energies

MDPI

Article

Event-Based Simulation of a Decentralized Protection System Based on Secured GOOSE Messages

Pablo Ledesma [1,*], Peyman Jafary [2], Sami Repo [2], Amelia Álvarez [3], Francisco Ramos [3], Davide Della Giustina [4] and Alessio Dedè [4]

[1] Universidad Carlos III de Madrid, 28911 Madrid, Spain
[2] Tampere University of Technology, 33720 Tampere, Finland; peyman.jafary@tut.fi (P.J.); sami.repo@tut.fi (S.R.)
[3] Schneider Electric, 41092 Sevilla, Spain; amelia.alvarez@schneider-electric.com (A.Á.); francisco.ramos@schneider-electric.com (F.R.)
[4] Unareti, 25124 Brescia, Italy; davide.dellagiustina@unareti.it (D.D.G.); alessio.dede@unareti.it (A.D.)
* Correspondence: pablole@ing.uc3m.es

Received: 17 May 2020; Accepted: 16 June 2020; Published: 23 June 2020

Abstract: A new simulation library is developed on OMNeT++ to model faults in distribution systems. The proposed library makes it possible to calculate the status of lines and busbars from the point of view of a protection system, enabling the modeling of overcurrents, power outages and fault passage indicators. The library is applied to model a decentralized protection system based on the exchange of IEC 61850 Generic Object Oriented Substation Events (GOOSE) messages between intelligent electronic devices responsible for the operation of circuit breakers and disconnectors. The time needed to secure and transmit GOOSE messages over the Internet is characterized and included in the model. Several studies are carried out to analyze the effect of different parameters, such as GOOSE retransmission times and failure rates of switching devices and communication channels, on the performance of the protection system.

Keywords: modelling of power systems; fault location isolation and system restoration; digital protections; automation of distribution systems; IEC 61850

1. Introduction

The integration of information and communications technology (ICT) is one of the more relevant innovations in power systems and has a wide range of implications in many aspects of system operations. One of the fields in which ICT has a major impact is the protection of distribution systems. A number of decentralized and semicentralized protection systems have been developed in the last decade [1–3]; these systems have taken advantage of the functionality provided by ICT and the flexibility provided by communication protocols defined by the IEC 61850 standard [4]. Advantages of decentralized protection systems include a reduction in costs, in communications with the control center and in reconfiguration speed [5,6].

Decentralized protection systems rely on intelligent electronic devices (IEDs) communicating with each other, receiving information from metering devices and sending orders to switching devices. Some decentralized IEDs exchange Generic Object Oriented Substation Events (GOOSE) messages over an IP-based network. Therefore, cybersecurity becomes important, and GOOSE messages must be secured in order to reduce security risks and minimize the probability of cyberattacks [7,8].

Decentralized IEDs responsible for performing fault location and isolation functions are routinely tested in laboratories under working conditions [9], sometimes including software-in-the-loop simulations. In addition to laboratory tests, and before field deployment, extensive software simulations of a protection system are needed to assess the functional and economic performance of the solution [10]. These simulations make it possible to take into account the statistical nature of electrical faults and failures in switching devices and communication channels.

Increasingly, due to the reduction in the difference in price between circuit breakers and switch disconnectors, system operators are deploying circuit breakers not only in the primary substation but also along distribution lines [5]. Software simulations may be used as well for the design of the protection architecture. This design entails selecting which secondary substations should be equipped with circuit breakers and which should be equipped with disconnectors, depending on fault recovery times, customer loads and economic cost of the deployment.

Simulating power and communication systems together is difficult because the approaches used to represent both systems are different in nature. On the one hand, power system simulators apply numerical methods and matrix operations to solve problems such as load flows, short-circuit analyses and dynamic studies to assess the performance of protection algorithms. On the other hand, simulation of a communication network is typically an event-based process that involves several agents operating independently and sharing communication channels.

OMNeT++ is a discrete event simulator that provides libraries for a variety of network protocols and technologies, together with facilities for statistical analysis [11], and thus is a natural choice for the simulation of communication systems. However, this simulator does not provide any functionality to represent electrical networks. One alternative is to merge it with other software tools by applying cosimulation [12,13]. Cosimulation has the advantage of making it possible to use a simulator specialized for power systems; however, this often requires the development of a separate program to coordinate both simulators, which can complicate the development of new cases and the modification of old ones. For example, a component object model (COM) server is needed to combine OpenDSS and OMNeT++ [14–17].

A different approach is to model the communication and power systems with the same tool. While the modular, event-based approach of OMNeT++ makes it unsuitable to integrate a model of a power system based on its admittance or impedance matrix, it is possible to develop a model that identifies the nodes affected by a fault in a distribution network. This work proposes a procedure that uses OMNeT++ as a single simulation tool for both communication and power networks, thus facilitating the modeling and testing of fault location and isolation systems, on a single graphical user interface, as a logical entity. This paper is focused on the simulation of the electrical network; by using OMNeT++ and its INET framework, the user can model the communication network with the level of detail required for each specific study.

The main contributions of this work are:

- The description of an OMNeT++ simulation library that represents distribution power networks. The aim of the library is to model the status of the grid from the point of view of the logic of protection systems. It therefore does not simulate electrical variables such as voltage of short-circuit current, variables that are best calculated following a co-simulation approach. Instead, the emphasis of the proposed library is set on making a tool transparent to the user, making it easy to model new cases, and facilitating the collection of data.
- The characterization in a laboratory of the processing times needed to secure and transmit GOOSE messages between IEDs, and the application of the proposed simulation library to model a decentralized protection system that uses secure GOOSE messages. The resulting study case serves both as a demonstration of the library and as a validation of the specific protection system.

The simulator has been developed as part of the IDE4L demonstration project [18] and has been applied to a model of one of its demonstration sites.

2. Simulation of a Distribution System in OMNeT++

OMNeT++ offers a simulation framework designed primarily for communication networks. This work extends the OMNeT++ simulation library by developing a series of modules that represent electrical network components and the IEDs associated with some of them. Instead of calculating the actual values of voltage and current along the feeder, these modules calculate the status of the electrical network from the point of view of the logic of the protection system.

The new OMNeT++ modules are listed in Table 1. Busbar is the most basic component and can be used to represent any electrical node in the system. A Busbar module can have a load and a number of customers associated with it, and it can be electrically connected to any number of other modules. Busbar modules can represent actual busbars in substations, loads, or sections of the distribution line.

The status of a Busbar module in the simulator can be the following:

- *Normal*, when the busbar is connected to a stable power supply and not to a fault.
- *Disconnected*, when the busbar is not connected to a stable power supply.
- *Fault*, when a fault is applied at the busbar. To model a fault in a line, the corresponding section of the line must be represented by a Busbar module.
- *Fault current*, when the busbar is connected to a fault and is in the path of the fault current. Fault currents normally pass through at least one circuit breaker responsible for clearing the fault.
- *Abnormal voltage*, when the busbar is connected to a fault and is not in the path of the fault current.

