

# *Advanced simulation platform for Smart Grid communications*

J. Juárez, C. Rodríguez-Morcillo, J. A. Rodríguez-Mondéjar

Electronics and Automatics Research Group

Institute for Research in Technology (IIT), ICAI School of Engineering, Comillas Pontifical University  
Madrid, Spain

javier.juarez@iit.upcomillas.es; carlos.rodriguez@iit.upcomillas.es; mondejar@dea.icaupcomillas.es

**Abstract**—Simulation is an important tool for study, analysis, design and evaluation of real world systems that we are unable to experiment with directly. Due to complexity of Smart Grids communication systems, it is necessary to use advanced simulation systems with the aim of accelerate the development and implementation of Smart Grids. In this paper, an advanced simulation platform (based on OMNET++) for Smart Grid is presented, and an application case for an IEC 61850 communication system is analyzed. The IEC 61850 simulation model used in this application is available on web.

**Keywords**—Simulation; Smart Grids; IEC 61850; OMNET++

## I. INTRODUCTION

The development and implementation of Smart Grids involves a set of aspects and considerations that make the system very complex and with a huge amount of data to manage, thus the network needs more decentralized intelligent in order to create an automated system able to achieve all the objectives that the Smart Grid demands. Therefore, communication systems are one of the key points in the Smart Grid development.

These communication networks include a huge amount of technologies (wireless, PLC, fiber optic...), protocols and standards. Therefore, the convergence of all of this technologies and protocols is a complex and important issue to solve.

Due to complexity of Smart Grids communication systems it is necessary to use advanced simulation systems, like it is presented here, with the aim of accelerate the development and implementation of Smart Grids. Besides, the use of simulation in teaching and training could be very interesting for Smart Grids development too.

One standard with high relevance in Smart Grids is the IEC 61850 standard [1]. Initially, the application area of this standard was substation automation system, but nowadays there are several studies about its application out of substations and its extension in other areas of Smart Grids, such as, for example, in Distributed Energy Resources.

In this paper, an advanced simulation platform for Smart Grid is presented, and an application case for an IEC 61850 communication system is studied. Despite this, this platform

can be extended to simulate other Smart Grids protocols, like DLMS/COSEM.

Some basic concepts about IEC 61850 are supposed for the readers of this paper.

This paper is structured as follows: Section II is a review about simulation of IEC 61850 systems. In Section III, some simulation tools are analyzed. Section IV describes the advanced simulation platform proposed. Section V exposes a case study. Section VI is about applications of the platform. Finally, Section VII concludes the paper.

## II. SIMULATION OF IEC 61850 COMMUNICATION SYSTEMS

Simulation is an important tool for study, analysis, design and evaluation of real world systems that are unable to experiment with directly. IEC 61850-based substation system simulation will be very useful due to the fact that substations are complex and expensive systems; and having a platform to simulate and validate different architectures, validate object modeling and services implementation into devices, and develop IEC 61850 applications, could be very interesting.

It is important to note that one of the types of simulation more used in communication networks is the Discrete Event Simulation (DES), mainly because this paradigm of simulation adapts very well to these systems and it is easy to implement. In general, DES provides an easy, flexible and with high repeatability way to study the behavior of these systems under different conditions [2]. Thus, for example, DES could be used to simulate a complex network that, in real world, will be too difficult to have.

In the literature can be found several works about DES in IEC 61850-based Substation Automation Systems (SAS). Some of the most important are:

- Reference [3] studies the performance of a switched hub Ethernet network of 10Mbps and 100Mbps with COMNET III simulator.
- Reference [4] demonstrates that Ethernet meets with requirements that SAS requires from the point of view of delays allowed. It uses OPNET simulator.
- Reference [5] is a very well explained paper that exposes how to create and setup a platform for the performance study of an IEC 61850 based substation

communication system by modeling of devices with the specific protocol stack. It uses OPNET simulator.

- Reference [6] proposes a network architecture for SAS reliable, fast and deterministic, focusing on an Ethernet-based logical architecture. The network nodes, such as IEDs, switches, workstations, and servers, have been modeled according to [5]. It uses OPNET for simulation.
- Some papers [7], [8] use the open-source simulator NS-2 to evaluate different issues about IEC 61850.
- Other works that use OPNET simulator to evaluate some characteristic of IEC 61850 are [9-11].

