present the proposed framework for fault identification in power converters. The framework is divided into three stages: the first stage consists of the physical system section, this section can be configured with any power converter topology, the inputs of this stage are the physical can be the PWM switching signals for the semiconductor, DC link values, or voltage and frequency references from superior management layers. The second stage consists of an HIL data acquisition interface; this stage receives the output system measurement (voltage, current, active power, reactive power, frequency, phase angle) and translates them to compatible analog signals for the RT simulator. The third and last stage consists of the RTDT environment, this stage contains the DT model of the physical system, a comparison stage, and the SOM classification method. In the RTDT environment, the DT model is fed with the same system input signals of the physical system, producing reciprocal output signals that can be directly compared to those of the real replica. Additionally, the RTDT environment also contains the SOM neural network, this classification method is trained offline with offline simulation data and according to the type and number of faults to identify. The described proposal is illustrated in Figure 4.

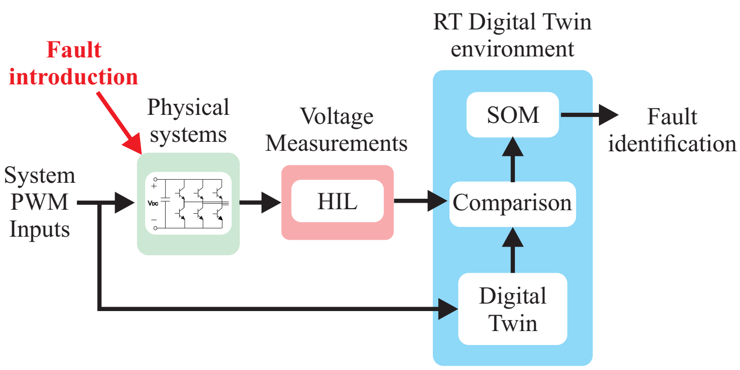

Figure 4. Proposed real-time neural network and digital twin-based fault identification framework.

As mentioned, the SOM stage consists of a previously trained SOM neural network, and the training process is done offline according to the faults of interest for the SOM to identify, meaning that the output of the SOM can be configured to have the desired number of neurons where each neuron represents a particular fault in the system; in addition to the fault conditions, a normal grid output must be considered as the grid may lack fault conditions and operate as intended. Furthermore, the inputs of the SOM are defined as the errors obtained through the comparison of the voltage waveforms of the real and DT system. The error is obtained per phase of the system, meaning there is an error for each of the three phases. The error for the measured voltage in phase A is defined in Equation (1); in such equation, the voltage measurements of the physical system are defined by the variable V_{aRS}; as such, the voltage measurements of the DT are represented by the variable V_{aDT}, the same composition of error equations is assumed for the remaining phases of the system. These voltage errors from the input vector in the SOM architecture, where the Output layer highlights the zones in which a fault can be identified according to the weights of each neuron in the output layer plane. Figure 5 illustrates the SOM architecture for the proposed fault identification framework:

$$\mathbf{e}_{Va} = V_{aRS} - V_{aDT} \tag{1}$$

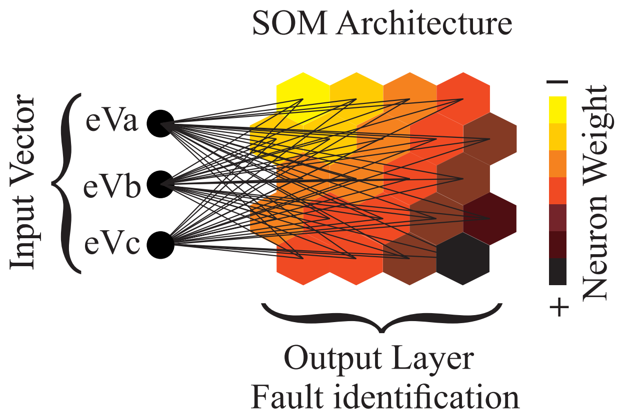

Figure 5. Self Organizing Map architecture for faults identification.

6. Case Study: Fault Identification of a Three-Phase Inverter in a Real-Time Digital Twin Environment

As a case study, we apply the proposed and previously described RT fault identification framework to a three-phase two-level inverter feeding a three-phase restive load (Figure 6). The proposed framework is tested only for open circuit fault since these do not represent a risk for the physical equipment or operator. As a first stage, two models are assembled in offline simulation (with and without faults), these systems are used for the offline training of the SOM neural network. As a second stage, an HIL setup is assembled using hardware from *Imperix*, *Festo*, and *OPAL-RT*. These two stages are detailed in the following subsections.

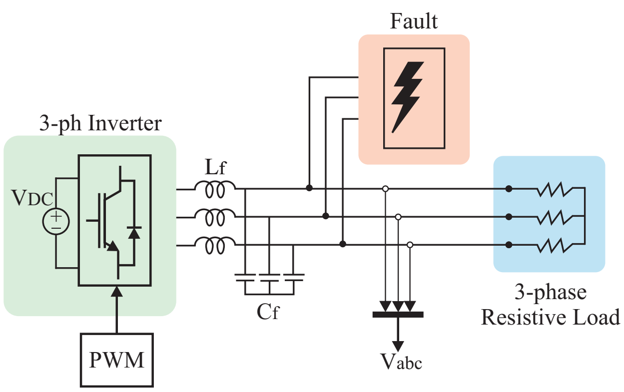

Figure 6. Case study schematic for a three-phase inverter.

6.1. Offline SOM Training

As mentioned, an SOM is also programmed to run in parallel with the DT model in the same RT environment. Such SOM is trained offline to identify the type of fault according to the errors obtained by comparing the measured voltage waveforms per phase. In this work, the training process is carried out by using *MATLAB Simulink* and the *Deep Learning MATLAB Toolbox*. To start with the offline training process, a *MATLAB Simulink* model is created containing two copies of the described system; however, only one copy is interfaced with programmed fault events, while the second one remains in normal grid conditions. It is important to mention that, during the training process, each fault must be induced sequentially; these must not overlap each other and normal grid conditions must be reestablished between fault events. This fault induction process is illustrated in Figure 7.

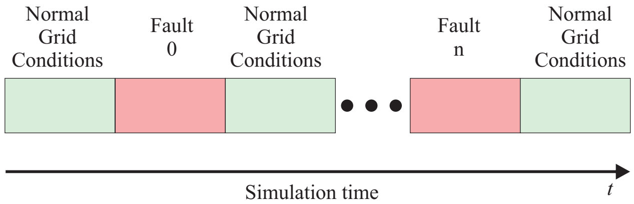

Figure 7. Fault induction sequence.

Since the testing and validation are limited by laboratory protection protocols, only open circuit faults are used to train the SOM neural network; these faults are introduced according to Figure 7 and in the order shown in Table 3.

Table 3. Fault types and labels.