The appearance of the corresponding module on the simulator varies according to the icons listed in Table 1.

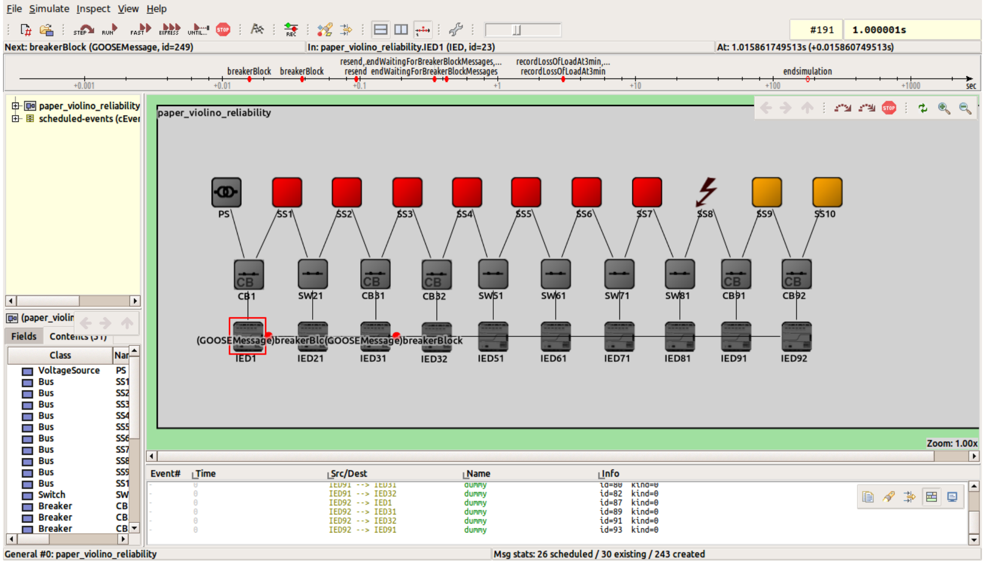
Table 1. Developed OMNeT++ modules.

Module	Icon/Status
Busbar	🟩 Normal
	⬜ Disconnected
	⚡ Fault
	🟥 Fault current
	🟨 Abnormal voltage
Power Source	⊚
Circuit Breaker	CB
Disconnector	⟋
IED	▤

Power Source modules represent the connection point of the distribution system to a higher voltage network. There must be at least one Power Source module in a simulation case. There must also be an electrical path from any Busbar module to a Power Source module; otherwise, the Busbar module is marked as *Disconnected*. A Power Source module acts as a source of fault current when a short-circuit is applied in a Busbar module.

Circuit Breaker and Disconnector modules represent switching devices with two electrical connection points, the main difference between them being that circuit breakers can interrupt short-circuit currents, while disconnectors can open only in the absence of current. IED modules are associated with a Circuit Breaker or a Disconnector module, can communicate with each other and contain the logic that performs the fault location and isolation procedures. The IED modules must be programmed to represent the logic of the protection system and the characteristics of the communication network. The extensive libraries provided with OMNeT++ can be used, if necessary, to model the corresponding wired or wireless link layer protocols.

OMNeT++ includes a simulation integrated development environment (IDE) based on the software development platform, Eclipse. The electrical modules can be easily combined to model a distribution network by using drag-and-drop functionalities. Figure 1 shows an example of the appearance of a simulated case in the IED. The complete, developed library can be downloaded from Reference [19].


File Simulate Inspect View Help
STEP RUN FAST EXPRESS UNTIL... STOP REC
#191
1.000001s
breakerBlock breakerBlock
resend,.endWaitingForBreakerBlockMessages,...
resend endWaitingForBreakerBlockMessages
recordLossOfLoadAt3min,...
recordLossOfLoadAt3min
endsimulation
+0.001 +0.01 +0.1 +1 +10 +100 +1000 sec
paper_violino_reliability
scheduled-events (cEver
(paper_violir
Fields Contents (31)
PS SS1 SS2 SS3 SS4 SS5 SS6 SS7 SS8 SS9 SS10
CB1 SW21 CB31 CB32 SW51 SW61 SW71 SW81 CB91 CB92
(GOOSEMessage)breakerBlc(GOOSEMessage)breakerBlock
IED1 IED21 IED31 IED32 IED51 IED61 IED71 IED81 IED91 IED92
Zoom: 1.00x

Next: breakerBlock (GOOSEMessage, id=249)
In: paper_violino_reliability.IED1 (IED, id=23)
At: 1.015861749513s (+0.015860749513s)

Class	Nar
VoltageSource	PS
Bus	SS1
Bus	SS2
Bus	SS3
Bus	SS4
Bus	SS5
Bus	SS6
Bus	SS7
Bus	SS8
Bus	SS9
Bus	SS1
Switch	SW
Breaker	CB
Breaker	CB
Breaker	CB

```
Event#  Time   Src/Dest          Name    Info
-       0      IED91 --> IED31   dummy   id=80  kind=0
-       0      IED91 --> IED32   dummy   id=82  kind=0
-       0      IED92 --> IED1    dummy   id=87  kind=0
-       0      IED92 --> IED31   dummy   id=89  kind=0
-       0      IED92 --> IED32   dummy   id=91  kind=0
-       0      IED92 --> IED91   dummy   id=93  kind=0
```

General #0: paper_violino_reliability
Msg stats: 26 scheduled / 30 existing / 243 created

Figure 1. OMNeT++ visual interface containing one of the simulated cases.

2.1. *Principle of Simulation of the Electrical Network*

The simulation of the status of the electrical network in the proposed library relies on the exchange of messages between electrical modules to actualize the status of the distribution network. OMNeT++ is an event-based simulation program. The approach adopted to simulate the electric network is based on the principle that any event in the electrical network, such as a short circuit or the opening or closing of a switching device, triggers the sending of an OMNeT++ message from the affected module to all its connected electrical modules. When a component of the circuit receives the message, it actualizes its own status and resends the message until the message reaches either the end of the line or an open switching device. During this process, the simulation time does not advance so that changes in the status of the electrical components are perceived as instantaneous in the simulation. It must be noted that the OMNeT++ messages used to simulate the electrical network are completely unrelated to the GOOSE messages used by the decentralized protection system.

Table 2 shows the messages used to actualize the status of the network. The way that these messages are used to simulate the electrical network can be described by differentiating between three different events: Initialization, application of a fault, and operation of a switching device.

2.1.1. Initialization

Figure 2 shows, as an example, the sequence of messages sent during the initialization of a small network consisting of four buses and a circuit breaker. At the beginning of the simulation, the Power Source module sends an *electrSrc* message. Each electrical module that receives this message actualizes its status to *Normal*, records the gate through which the message arrived, and resends the message to the rest of the electrical modules connected to itself. In this way, the *electrSrc* message spreads through the radial network, allowing each component to know the direction of the power source and defining the upstream and downstream directions along the grid. Any Power Source module that does not receive an *electrSrc* message is set to *Disconnected*.