Most of papers presented above have been oriented to evaluate the performance or other characteristics of an IEC 61850 based communication network and to demonstrate that the IEC 61850 requirements are fulfilled. But the simulation results are very sensible to the accuracy of the message estimation, among other things, so could be difficult to evaluate in a realistic way some specific situations. Therefore, a detailed and flexible simulation platform, more real with respect to algorithms and real messages format that devices implement, will provide a lot of advantages, like the possibility of simulate more concrete situations to realize more complete cases that help to the design of SAS architectures and devices. Besides, DES provides a simple, fast, with high repeatability and flexible way to test these designs.

Apart from DES, there are other important simulation techniques, such as real-time simulation, hybrid simulation, emulation and hardware-in-the-loop (hardware-in-the-loop means that some external device is taking part in the simulation), or parallel and distributed simulation.

Combining these simulation techniques in a platform will allow us having a simulation more flexible, complete and detailed. Therefore, simulation could go from simple message exchange analysis to device configuration evaluation; or in other words, from low detail to high detail. Thus, it will be possible to carry out more real simulations or simple and fast simulations, depending on whatever user wants and situation requires, providing very high flexibility.

In next section, some simulation tools are analyzed in order to study how these simulation techniques could be used together.

### III. SIMULATION TOOLS

In academic and research area is possible to find a large number of network simulators. Historically, the network simulator NS-2 [12] has been one of the most used by researchers, but today this simulator presents a set of disadvantages, like its lack of scalability for example, so nowadays, a new generation of simulators are achieving a lot of relevance.

With the aim of develop the simulation platform, it has been taken into account these simulators:

- OPNET [13]: Some characteristics of this simulator will be analyzed in more detail below.

- COMNET III: Simulator used in [3] in order to study the network features.
- NS-2: As mentioned above, one of the most widely used network simulator in research area.
- NS-3 [14]: It is a network simulator implemented in pure C++ that pretends to replace NS-2 for networking research.
- OMNET++ [15]: this simulator will be analyzed below.
- JiST [16]: A discrete event simulator which allows the implementation of networks simulations in JAVA.
- SimPy [17]: A process-oriented discrete event simulator written in Python.

There are several studies that evaluate different characteristics and performance of simulators [18-21]. Reference [18], compare five open-source simulators: OMNET++, Jist, SimPy, NS-2 and NS-3, and concludes that NS-3 has the best overall performance, but NS-3 is under development so a lot of models should still be ported from NS-2 to NS-3. On the other hand, this study considers OMNET++ as a viable alternative.

As has been observed in section above, OPNET simulator is one of the most used for IEC 61850 simulation. This simulator has probably the largest protocol model library.

With respect to OMNET++, it is a very well documented, open-source, and well-structured simulator.

Some characteristics about OMNET++ and OPNET are [22]:

- OPNET and OMNET++ have enough models to simulate an IEC 61850 network.
- OPNET simulator is more used than OMNET++ in IEC 61850 simulation studies.
- OPNET models are open-source, but simulator does not. OMNET++ models and simulator are open-source.
- OPNET is written in C, OMNET++ in C++.
- Since these simulators use general purpose programming languages, such as C or C++, it is possible to implement real code into them.

Without analyzing economic issues, it was decided to use OMNET++ simulator with the aim of achieve the same performance that OPNET simulator, but with the advantage of that OMNET++ is open-source and it can be modified and adapted to realize more advanced simulations, such as for example:

- Parallel and distributed simulation: If simulation requires a huge amount of memory, distribute it over a cluster may be a suitable solution.
- Real Time Simulation: The OMNET++ distribution contains a scheduler for Real Time simulation that synchronizes the simulation time to the system clock.

- **Hardware-in-the-loop and emulation:** It is possible to capture packets from real network devices, and insert them into the simulation. An application example of this can be found in [23].
- **Hybrid simulation:** Such as interfacing OMNET++ with other simulators (hybrid operation).

In next section, the simulation platform (based on OMNET++) that combines all of these simulation concepts will be presented.

#### IV. ADVANCED SIMULATION PLATFORM

In order to describe the advanced simulation platform developed, it can be structured by two main components: the simulation model, and the simulation core.

The simulation model describes the behavior of the network, that is, describes the devices, protocols and applications, and the way in which all of them interact inside the network. In case of IEC 61850, the protocol stack, message format, applications and devices behavior are implemented here.