Fault Type	Fault Number
Three-phase open circuit (ABCO)	0
Single phase open circuit (AO)	1
Single phase open circuit (BO)	2
Single phase open circuit (CO)	3

In addition to the programmed fault conditions, the *MATLAB Simulink* model is also configured to compare the voltage wave-forms forms from both systems, comparing healthy voltage wave-forms against faulty ones. The data of the obtained errors are stored and exported to *MATLAB's* work-space as an array. The following step is to create a new network using *MATLAB's* deep learning functions where the input data are defined as three different data sets for each wave-form errors. Since the number of faults is small, training time can be reduced by having a small size of the two-dimensional SOM map; for such reason, the selected dimension is a 4×4 neurons map. In addition to the small network, a second and bigger network is trained to test the capability of a SOM running in RT, the secondary network is 25 times bigger with dimension equal to 20×20 neurons. The training process is carried out multiple times to improve the accuracy of the classification method in each iteration, this training process is shown in Figure 8. Lastly, the trained networks are exported to a *MATLAB Simulink* file for seamless integration into the RT environment.

After the training process, the input planes for each phase are obtained, where the weights from inputs 1, 2, and 3 correspond to the data from phases A, B, and C, respectively. These input planes are shown in Figure 9 and match the SOM map of the small network. These input planes have a clear segmentation of neuron weights according to the given information; then, the identification between a healthy and a faulty phase can be assumed. In addition, the obtained input planes indicate that the inputs are independent of each other, since none of the obtained planes are similar to another.

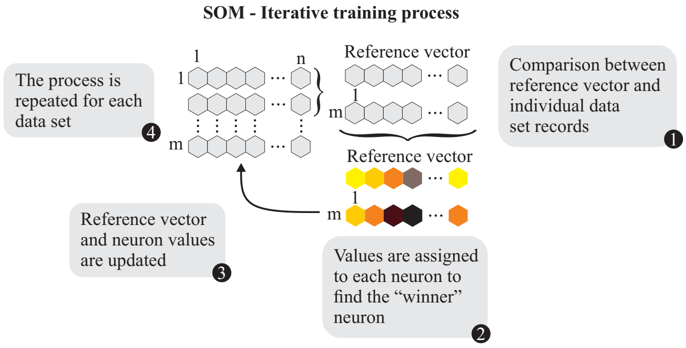

Figure 8. Self Organizing Map (SOM) iterative training (step 1 comparison between reference vector and individual data, step 2 assigned values to get the winner, step 3 vectors are updated, step 4 go to step one until the stop condition is achieved).

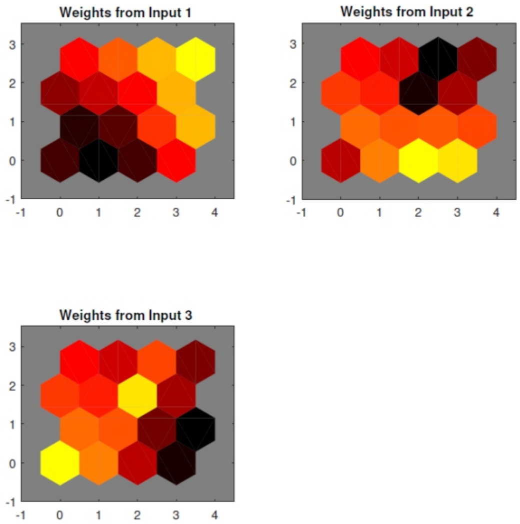

Figure 9. Self Organizing Map (SOM) input planes per phase of the system.

Figure 10 shows the neighbor weight distances between the 16 neurons in the small network, where each neuron is represented by the gray hexagons; the distance between each neuron with similar clusters is illustrated by bright colors, whereas darker colors represent the distance of clusters that is further apart. In a more detailed explanation, the training process starts by having an array of $n \times m$ neurons that constructs a two-layer map, the input layer, and the processing layer, it is then when the neurons of the formed map are assigned with a starting random weight. Once each neuron has an initial weight, an evaluation criterion is used to determine the space between each neuron; this evaluation criterion is mostly selected as the Euclidean space. The space between neurons is then used to select the winning neuron according to the evaluated input vector. After finalizing the selection process, the weights of each neuron are updated, and the distance valuation process starts once again. This process can be carried out according to a given stopping criterion (mean square error, number of iterations, etc...). At the end, the final weight of each neuron is a representation of how these light up when a given input is applied, meaning that inputs can be classified according to a stimulated neuron [37].

As a complement of Figure 10, Figure 11 illustrates the number of hits of the sampling data are obtained. Since the simulation data contain a large number of points that contain information with regard to a healthy grid condition, it is expected for the majority of sampling points to hit a single area. On the other hand, since faults are non-dependent, three main hit areas are formed; these also correspond to the neurons illustrated in Figure 9. To exemplify the training progression, Figure 12 illustrates the input planes of each phase for the larger network of 20×20 neurons; in this figure, it is clear how the training and re-training process affects different zones of each input map with each iteration, lighting different zones for each fault and refining the classification for each case. Furthermore, the parameters that describe the structure of each SOM neural network are listed in Table 4.

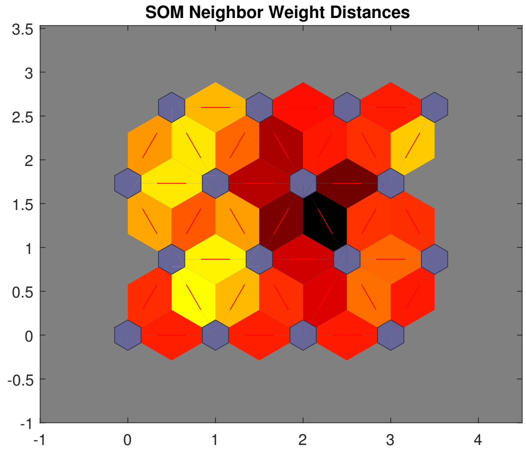

Figure 10. Self Organizing Map (SOM) neighbor distances.

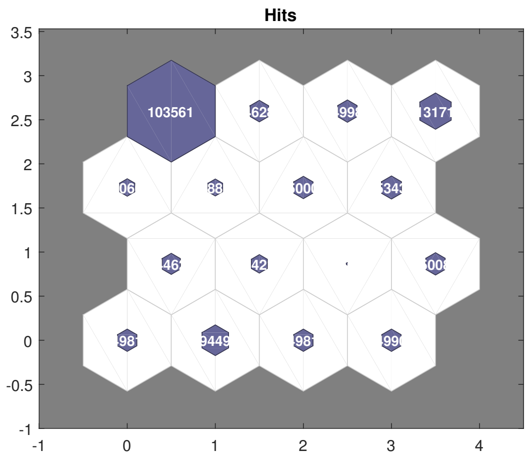

Figure 11. Self Organizing Map (SOM) sample hits.

Table 4. SOM structure.