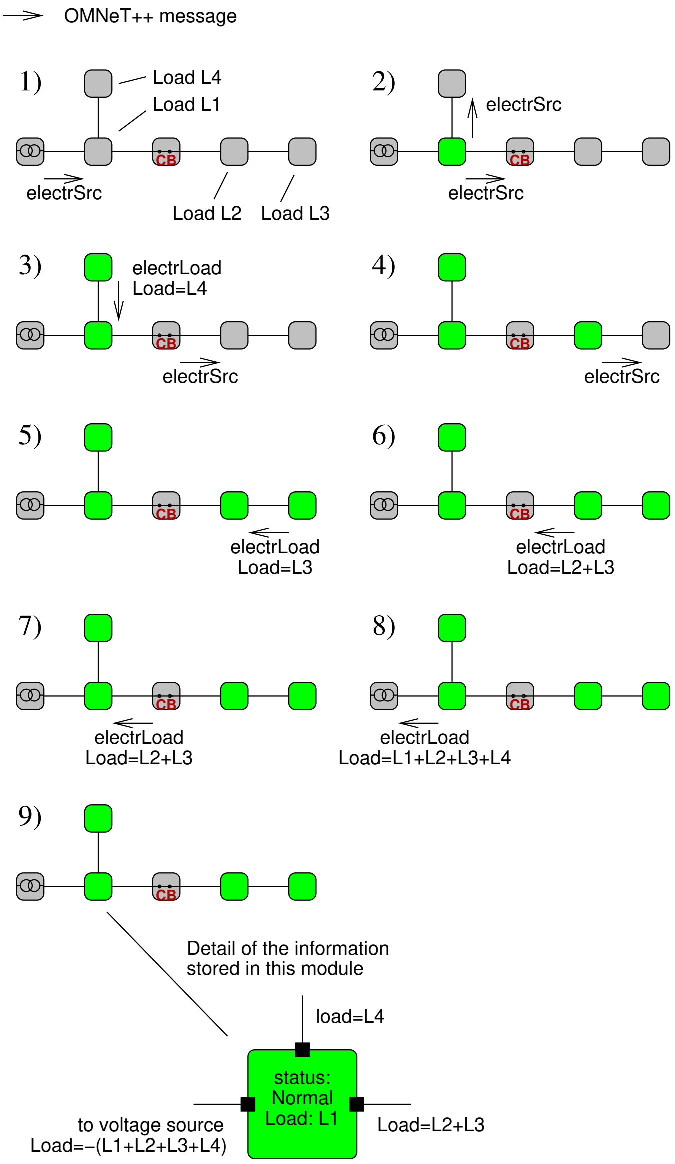


Figure 2. Steps (1)–(9) during the initialization of the electric network.

When an *electrSrc* message arrives at a Busbar module that is a dead end (i.e., the last busbar in a feeder), this Busbar module sends back an *electrLoad* message that contains its load and number of customers. Each electrical module that receives an *electrLoad* message waits for the rest of the downstream connected components to send their corresponding *electrLoad* messages. Once the module has received all of them, the electrical module sends its own *electrLoad* through the upstream gate containing the sum of its own load and customers as well as all the downstream loads and customers. Accordingly, the active power, excluding losses, and the number of customers is calculated at every section of the line. During the initialization, closed Circuit Breaker and Disconnector modules behave in a similar way as Busbar modules, the only difference being that they have zero load and only two connection points. Similarly, open Circuit Breaker and Disconnector modules act as end-of-line Busbar modules with zero load.

For the sake of simplicity, the results in this paper are limited to a radial network with no distributed generation. However, the proposed technique can be easily adapted to model networks with several energy sources by assigning a different message *electrSrc* to each source. Information about whether each energy source is able to provide short-circuit current can be stored in the elements of the circuit, and later be used to calculate the path of short-circuit current from the sources to the fault. As for non-radial topologies, they can be detected because at least one bus would receive the same *electrSrc* message from two different directions. This make it possible to raise a warning message if the distribution network is supposed to be always radially operated.

2.1.2. Application of a Fault

Faults are applied in a Busbar module by the OMNeT++ self-message *electrApplyFault* programmed to be sent at the desired simulation time.

When a fault is applied, the corresponding Busbar module sends an *electrFault* message to all its connected electrical modules. Each electrical module resends this message until it reaches either an open switching device, the Power Source module or the end of the line.

During this process, a Busbar module that receives the *electrFault* message from a downstream component (a component not situated in the path to the voltage source) actualizes its status to *Fault current*, and a Busbar module that receives the *electrFault* message from an upstream component (a component situated in the path to the voltage source) actualizes its status to *Abnormal voltage*. This message travels along the circuit, making it possible to determine the busbars affected by the fault and the path of the fault current from the stable power supply to the faulted busbar.

Figure 3 shows the sequence of messages sent to apply a fault. As in the initialization, closed Circuit Breaker and Disconnector modules behave in a similar way to Busbar modules, while open Circuit Breaker and Disconnector modules act as end-of-line Busbar modules.

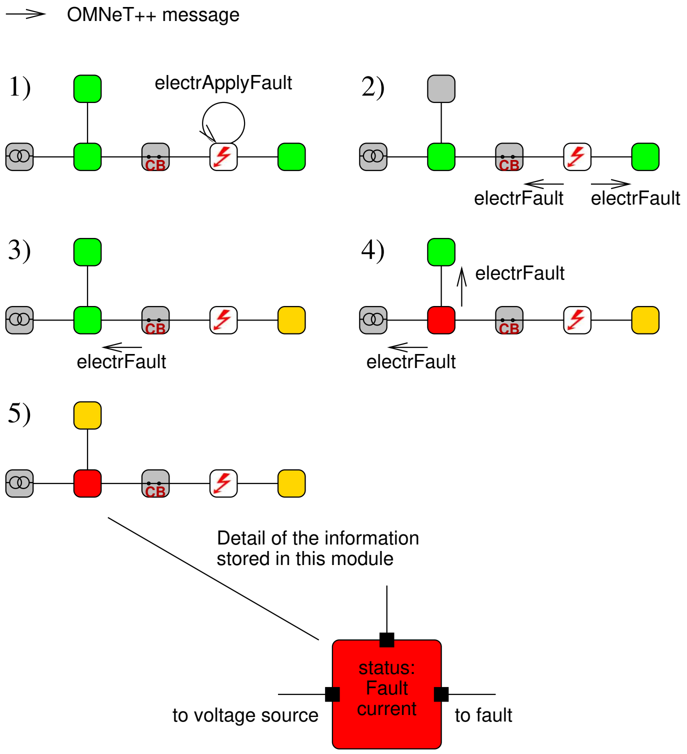


Figure 3. Steps (1)–(5) during the application of a fault.

2.1.3. Operation of a Switching Device

Switching device modules (circuit breakers and disconnectors) keep a record of which one of their two terminals is connected to a power source and/or to a fault.

- When a switching device opens and one of its terminals is connected to a power source, it sends an *electrSrcRem* message through the other terminal. The electrical module that receives this message actualizes its status to *Disconnected* and resends the message to all its connected electrical modules.

- When a switching device opens and one of its terminals is connected to a fault, it sends an *electrFaultRem* message through the other terminal. The electrical module that receives this message actualizes its status to *Normal* or *Disconnected*, depending on whether it is connected to a power source or not, respectively. The module then resends the message to all its connected electrical modules.
- When a switching device closes and one of its terminals is connected to a power source, it sends an *electrSrc* message through the other terminal. Downstream components react as described in the initialization, and the status of the grid is actualized.
- When a switching device closes and one of its terminals is connected to a fault, it sends an *electrFault* message through the other terminal. The process is then similar to the one described for applying a new fault.

Figure 4 shows the sequence of messages sent to open a circuit breaker.

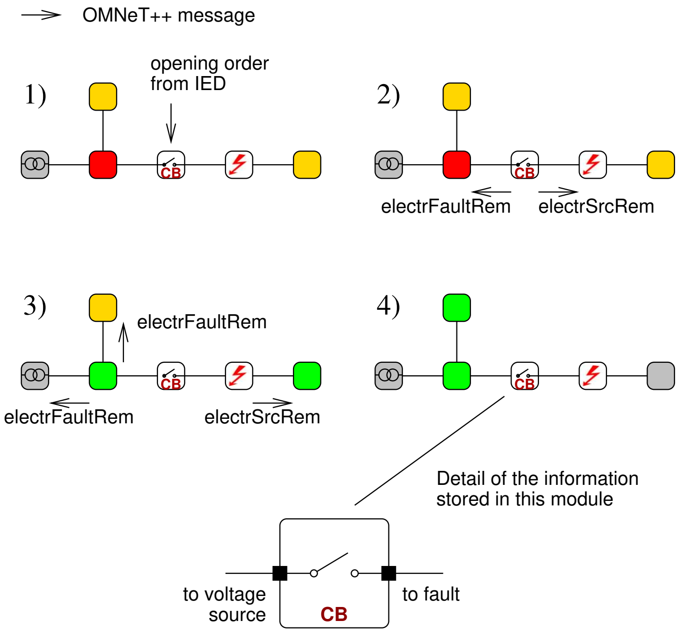


Figure 4. Steps (1)–(4) during the opening of a circuit breaker.

2.2. IEDs and Communication

In the simulator, each switching device is connected to an IED provided with protection or fault passage indicator functionalities, from which it receives messages, such as opening or closing orders, and to which it sends information such as the open or close status, the success or failure of switching operations, and the presence or absence of a normal voltage or fault current. Separate modules to model measuring and control devices have not been developed for the sake of simplicity, although they can be included. IED modules must be programmed to execute the logic that constitutes the actual fault location and isolation system. Communication between IEDs is defined in the simulator using OMNeT++ *in* and *out* gates.

Table 2. OMNeT++ messages used to calculate the status of the electrical network.

Message Name	Created by	Sent to	Purpose
electrSrc	• Power source • Closing switching device	All connected electrical modules	Identify busbars connected to the main grid, and any possible loops.
electrLoad	• Busbar • Closing switching device	Electrical modules connected upstream	Identify loads at all electrical connections.
electrApplyFault	• Faulted busbar	Self-message	Apply a fault.
electrFault	• Faulted busbar • Closing switching device	All connected electrical modules	Identify busbars affected by a fault and identify the path followed by the fault current.
electrSrcRem	• Opening switching device	All connected electrical modules	Identify busbars disconnected from the grid as a result of the opening of a switching device.
electrFaultRem	• Opening switching device	All connected electrical modules	Identify busbars disconnected from a fault as a result of the opening of a switching device.

3. Study Case

3.1. Base Case

The proposed simulation framework is applied to a decentralized protection system that relies on encrypted GOOSE messages between IEDs. The base case is a modified version of one of the medium voltage lines connected to the Il Violino substation, owned by Unareti and located in the Italian town of Brescia, that served as a demonstration site in the IDE4L project [18]. Figure 5 shows the single-line diagram of the system, which consists of a 15 kV line that feeds ten 15 kV/380 V secondary substations. The load and the number of customers connected to each substation are shown in Table 3.

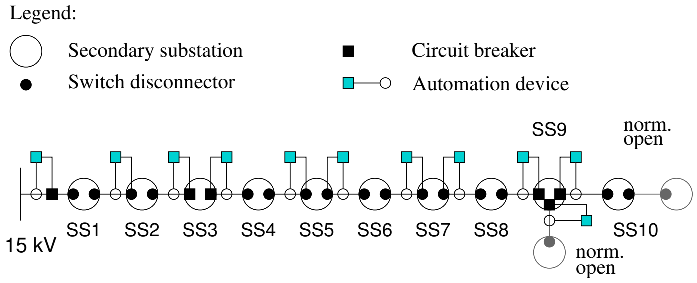


Figure 5. Study case.

Table 3. Study case loads and customers.

Substation	**SS1**	**SS2**	**SS3**	**SS4**	**SS5**	**SS6**	**SS7**	**SS8**	**SS9**	**SS10**
Load (kW)	456	103	91	120	591	73	184	57	73	57
Customers	102	4	22	1	18	15	39	4	38	19

Secondary substations, identified as SS1 through SS10, are connected to the upstream and downstream sections of the line by means of two switching devices. Some of these switching devices are circuit breakers that can be used to interrupt fault currents. The rest are disconnectors that can only be opened when there is no effective current across them. Circuit breakers are installed at the primary substation at the head of the line and at substations SS3 and SS9.

Two switching devices connect substations SS9 and SS10 to adjacent lines and provide operational flexibility by making it possible to reconfigure the network. These devices are normally open and do not intervene in the fault isolation process.

All circuit breakers, and some disconnectors, are automated and connected to IEDs that are responsible for the fault location and isolation functions. As can be seen in Figure 5, automated disconnectors have been placed on the distribution line so that there is always one automated switching device between each two secondary substations.

3.2. Two-Step Protection Scheme

The distributed protection system that has been modeled is based on the coordination of IEDs distributed along the medium-voltage feeder and has been implemented and tested as part of the IDE4L project [20]. The protection system acts in two steps. The first step involves circuit breakers and their corresponding IEDs and is responsible for the clearance of the fault. The second step involves switch disconnectors and their IEDs and reduces the area affected by the fault.

The principle of operation of the first step is as follows. Each breaker IED subscribes to messages from breaker IEDs located downstream of the same feeder. When a fault occurs, any breaker IED that detects the fault publishes a GOOSE *block* message. The only breaker IED that does not receive any block messages is the one nearest to the fault, and it is therefore the one that trips its breaker and isolates the fault. When a breaker IED that detects the fault receives a block message, it waits for a certain time for downstream protections to clear the fault; if the fault persists, it assumes that the protections located downstream have failed and trips its own breaker. The amount of time that breaker IEDs wait before tripping increases with the number of block messages received.