Parameter	Value
Input size	3
Input variables	Phase A, B, and C
SOM dimension	4×4 & 20×20
Output size	16 & 400
Output variables	Fault Type
Training epochs	1000
Training algorithm	Batch
Performance	Mean square error

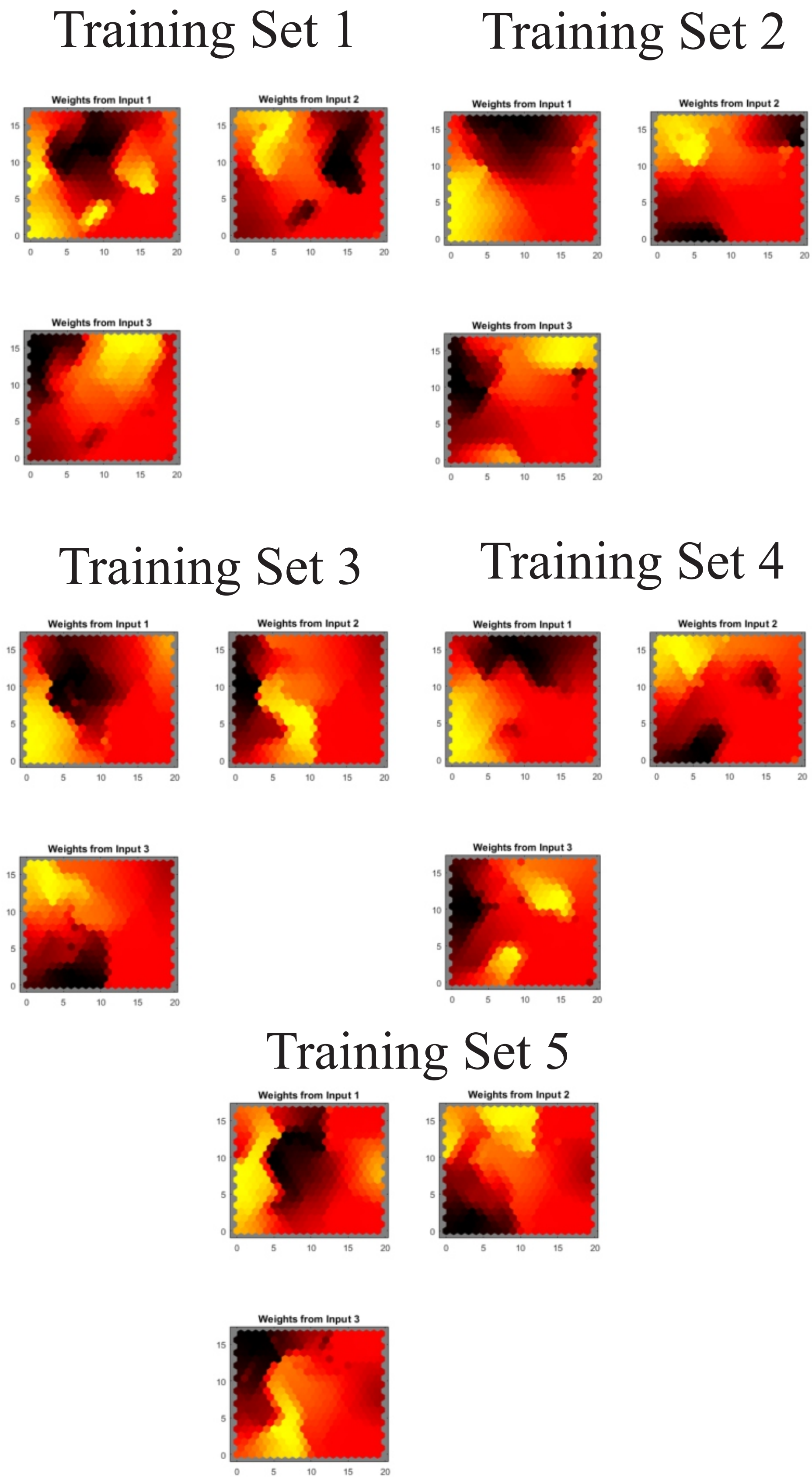

Figure 12. Self Organizing Map (SOM) input planes per phase of the system in a 20 × 20 SOM.

6.2. Hardware and DT Setup

The following step in the offline training process is the assembling of the physical and digital systems. To construct the DT, we used the Simscape Electrical Specialized Power System Library from *MATLAB Simulink*. Furthermore, the parameters of the DT system are chosen according to the hardware available in the laboratory to guarantee digital replicability. Such is the case of the available loads, LC filters, and DC link value.

The equipment used to assemble the power electronic converter consisted of *Imperix* hardware including the rapid prototyping controller BoomBox and two full bridge IGBT-based power modules. In this arrangement, the BoomBox is programmed using *MATLAB Simulink* with a complementary *Imperix* configuration blocks library, such *MATLAB Simulink* model is set to send the output switching signals to the power modules and to the RT simulator through digital output channels connected via optic fiber cables. The set of PWM signals intended for the physical system is directly connected to the IGBT-based power modules, whereas the PWM signals intended for the DT have interfaced with the *OPAL* thanks to a simulation interface developed by *OPAL* and *Imperix* to communicate both platforms.

Moreover, in the hardware configuration, the output legs of the three-phase inverter are connected to an LC filter and then directly to the three-phase restive load. The load and filtering elements are configured using *Festo* equipment included in the *LabVolt Series*. In addition, a *Festo* power supply is used to set the DC link voltage of the inverter system, and to monitor the voltage per phase of the system.

Lastly, the equipment by *OPAL* includes the RT simulator and the HIL data acquisition interface, the latter is used to measure the high voltages per phase and translate them into analog inputs for the RT simulator. The RT simulator is programmed to run the offline SOM neural network and the digital replica of the real system. The SOM outputs are translated to analog outputs of the RT simulator to be interfaced with a 4-channel oscilloscope and obtain the results of the proposed framework. These simulation and hardware parameters are listed in Table 5. In particular, physical parameters are selected according to hardware availability, while the PWM switching frequency is chosen according to conventional values as reported by Smadi et al. [38]. In addition, the complete hardware setup is shown in Figure 13. At the same time, Table 6 complements the proposed framework architecture shown in Figure 4 with the description of hardware and system tasks, pointing out the purpose of each hardware element within the experimental setup.

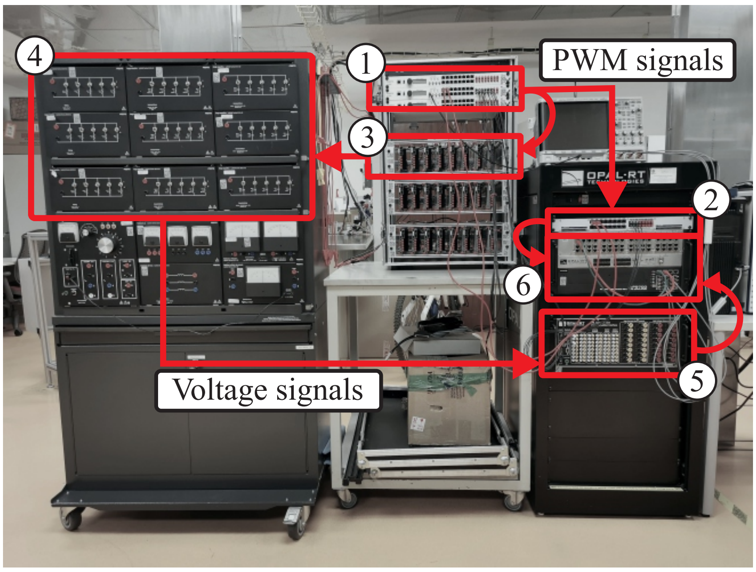

Figure 13. Hardware setup: 1. Imperix Boombox; 2. Imperix/OPAL simulation interface; 3. Imperix IGBT-based power module; 4. LabVolt by Festo; 5. OPAL HIL Controller; 6. OPAL 5700.