Figure 6 shows an example of the response of the protection system to a fault at Substation SS8. IEDs IED1, IED3 and IED4 participate in the first step. When the fault is detected, IED3 receives a block message from IED4, and IED1 receives block messages from IED3 and IED4. If CB3 opens successfully, the end of the first step is as shown in Figure 6c). If CB3 failed to open, the fault would be cleared by CB2 and, eventually, by CB1. A detailed description of the implementation of this system, including the mapping of the protection functions to the IEC 61850 data model, can be found in References [21,22].

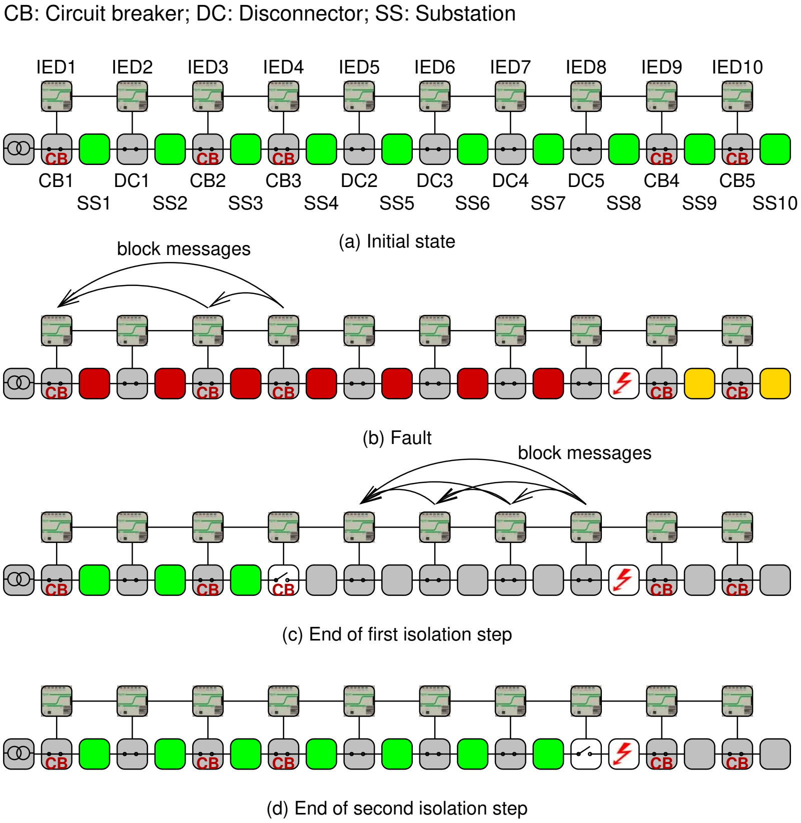


Figure 6. Two-steps isolation procedure.

The second step acts in a similar way but is performed by disconnector IEDs provided with a fault passage indicator (FPI) function, and begins once the first step has finished. Each disconnector IED subscribes to messages from disconnector IEDs located downstream and in the same section of the feeder between two circuit breakers. Once the fault is cleared, each disconnector IED that is connected downstream of the open circuit breaker, and that therefore may reduce the area affected by the fault by opening its disconnector, signals the presence of a fault and publishes a GOOSE *block* message.

Following a procedure similar to the first isolation step, an IED that receives a block message waits a certain time for a message from a downstream disconnector indicating that the fault has been isolated. If no such message is received, the IED assumes that the downstream disconnectors have failed to isolate the fault and sends its own disconnector an order to open. The opening of the disconnector reduces the area affected by the fault once the circuit breaker that originally cleared the fault re-closes.

In the example in Figure 6, IEDs IED5, IED6, IED7 and IED8 participate in the second isolation step. When the fault is detected, IED7 receives a block message from IED8; IED6 receives block messages from IED7 and IED8; and IED5 receives block messages from IED6, IED7 and IED8. If DC5 opens succesfully, the end of the first step is as shown in Figure 6d). If DC5 failed to open DC4, and eventually DC3 and DC2, would open.

This approach, based on subscription to messages from downstream IEDs, has the advantage of being easy to adapt to network reconfigurations in grids where the same load can be supplied from different feeders. The IDE4L project introduced a specific modeling to allow GOOSE subscription lists to be changed under operation [22] since this feature was not contemplated in IEC 61850. In the developed OMNeT++ simulation library, the user models the subscription of an IED to GOOSE messages from another IED by connecting an *out* gate of the publisher IED to an *in* gate of the subscriber IED.

4. Characterization of the Latency Times of Secured GOOSE Messages

GOOSE messages are OSI model (ISO/IEC7498-1) Layer 2 messages, originally defined for substation local communications. Thus, additional mechanisms are needed to secure and exchange GOOSE messages between substations over IP. In this work, Layer 2 Tunneling Protocol Version 3 (L2TPv3) over IPsec, as proposed in Reference [23], is used for tunneling and securing GOOSE communications over the Internet. These mechanisms impose both additional communication headers and processing times.

Figure 7 shows that the latency time is equal to the processing times in the routers plus the transmission time of the secured GOOSE messages over the Internet. The processing time in the routers is the time required to tunnel (by L2TPv3) and secure GOOSE messages by IPsec, encrypting and digitally signing them. The transmission time is the time taken to exchange secured messages (GOOSE over L2TPv3 over IPsec) between the routers.

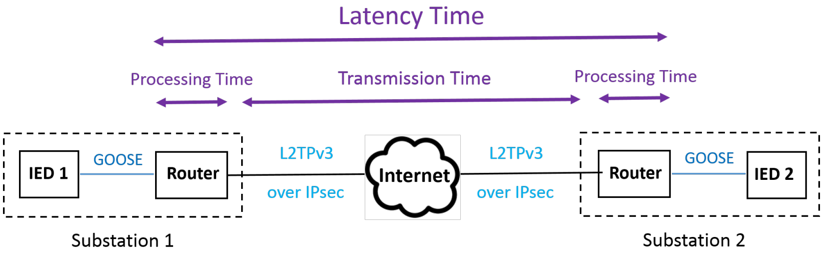


Figure 7. Transmission of secured Generic Object Oriented Substation Events (GOOSE) messages by L2TPv3 over IPsec.

To measure the latency time in the laboratory two routers are used, each of them connected to an IED as shown in Figure 7. Additionally, one desktop computer is used with two Ethernet ports (Eth1 and Eth2), each of which is connected to one of the routers. Each router has two connections on its LAN ports—one to an IED and one to Eth1/Eth2. The WAN side of the routers is configured for L2TPv3 over IPsec communication. Consequently, GOOSE messages published by the IEDs are securely exchanged over the Internet.

The network analyzer software Wireshark receives the GOOSE messages published by both IEDs on ports Eth1 and Eth2. Wireshark captures the GOOSE messages with their timestamps, port numbers (Eth1 or Eth2) and GOOSE protocol data units as described in the IEC 61850-8-1 standard. GOOSE message latency times can be calculated by subtracting the timestamps of GOOSE messages that have similar data units but different Ethernet port numbers (Eth1 or Eth2). GOOSE transmission data are captured in the laboratory and saved to a text file. This file is then analyzed using MATLAB to calculate the probability distributions of the latency times.

Figure 8 represents the distribution of the latency times obtained during an experiment in the laboratory. A total of 79558 packets were sent, of which 644 were lost and 14 had latency times larger than 300 ms. The figure also includes the best fit Normal, Weibull and Lognormal distributions. These results can be used later in the OMNeT++ simulations, where latency times can be modeled using the original histogram or a standard continuous probability distribution. Figure 9, for example, shows the probability plot corresponding to the best-fit Weibull distribution, which has a scale parameter of 31.7 ms and a shape parameter of 1.64 ms.

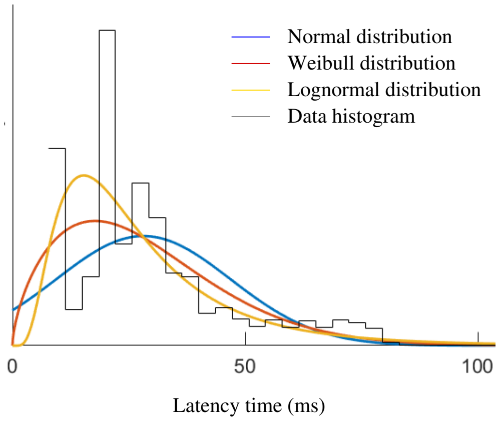


Figure 8. Histogram of measured latency times together with several adjusted probability distributions; in the Y axis, number of occurrences.

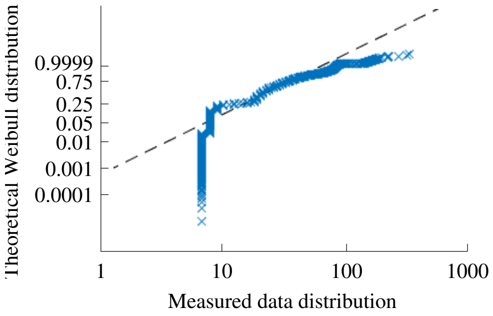


Figure 9. Probability plot for the Weibull (31.7,1.64) distribution.

5. Results

The simulator has been applied to the study case to assess the retransmission times of GOOSE messages and to analyze the effect of several faults on the system.

5.1. Validation of GOOSE Message Retransmission Times

Standard IEC 61850 specifies that GOOSE messages are to be retransmitted with varying and increasing retransmission intervals T_1, T_2 and T_3. After the fourth, retransmission messages are repeated with an interval, T_0, that corresponds to stable conditions. Specific retransmission times are not provided in the standard and depend on the specific implementation.

Retransmission times in the study case are constrained by the latency time of secured GOOSE messages and by the time in which the isolation procedure must be completed. Here, the latency time is modeled as a Weibull distribution with scale and shape parameters of 31.7 ms and 1.64 ms, respectively, as explained in Section 4. Setting the probability of a latency time at 95%, the quantile function for the Weibull distribution yields a value of 62 ms that is taken as the first retransmission time T_1.

Each step of the isolation procedure is set to be completed after 150 ms plus the time delay added by the circuit breaker, which is typically shorter than 100 ms. The second retransmission time T_2 is calculated as $T_2 = T_1 \times 1.3 = 81$ ms. As $T_1 + T_2 = 143$ ms is lower than 150 ms, there is time for two retransmissions during the isolation procedure, which is considered a sufficient guarantee that the message will arrive. T3 and T0 are set at higher values and are not considered relevant for the fault isolation procedure.

One concern when selecting the retransmission times is the possibility of saturating the IEDs, which can occur if the frequency of the arrival of GOOSE messages is too high compared with the time required to process them. This possibility is related to the number of IEDs communicating with each other because even if an IED is not subscribed to the sender of a GOOSE message, it has to process the message to determine the identity of the publisher. The selected retransmission times have been validated in the study case using the simulation tool. A total of 1000 simulations have been performed; each simulation covered 3 min and included a fault and the actuation of the two fault isolation steps. The queuing delay suffered by GOOSE messages due to the saturation of IEDs has been recorded in all the IEDs in the case. This queuing delay has been found to be small, only 1.8 ms on average, and has therefore been considered valid.

5.2. Performance after a Fault in Substation SS8

The study of a specific fault makes it possible to analyze its effect on the different substations and serves as a demonstration of the application of the simulator. Figure 6 shows an example of the main steps during the isolation of a short circuit in Substation SS8 in which the isolation procedure works as expected. Failures in switching devices, or in communications, may alter this sequence and result in additional losses of load. Failures in switching devices may include an opening failure of the circuit breaker between substations SS3 and SS4, a closing failure of the same circuit breaker, or an opening failure of the disconnector between substations SS7 and SS8. Failures in communications may include any loss of GOOSE messages between IEDs.

It is assumed in this case that the failure rate of switching devices is 3% (which is on the order of typical values according to Reference [24]) and that the failure rate of GOOSE messages between IEDs is 0.1%. Retransmission times are set as specified in Section 5.1.

Table 4 shows the average loss of load and customers in the secondary substations SS1 to SS7, as calculated after 1000 simulations. It can be seen that, by using the automated disconnectors along the feeder, the second isolation step substantially reduces the average loss of load and customers.

Table 4. Average losses after a short circuit in SS8.

Isolation Step	Load (kW)	Customers	Percentage
First	971.6	73.8	60.0
Second	81.2	7.4	5.0

Figure 10 shows the average loss of load at each secondary substation after the second isolation procedure. Substation SS7 is the most likely to suffer a loss of load because this loss can occur as a result of several circumstances: failure to open or close in the upstream breaker, failure to open in the upstream disconnector, or communication failure. Substation SS1, in contrast, only loses its load after a failure to open in both circuit breakers connected in substation SS3, which is very unlikely.

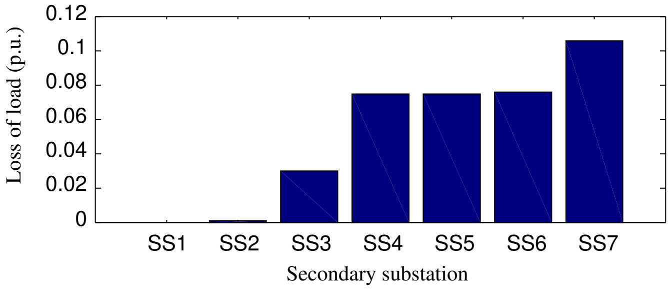


Figure 10. Average loss of load per substation in case 2.

Regarding the computation time, all 1000 simulations are completed in a computer with a 2.50 GHz processor in a few seconds, with a significant part of the time dedicated to store the results in the drive. The same analysis has been performed solving in parallel the electromechanical equations of the grid, using software PSSE, to quantify the reduction in computing time of the proposed approach. In this case the simulation time increases to 7.3 min, which means a difference of two orders of magnitude.