Table 5. Case Study system and simulation platforms' parameters.

Parameter	Value
OPAL step time	50 µs
Imperix step time	50 µs
PWM switching frequency	10 kHz
DC link	120 V
R_L	120 Ω
L_f	640 mH
C_f	11 µF

Table 6. Hardware purpose and system tasks.

Hardware	Hardware Purpose	Tasks
1. Imperix Boombox	System PWM inputs	Generate PWM signals for the real and DT system
2. Imperix/OPAL simulation interface		Translate PWM signals to Digital inputs for the OPAL interface
3. Imperix IGBT-based power module	Physical system	Assemble a three-phase two-level inverter
4. LabVolt by Festo		Complement the physical system with the DC link, LC filter, and R loads
5. OPAL HIL Controller	V_{abc} measurements	Translate High-Voltage measurements to analog inputs for the OPAL interface
6. OPAL 5700	Digital Twin	Simulate the DT of the physical system
	Comparison	Compare the voltage readings of the physical system and the DT to generate the error signals.
	SOM	Identify the physical system fault based on the generated error signals

7. Experimental Results and Discussion

The proposed fault identification framework is validated in an experimental setup. The previously described system and hardware setup are tested against four different types of open circuits since these do not cause serious damage to the equipment and comply with the safety laboratory protocols. In addition, each phase is tested individually, in order to guarantee a decoupled identification of the trained clustering neural network as also tested by Poon et al. [11]. The first test consists of the identification of a three-phase open circuit fault; this test is performed by manually disconnecting the three phases from the HIL data acquisition interface. In doing so, a triggering signal is generated in the oscilloscope, where the output of each SOM can also be observed. Each SOM is tested separately; first, Figure 14 shows the result of the small SOM, highlighting the successful detection of a three-phase open circuit fault in a total time of 13 ms of its occurrence. In the same test with the bigger SOM, the detection time demonstrated being faster with a detection time of 200 µs; the results for the bigger SOM are shown in Figure 15.

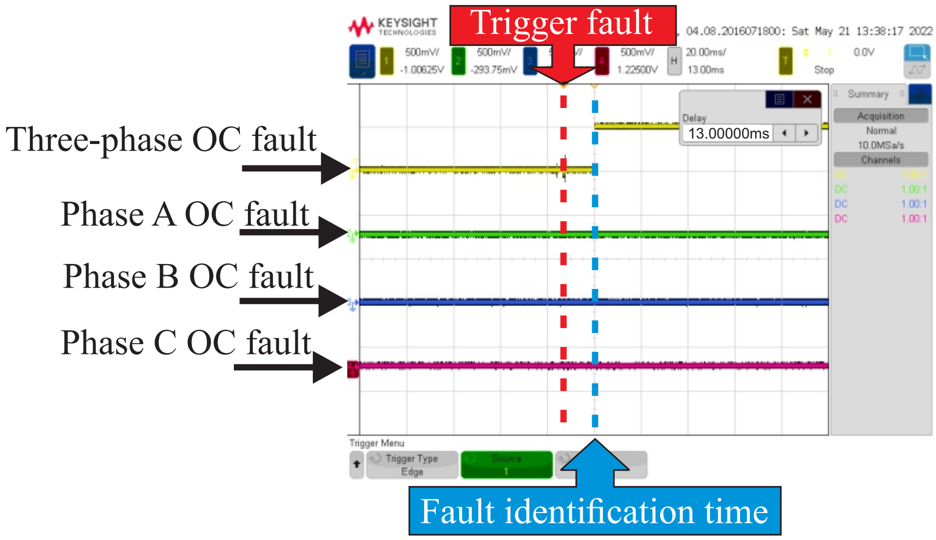

Figure 14. Three-phase open circuit fault identification with 4 × 4 SOM: detection time 13 ms.

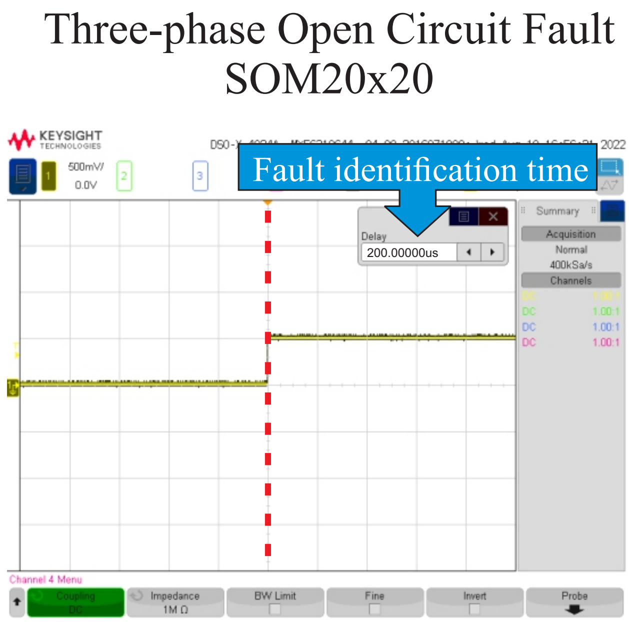

Figure 15. Three-phase open circuit fault identification with 20 × 20 SOM: detection time 200 µs.

The following test consists of identifying single-phase open circuit faults. This is done by manually disconnecting each phase at separate times; each disconnection causes a trigger event that activates the reading of the oscilloscope. Once again, each test is done separately for each SOM; Figure 16 shows the results of successful identification of an open-circuit fault in phase A for the small SOM, while Figure 17 shows the results for the same test scenario for the bigger SOM; the results for each test are 13 ms for the small SOM and 70 µs for the bigger SOM.

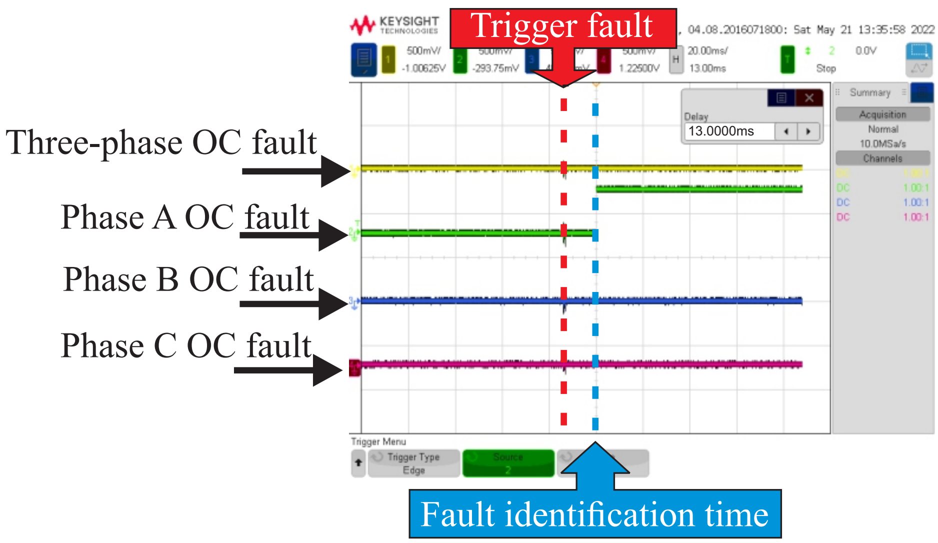

Figure 16. Single-phase open circuit fault identification with 4 × 4 SOM: Phase A; detection time 13 ms.

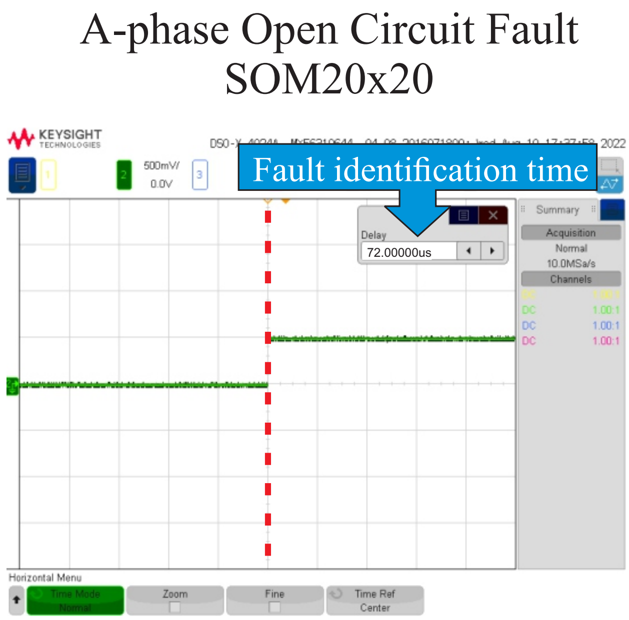

Figure 17. Single-phase open circuit fault identification with 20 × 20 SOM: Phase A; detection time 70 µs.

In the same test condition, phases' open-circuit faults are tested for phase B. In each case, the bigger SOM resulted in a faster detection time for open-circuit faults. The results for phase B are shown in Figures 18 and 19 for the small and bigger SOM, respectively. The detection times are obtained as 1.6 ms for the small SOM and 80 µs for the bigger SOM.

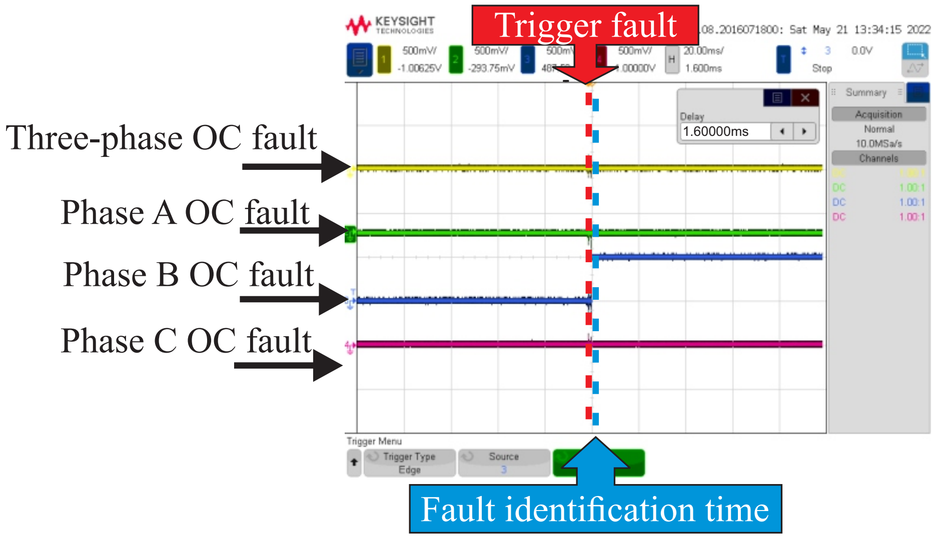

Figure 18. Single-phase open circuit fault identification with 4 × 4 SOM: Phase B; detection time 1.6 ms.

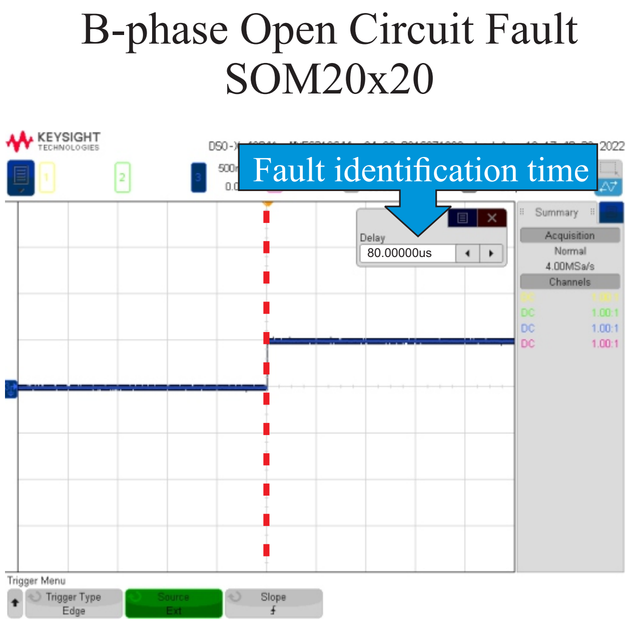

Figure 19. Single-phase open circuit fault identification with 20 × 20 SOM: Phase B; detection time 80 µs.

Lastly, open-circuit faults' tests for phase C is carried out by following the same disconnection procedure for each SOM. Coincident with the above results, the smaller SOM had a higher fault identification time, with obtained detection times of 1.4 ms and 100 µs for the small and bigger SOM, respectively. The experimental results for phase C are shown in Figures 20 and 21 for the small and bigger SOM, respectively.