5.3. Reliability Analysis

Statistical analysis over repeated simulations makes it possible to evaluate the effect of different parameters on the performance of the decentralized fault isolation system. This section analyzes the effect of the failure rate of GOOSE messages and switching devices on the loss of load located upstream from the fault. The failure rate of GOOSE messages is defined as the rate of lost messages over the total number of sent messages, and the failure rate of switching devices is defined as the rate of failed opening or closing switching operations. Figure 11 shows the average loss of load in substations SS1 to SS7 when the failure rates of GOOSE messages and switching devices range from 0 (no failures) to 20%. Each point in the figure is obtained after 5000 simulations.

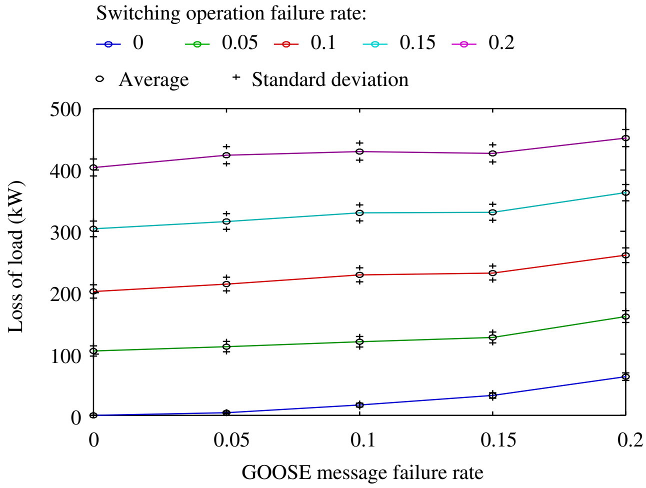


Figure 11. Reliability analysis.

It can be seen that the failure rate of GOOSE messages has a minimal effect on the loss of load. This result is due to the retransmission of GOOSE messages and to the setting of the retransmission times to allow two retransmissions before any action is decided on by the corresponding IED. Any GOOSE message has to be lost three times, which is very unlikely, before that loss has any effect on the fault isolation procedure.

Figure 11 also shows that the loss of load depends strongly on the failure rate of the switching devices, which is logical given that any switching operation failure results in additional loss of load.

5.4. Systematic Fault Analysis

This section applies the simulator to the systematic analysis of different fault locations along the feeder. Figure 12 shows the loss of load per unit at each secondary substation depending on the location of the fault. For each fault only substations located upstream from the fault are shown, because substations downstream from the fault are always disconnected. The failure rate of switching devices is set to 3% as before, and the failure rate of transmission of GOOSE messages is set to 0.1%.

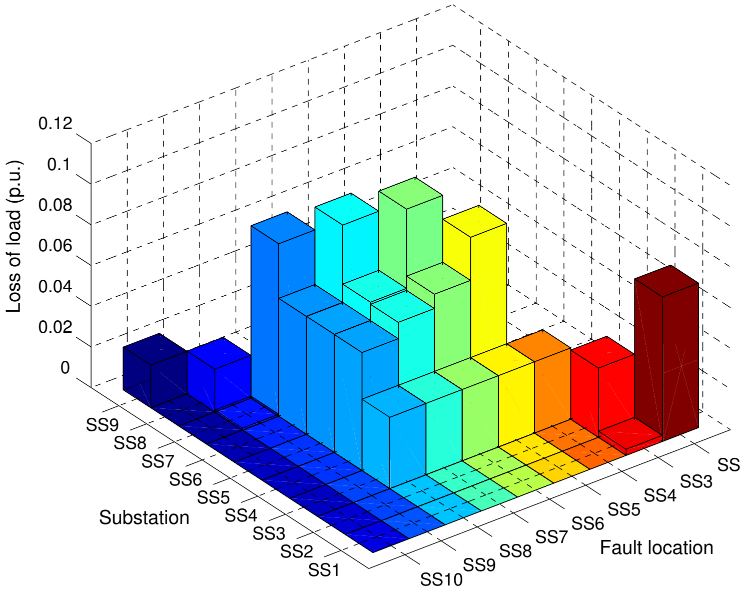


Figure 12. Average loss of load.

It can be seen that faults at substations SS5 to SS8 have the largest effect on the loss of load. This can be explained by the distribution of circuit breakers and disconnectors along the feeder. Looking at Figure 5, it can be seen that substations SS3, SS4, SS9 and SS10 are directly protected by a circuit breaker. Therefore, a circuit breaker must fail to open to produce any additional loss of load after the fault. On the contrary, substations SS5, SS6, SS7 and SS8 are protected by disconnectors. Therefore, the probability of an additional loss of load after a fault in these substations is increased because it can be the result of a failure in a circuit breaker or in a disconnector.

Figure 12 illustrates how circuit breakers, although typically more expensive than disconnectors, result in a more effective protection. A study like the one shown in Figure 12 can be used as the basis for an economic assessment of the decentralized protection system by assigning probabilities to the faults at each substation and penalties for the loss of load or customers.

6. Conclusions

The results from the experiment in the laboratory show that securing and tunneling GOOSE messages over the IP network increases their latency time to approximately 30 ms on average. This latency time, which is large compared to the 10 ms specified in Section 5 of the IEC 61850 standard for remote protection between substations (TT5 class), can be regarded as the price to pay for taking preventive action against cyberattacks aimed at disrupting a decentralized protection system between distant substations.

The application of the proposed simulation library makes it possible to translate the latency times obtained in the laboratory into the performance of a decentralized protection system that secures GOOSE messages by authenticating and encrypting packets of data using L2TPv3 over Ipsec. Results show that an acceptable compromise can be reached between the speed of response of the protection system and its resilience to communication failures. With respect to other simulation methods of electrical and communication networks, which are based on co-simulation on different platforms, the proposed method has the advantage of reducing the computation load by avoiding the solution of the non-linear set of equations that define the electrical network.

Author Contributions: Conceptualization, P.L., P.J., S.R., A.Á, F.R., D.D.G. and A.D.; formal analysis, P.L. and P.J.; software and writing, P.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the European Commission 7th Framework Program, grant number 608860.