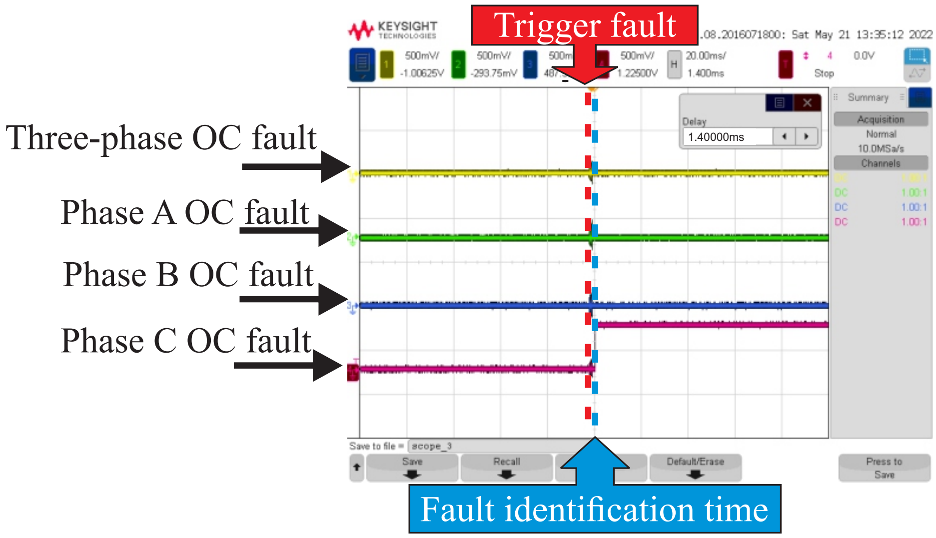

Figure 20. Single-phase open circuit fault identification with 4 × 4 SOM: Phase C; detection time 1.4 ms.

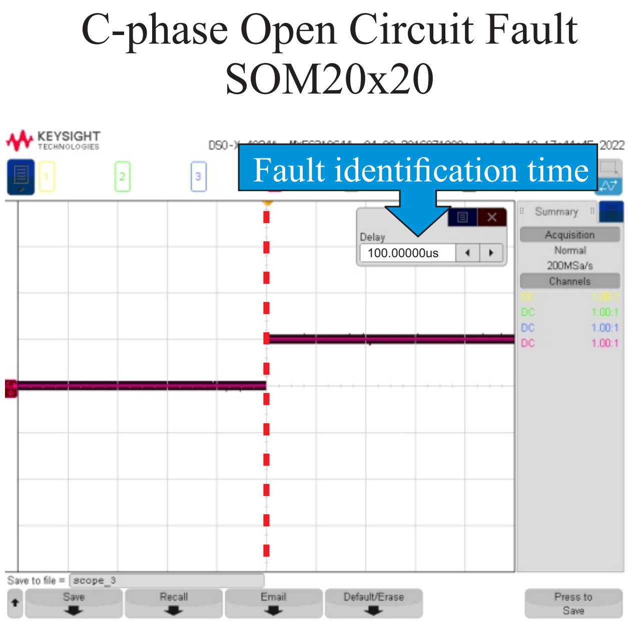

Figure 21. Single-phase open circuit fault identification with 20 × 20 SOM: Phase C; detection time 100 µs.

Compared with the different fault identification methods found during the literature review, the proposed fault identification framework offers a simple, noninvasive, and fast identification approach for single and three-phase open circuit faults. When compared to the work proposed by Poon et al. [11], the proposed RT fault identification framework does not separate the detection and the identification stages; on the contrary, these stages happen at the same time, resulting in a faster detection time despite the size of the SOM, with identification scores two times faster for single-phase open circuit faults. On the other hand, when the proposed method is compared against the system-based method proposed by Estima and Marques Cardoso [12], only the bigger SOM results in an identification time score from 11 to 16 times faster for single-phase open circuit faults. In addition, the proposed method in this work also has the advantage of identifying three-phase open circuit faults since this fault scenario is not tested by the authors in [11] and Estima and Marques Cardoso [12]. The fault identification time scores is shown in Table 7 for better comparison.

As an advantage of our proposal, the fault identification time does not depend on how fast the fault signature evolves as in model-based or system-based methods; it depends on the SOM training efficiency to distinguish the different fault signatures and size of the SOM as showed by the experimental results. In addition, the proposal of this work does not require complex programming or an extensive mathematical analysis as required in other model-based techniques. It also has no requirement for external hardware, making this a noninvasive approach. Furthermore, the application of RT simulation as the environment for model validation and fault identification increases the reliability of the system and did not require additional processing time for the bigger SOM. Essentially, RT becomes a part of the operation of the fault identification process and it is not just used as a validation tool. A summary of the experimental results is shown in Table 7.

Table 7. Experimental results.

Experimental Summary and Comparison for Open Circuit Fault Identification			
Fault Type	**Identification Time SOM 4 × 4**	**Identification Time SOM 20 × 20**	**Identification Time [11,12]**
Three-phase open circuit	13 ms	200 µs	NA, NA
Single-phase open circuit (Phase A)	13 ms	70 µs	3.45 ms, 1.13 ms
Single-phase open circuit (Phase B)	1.6 ms	80 µs	3.45 ms, 1.13 ms
Single-phase open circuit (Phase C)	1.4 ms	100 µs	3.45 ms, 1.13 ms
Summary of Training Parameters			
Total number of neurons	16	400	
SOM training iterations	5	5	
time per iteration	≈4 min	≈20 min	
SOM training process	≈20 min	≈120 min	

8. Trends of Digital Twin in the Microgrid Landscape

As the application of DT, technologies have gained popularity among a wide variety of sectors within different industries such as manufacturing processes and energy management systems, its application in the operation sector of energy systems still remains an open research field with open challenges in the modeling, execution, and validation areas. For instance, the challenge of having a holistic model able to contemplate the total life-cycle of a given component of the microgrid by considering operation scenarios and operating conditions that help in the planning of production, operation, maintenance, and end-of-life stages of power system components. The resolution of such a challenge could conduct more sustainable and cost-effective systems. Certainly, one of the future research trends would be the connection of all three stages in the life-cycle of a given microgrid asset, diving more into the operation technologies realm and not just sticking to the management and high-level control strategies as reviewed in the literature.

With the incremental complexity of microgrids and power systems, the application of ML as an auxiliary engine has been on the rise in recent years. ML is commonly paired with DTs with the objective of having a smarter system that increases the automation level of certain tasks and even provides solutions to immediate problems. In addition, ML is also used to learn more about how this system performs during specific conditions and contributes to the prediction capabilities of DT technologies. Nonetheless, the handling of data are still a major topic within the DT technology scheme. This is being solved by the many data-analytic methods behind the Data Science area, an area which has been described as crucial in the further development of DTs [22]. Further research regarding the analytics of the obtained data is required to allow a seamless flow of information, especially since the integration of intelligent measurement devices has been increasing over the last few years. Assuredly, this has an impact on the amount of data that needs to be processed;

therefore, it is imperative that further research includes the integration of DT technologies and the Data Science area.

On another topic, the validation tools used for DTs have not been excessively studied in the literature, since DTs are mostly used in the manufacturing area, conventional validation tools have been shown to be an effective tool for this task; therefore, the exploration of new solution has not been entirely addressed. However, as described in this work, these validation tools may not be accurate enough in the power system area due to dynamic complexity. Therefore, there is a present need to improve the validation process when applying the DT technologies to the area of power systems.

By improving the validation stage, the planning and deployment of microgrids can be achieved in a more seamless manner, by including a specific design characteristic in the obtained model, one could simulate how the microgrid dynamic would respond to the new design characteristic and find early fault conditions. For example, the microgrid operator would be able to verify the response of the microgrid if a new cluster of AGUs is integrated within its generation network, corroborating whether this action would compromise or benefit the system. In another example, due to recent increments in the Electric Vehicles market, electric networks need to be modified to cope with the nonlinear dynamic of installing charging stations across the network, causing extra strain on the power grid. This paradigm can be dealt with at a competent validation stage in the DT of the electrical network, by knowing how the grid would respond to the nonlinearities before these are installed. Essentially, the provided singularities of combining DTs with a reliable validation tool such as RT simulation can lead to improved planning, prevention, and fast fault identification of microgrid systems; it would be the responsibility of future researchers to find better solutions and applications where these tools can be used as a complement of each other to improve on how microgrid technologies are studied and developed.

9. Conclusions

This work introduces a generic fault identification framework based on the application of DT technologies, RT simulation, and ML clustering techniques. The presented work evaluates different sizes of a clustering ML strategy within a high dynamic fidelity DT environment. Due to the complexity of power systems, the application of RT creates an adequate simulation environment thanks to the dynamic consistency advantages over traditional simulation systems, giving consistent results between the digital model and the experimental setup. Overall, this environment also contributes to a consistent performance of the trained classification method. In addition, this work validates that the application of a SOM for power systems fault identification is an adequate alternative to traditional fault identification methods with fast and consistent results. As presented in this work, one of the main advantages of having open circuit fault identification with an ML classification method is that it does not depend on how fast the fault signature evolves to be identified correctly. On the contrary, the fault identification method proposed in this work demonstrates that the ML clustering technique depends on the SOM training process and its dimensions rather than the model or other metering strategies found during the literature review. This work demonstrates that high dimension SOM does improve the fault identification time when compared to a smaller SOM and other model-based and system-based methods with up to 2 and 16 times faster detection time, respectively. A higher dimension SOM produces faster fault identification times without requiring additional processing or computing during the online simulation. Principally, the proposed framework identifies the fault type induced in the experimental system in a successful and rapid manner. By having the DT within the RT environment, the successful implementation of this proposal is achieved, since the fast dynamics of the power converts can be guaranteed, leading to a more reliable fault identification process. Furthermore, this framework involves RT technologies as a part of the end-product, meaning that RT is not only used for validation and proof of concept but is also involved in the final operating stage where the correct operation of the proposed framework has been validated through an HIL experimental approach.

Author Contributions: Conceptualization, J.R.L., P.P., and B.M.; methodology, J.R.L., P.P., and B.M.; validation, J.R.L. and P.P.; investigation, J.R.L. and J.d.J.C.; writing—original draft preparation, J.R.L. and J.d.J.C.; writing—review and editing, J.R.L. and P.P.; visualization, J.R.L.; supervision, J.R.L., P.P., B.M. and A.M.; project administration, P.P. and A.M.; funding acquisition, A.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Not applicable, the study does not report any data.

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

References

1. Olivares, D.E.; Mehrizi-Sani, A.; Etemadi, A.H.; Canizares, C.A.; Iravani, R.; Kazerani, M.; Hajimiragha, A.H.; Gomis-Bellmunt, O.; Saeedifard, M.; Palma-Behnke, R.; et al. Trends in Microgrid Control. *IEEE Trans. Smart Grid* **2014**, *5*, 1905–1919. [CrossRef]
2. Kroposki, B.; Lasseter, R.; Ise, T.; Morozumi, S.; Papathanassiou, S.; Hatziargyriou, N. Making microgrids work. *IEEE Power Energy Mag.* **2008**, *6*, 40–53. [CrossRef]
3. Guerrero, J.M.; Vasquez, J.C.; Matas, J.; de Vicuna, L.G.; Castilla, M. Hierarchical Control of Droop-Controlled AC and DC Microgrids A General Approach Toward Standardization. *IEEE Trans. Ind. Electron.* **2011**, *58*, 158–172. [CrossRef]
4. Luo, F.; Ranzi, G.; Wang, S.; Dong, Z.Y. Hierarchical Energy Management System for Home Microgrids. *IEEE Trans. Smart Grid* **2019**, *10*, 5536–5546. [CrossRef]
5. Han, Y.; Li, H.; Shen, P.; Coelho, E.A.A.; Guerrero, J.M. Review of Active and Reactive Power Sharing Strategies in Hierarchical Controlled Microgrids. *IEEE Trans. Power Electron.* **2017**, *32*, 2427–2451. [CrossRef]
6. Mirafzal, B. Survey of Fault-Tolerance Techniques for Three-Phase Voltage Source Inverters. *IEEE Trans. Ind. Electron.* **2014**, *61*, 5192–5202. [CrossRef]
7. Shao, S.; Wheeler, P.W.; Clare, J.C.; Watson, A.J. Fault Detection for Modular Multilevel Converters Based on Sliding Mode Observer. *IEEE Trans. Power Electron.* **2013**, *28*, 4867–4872. [CrossRef]
8. Ibarra, L.; Rosales, A.; Ponce, P.; Molina, A.; Ayyanar, R. Overview of Real-Time Simulation as a Supporting Effort to Smart-Grid Attainment. *Energies* **2017**, *10*, 817. [CrossRef]
9. Lu, B.; Wu, X.; Figueroa, H.; Monti, A. A Low-Cost Real-Time Hardware-in-the-Loop Testing Approach of Power Electronics Controls. *IEEE Trans. Ind. Electron.* **2007**, *54*, 919–931. [CrossRef]
10. Lopez, J.R.; Ibarra, L.; Ponce, P.; Molina, A. A Decentralized Passive Islanding Detection Method Based on the Variations of Estimated Droop Characteristics. *Energies* **2021**, *14*, 7759. [CrossRef]
11. Poon, J.; Jain, P.; Konstantakopoulos, I.C.; Spanos, C.; Panda, S.K.; Sanders, S.R. Model-Based Fault Detection and Identification for Switching Power Converters. *IEEE Trans. Power Electron.* **2017**, *32*, 1419–1430. [CrossRef]
12. Estima, J.O.; Marques Cardoso, A.J. A New Approach for Real-Time Multiple Open-Circuit Fault Diagnosis in Voltage-Source Inverters. *IEEE Trans. Ind. Appl.* **2011**, *47*, 2487–2494. [CrossRef]
13. Saad, A.; Faddel, S.; Youssef, T.; Mohammed, O.A. On the Implementation of IoT-Based Digital Twin for Networked Microgrids Resiliency Against Cyber Attacks. *IEEE Trans. Smart Grid* **2020**, *11*, 5138–5150. [CrossRef]
14. Botín-Sanabria, D.M.; Mihaita, A.S.; Peimbert-García, R.E.; Ramírez-Moreno, M.A.; Ramírez-Mendoza, R.A.; Lozoya-Santos, J.d.J. Digital Twin Technology Challenges and Applications: A Comprehensive Review. *Remote Sens.* **2022**, *14*, 1335. [CrossRef]
15. Solman, H.; Kirkegaard, J.K.; Smits, M.; Van Vliet, B.; Bush, S. Digital twinning as an act of governance in the wind energy sector. *Environ. Sci. Policy* **2022**, *127*, 272–279. [CrossRef]
16. Jafari, M.A.; Zaidan, E.; Ghofrani, A.; Mahani, K.; Farzan, F. Improving Building Energy Footprint and Asset Performance Using Digital Twin Technology. *IFAC-PapersOnLine* **2020**, *53*, 386–391. [CrossRef]
17. Onile, A.E.; Machlev, R.; Petlenkov, E.; Levron, Y.; Belikov, J. Uses of the digital twins concept for energy services, intelligent recommendation systems, and demand side management: A review. *Energy Rep.* **2021**, *7*, 997–1015. [CrossRef]
18. Glaessgen, E.; Stargel, D. The Digital Twin Paradigm for Future NASA and U.S. Air Force Vehicles. In Proceedings of the 53rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference 20th AIAA/ASME/AHS Adaptive Structures Conference 14th AIAA, Honolulu, HI, USA, 23–26 April 2012; American Institute of Aeronautics and Astronautics: Honolulu, HI, USA, 2012; p. 14. [CrossRef]
19. Tao, F.; Zhang, H.; Liu, A.; Nee, A.Y.C. Digital Twin in Industry: State- and f-the-Art. *IEEE Trans. Ind. Inform.* **2019**, *15*, 2405–2415. [CrossRef]
20. Wang, X.; Wang, Y.; Tao, F.; Liu, A. New Paradigm of Data-Driven Smart Customisation through Digital Twin. *J. Manuf. Syst.* **2021**, *58*, 270–280. [CrossRef]
21. Park, H.A.; Byeon, G.; Son, W.; Jo, H.C.; Kim, J.; Kim, S. Digital Twin for Operation of Microgrid: Optimal Scheduling in Virtual Space of Digital Twin. *Energies* **2020**, *13*, 5504. [CrossRef]