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

References

1. Jamborsalamati, P.; Sadu, A.; Ponci, F.; Monti, A. Implementation of an agent based distributed FLISR algorithm using IEC 61850 in active distribution grids. In Proceedings of the 2015 International Conference on Renewable Energy Research and Applications (ICRERA), Palermo, Italy, 22–25 November 2015; pp. 606–611, doi:10.1109/ICRERA.2015.7418485. [CrossRef]
2. Eriksson, M.; Armendariz, M.; Vasilenko, O.O.; Saleem, A.; Nordstrom, L. Multiagent-Based Distribution Automation Solution for Self-Healing Grids. *IEEE Trans. Ind. Electron.* **2015**, *62*, 2620–2628, doi:10.1109/TIE.2014.2387098. [CrossRef]
3. Parikh, P.; Voloh, I.; Mahony, M. Fault location, isolation, and service restoration (FLISR) technique using IEC 61850 GOOSE. In Proceedings of the 2013 IEEE Power & Energy Society General Meeting, Vancouver, BC, Canada, 21–25 July 2013; pp. 1–6, doi:10.1109/PESMG.2013.6672862. [CrossRef]
4. Ling, W.; Liu, D.; Lu, Y.; Du, P.; Pan, F. IEC 61850 Model Expansion Toward Distributed Fault Localization, Isolation, and Supply Restoration. *IEEE Trans. Power Deliv.* **2014**, *29*, 977–984, doi:10.1109/TPWRD.2013.2289955. [CrossRef]
5. Uluski, R.W. Using distribution automation for a self-healing grid. In Proceedings of the PES T&D 2012, Orlando, FL, USA, 7–10 May 2012; pp. 1–5, doi:10.1109/TDC.2012.6281582. [CrossRef]
6. Aguero, J.R. Applying self-healing schemes to modern power distribution systems. In Proceedings of the 2012 IEEE Power and Energy Society General Meeting, San Diego, CA, USA, 22–26 July 2012; pp. 1–4, doi:10.1109/PESGM.2012.6344960. [CrossRef]
7. Caserza Magro, M.; Pinceti, P.; Rocca, L.; Rossi, G. Safety related functions with IEC 61850 GOOSE messaging. *Int. J. Electr. Power Energy Syst.* **2019**, *104*, 515–523, doi:10.1016/J.IJEPES.2018.07.033. [CrossRef]
8. Sun, C.C.; Hahn, A.; Liu, C.C. Cyber security of a power grid: State-of-the-art. *Int. J. Electr. Power Energy Syst.* **2018**, *99*, 45–56, doi:10.1016/J.IJEPES.2017.12.020. [CrossRef]
9. Spalding, R.A.; Rosa, L.H.L.; Almeida, C.F.M.; Morais, R.F.; Gouvea, M.R.; Kagan, N.; Mollica, D.; Dominice, A.; Zamboni, L.; Batista, G.H.; et al. Fault Location, Isolation and service restoration (FLISR) functionalities tests in a Smart Grids laboratory for evaluation of the quality of service. In Proceedings of the 2016 17th International Conference on Harmonics and Quality of Power (ICHQP), Belo Horizonte, Brazil, 16–19 October 2016; pp. 879–884, doi:10.1109/ICHQP.2016.7783370. [CrossRef]
10. Jinsong, L.; Dong, L.; Wangshui, L.; Zhibin, L. Study on Simulation and Testing of FLISR. In Proceedings of the CICED 2010 Proceedings, Nanjing, China, 13–16 September 2010.
11. Varga, A. OMNeT++. In *Modeling and Tools for Network Simulation*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 35–59. [CrossRef]
12. Palensky, P.; Van Der Meer, A.; Lopez, C.; Joseph, A.; Pan, K. Applied Cosimulation of Intelligent Power Systems: Implementing Hybrid Simulators for Complex Power Systems. *IEEE Ind. Electron. Mag.* **2017**, *11*, 6–21, doi:10.1109/MIE.2017.2671198. [CrossRef]
13. Hammad, E.; Ezeme, M.; Farraj, A. Implementation and development of an offline co-simulation testbed for studies of power systems cyber security and control verification. *Int. J. Electr. Power Energy Syst.* **2019**, *104*, 817–826, doi:10.1016/J.IJEPES.2018.07.058. [CrossRef]
14. Levesque, M.; Xu, D.Q.; Joos, G.; Maier, M. Co-Simulation of PEV coordination schemes over a FiWi Smart Grid communications infrastructure. In Proceedings of the IECON 2012-38th Annual Conference on IEEE Industrial Electronics Society, Montreal, QC, Canada, 25–28 October 2012; pp. 2901–2906, doi:10.1109/IECON.2012.6389434. [CrossRef]
15. Troiano, G.O.; Ferreira, H.S.; Trindade, F.C.; Ochoa, L.F. Co-simulator of power and communication networks using OpenDSS and OMNeT++. In Proceedings of the IEEE PES Innovative Smart Grid Technologies Conference Europe, Melbourne, Australia, 28 November–1 December 2016; pp. 1094–1099, doi:10.1109/ISGT-Asia.2016.7796538. [CrossRef]
16. Bhor, D.; Angappan, K.; Sivalingam, K.M. Network and power-grid co-simulation framework for Smart Grid wide-area monitoring networks. *J. Netw. Comput. Appl.* **2016**, *59*, 274–284, doi:10.1016/j.jnca.2015.06.016. [CrossRef]

17. Awad, A.; Bazan, P.; German, R. SGsim: Co-simulation framework for ICT-enabled power distribution grids. In *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*; Springer: Cham, Switzerland, 2016; Volume 9629, pp. 5–8, doi:10.1007/978-3-319-31559-1_2. [CrossRef]
18. Repo, S.; Ponci, F.; Della Giustina, D.; Alvarez, A.; Corchero Garcia, C.; Al-Jassim, Z.; Amaris, H.; Kulmala, A. The IDE4L Project: Defining, Designing, and Demonstrating the Ideal Grid for All. *IEEE Power Energy Mag.* **2017**, *15*, 41–51, doi:10.1109/MPE.2017.2662329. [CrossRef]
19. Ledesma, P. *OMNeT++ Electrical Network Simulation Library*; Online Repository of Universidad Carlos III de Madrid; Consorcio Madroño: Madrid, Spain, 2018, doi:10.21950/QXWDEL. [CrossRef]
20. Pegueroles, J.; Álvarez, A.; Ledesma, P.; Ramos, F. *Preliminary Assessments of the Algorithms for Network Reliability Improvement: Laboratory Verification of Algorithms for Network Reliability Enhancement by FLISR*; Technical Report; Catalonia Institute for Energy Research: Barcelona, Spain, 2016.
21. Della Giustina, D.; Dede, A.; de Sotomayor, A.A.; Ramos, F. Toward an adaptive protection system for the distribution grid by using the IEC 61850. In Proceedings of the 2015 IEEE International Conference on Industrial Technology (ICIT), Seville, Spain, 17–19 March 2015; pp. 2374–2378, doi:10.1109/ICIT.2015.7125448. [CrossRef]
22. Alvarez de Sotomayor, A.; Della Giustina, D.; Massa, G.; Dedè, A.; Ramos, F.; Barbato, A. IEC 61850-based adaptive protection system for the MV distribution smart grid. *Sustain. Energy Grids Netw.* **2017**, doi:10.1016/J.SEGAN.2017.09.003. [CrossRef]
23. Jafary, P.; Raipala, O.; Repo, S.; Salmenpera, M.; Seppala, J.; Koivisto, H.; Horsmanheimo, S.; Kokkoniemi-Tarkkanen, H.; Tuomimaki, L.; Alvarez, A.; et al. Secure layer 2 tunneling over IP for GOOSE-based logic selectivity. In Proceedings of the 2017 IEEE International Conference on Industrial Technology (ICIT), Toronto, ON, Canada, 22–25 March 2017; pp. 609–614, doi:10.1109/ICIT.2017.7915428. [CrossRef]
24. EPRI. *A Review of the Reliability of Electric Distribution System Components: EPRI White Paper*; Technical Report; EPRI: Palo Alto, CA, USA,2001.