22. Kaur, M.J.; Mishra, V.P.; Maheshwari, P. The Convergence of Digital Twin, IoT, and Machine Learning: Transforming Data into Action. In *Digital Twin Technologies and Smart Cities*; Farsi, M., Daneshkhah, A., Hosseinian-Far, A., Jahankhani, H., Eds.; Internet of Things; Springer International Publishing: Cham, Switzerland, 2020; pp. 3–17. [CrossRef]
23. Sekine, Y.; Takahashi, K.; Sakaguchi, T. Real-time simulation of power system dynamics. *Int. J. Electr. Power Energy Syst.* **1994**, *16*, 145–156. [CrossRef]
24. Bélanger, J.; Venne, P. The What, Where and Why of Real-Time Simulation. *Planet RT* **2010**, *1*, 25–29.
25. Zhang, B.; Deng, W.; Ruan, L.; Wang, T.; Quan, J.; Cao, Q.; Teng, Y.; Wang, W.; Yuan, Y.; Li, L. Field–Circuit Cosimulation of 500-kV Transformers in AC/DC Hybrid Power Grid. *IEEE Trans. Appl. Supercond.* **2016**, *26*, 1–5. [CrossRef]
26. Omar Faruque, M.D.; Strasser, T.; Lauss, G.; Jalili-Marandi, V.; Forsyth, P.; Dufour, C.; Dinavahi, V.; Monti, A.; Kotsampopoulos, P.; Martinez, J.A.; et al. Real-Time Simulation Technologies for Power Systems Design, Testing, and Analysis. *IEEE Power Energy Technol. Syst. J.* **2015**, *2*, 63–73. [CrossRef]
27. Dangeti, P. *Statistics for Machine Learning*; Packt Publishing Ltd.: Birmingham, UK, 2017.
28. Almutairy, I.; Alluhaidan, M. Fault diagnosis based approach to protecting DC microgrid using machine learning technique. *Procedia Comput. Sci.* **2017**, *114*, 449–456. [CrossRef]
29. Ali, W.; Ulasyar, A.; Mehmood, M.U.; Khattak, A.; Imran, K.; Zad, H.S.; Nisar, S. Hierarchical Control of Microgrid Using IoT and Machine Learning Based Islanding Detection. *IEEE Access* **2021**, *9*, 103019–103031. [CrossRef]
30. Mele, E.; Elias, C.; Ktena, A. Electricity use profiling and forecasting at microgrid level. In Proceedings of the 2018 IEEE 59th International Scientific Conference on Power and Electrical Engineering of Riga Technical University (RTUCON), Riga, Latvia, 12–13 November 2018; pp. 1–6. [CrossRef]
31. Llanos, J.; Morales, R.; Núñez, A.; Sáez, D.; Lacalle, M.; Marín, L.G.; Hernández, R.; Lanas, F. Load estimation for microgrid planning based on a self-organizing map methodology. *Appl. Soft Comput.* **2017**, *53*, 323–335. [CrossRef]
32. Kohonen, T. Self-organized formation of topologically correct feature maps. *Biol. Cybern.* **1982**, *43*, 59–69. [CrossRef]
33. Isa, D.; Kallimani, V.; Lee, L.H. Using the self organizing map for clustering of text documents. *Expert Syst. Appl.* **2009**, *36*, 9584–9591. [CrossRef]
34. Bhavsar, H.; Ganatra, A. A comparative study of training algorithms for supervised machine learning. *Int. J. Soft Comput. Eng. (IJSCE)* **2012**, *2*, 2231–2307.
35. Alashwal, H.; El Halaby, M.; Crouse, J.J.; Abdalla, A.; Moustafa, A.A. The application of unsupervised clustering methods to Alzheimer's disease. *Front. Comput. Neurosci.* **2019**, *13*, 31. [CrossRef] [PubMed]
36. Shahin, I.; Nassif, A.B.; Hamsa, S. Emotion recognition using hybrid Gaussian mixture model and deep neural network. *IEEE Access* **2019**, *7*, 26777–26787. [CrossRef]
37. Rodrigues Ferreira, F.; Silveira, I.; Notargiacomo, P. A Konet-based Tool for Adaptive Learning: An Application for Ethnic Learning of Music. In Proceedings of the XVIII Simposio Brasileiro de Informática na Educacao 2007, São Paulo, Brazil, 28–30 November 2007. [CrossRef]
38. Smadi, I.A.; Albatran, S.; Ahmad, H.J. On the Performance Optimization of Two-Level Three-Phase Grid-Feeding Voltage-Source Inverters. *Energies* **2018**, *11*, 400. [CrossRef]