The simulation core describes the way in which the simulator uses the simulation model in order to evaluate it. Simulation core in OMNET++ is based on discrete event simulation. The idea of the DES is to jump from one event to the next, whereby the occurrence of an event may trigger changes in the system state as well as the generation of new ones, called future events. An event is composed for, at least, two data: time and type, where *time* tag specifies the time when the event will occur, and *type* tag gives the kind of the event that specify the routine to be executed. Therefore, the simulation core is composed by two elements: an event list, where the events are stored, and a scheduler, that selects the next event in the event list to be executed.

In next sub-sections will be presented the OMNET++ simulation model for IEC 61850 networks used, and a redefined simulation core whose new scheduler uses different techniques, like parallel simulation and real time simulation, in order to realize more complete and complex simulations.

##### A. IEC 61850 simulation model for OMNET++

To design the simulation model for IEC 61850-based substation system has been used the OMNET++ INETMANET framework, which is an open-source communication networks simulation package that contains several models needed for IEC 61850 communication model, including TCP, IP, Ethernet and many others.

The IEC 61850 model used has been developed and can be downloaded from [24]. It has been tested with OMNET++ version 4.1 in Ubuntu 10.04 LTS Lucid Lynx operating system.

The IEC 61850 communication model is based on Ethernet technology. For Core ACSI services, TCP, IP and Ethernet layers with some modifications have been imported from INETMANET framework; and MMS OSI layers (from RFC

1006 layer to MMS application layer) have been imported from an external library.

One important module developed in this model is the Sniffer module, which has the function of monitor all the traffic that passes through it and record the messages in a PCAP file in order to analyze them with Wireshark, which is an open-source network protocol analyzer. This module will allow to check the correct format of the messages and to evaluate in detail what happen in the network simulation easily.

This simulation model uses the real format of messages (from the point of view of application layer), so it will be possible to realize more realistic and advanced simulations (combining emulation, hardware-in-the-loop...).

##### B. Simulation core

As seen before, the main elements of the simulation core are the event list and the scheduler.

OMNET++ has three schedulers defined, and allows the implementation of new ones.

The basic scheduler defined is the cSequentialScheduler, whose behavior is simple: it works in a loop in which selects the first element in the event list, executes the routine of this event, and deletes it from the event list.

The second scheduler is cRealTimeScheduler. It uses wait calls that synchronize the simulation time to the system clock.

This scheduler has some limitations, because a simulation with a huge amount of modules could limit the use of this type of simulation. This limit is given by the ratio between simulation time and real time, which has to be near to 1. Fig. 1 illustrates this concept.

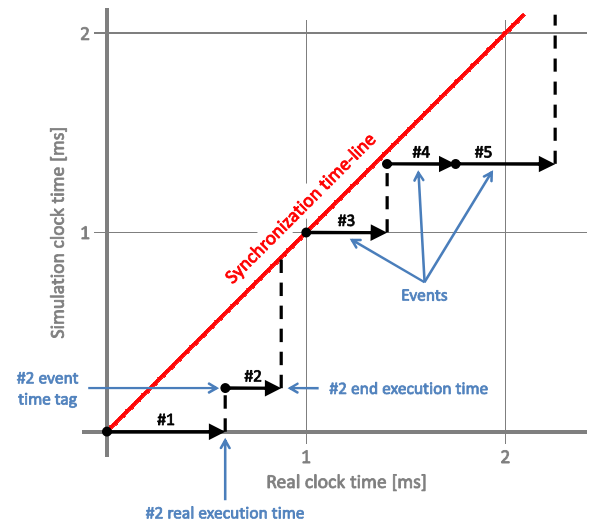


Fig. 1. Simulation time vs real time.

The Synchronization time-line represents when simulation time and real time are completely synchronized, but this is unrealistic since OMNET++ is a discrete event simulator, and sometimes there will be a delay between simulation and real time due to the CPU only allows a limited number of events to be handled in real-time.

The third scheduler is cSocketRTScheduler. Its behavior is similar to cRealTimeScheduler, but during wait calls, the scheduler waits for incoming messages from an external device using raw sockets, and when a message arrives, the scheduler inserts it in the event list. Events can send messages to the real network too. This scheduler allows to use the simulator for emulation and hardware-in-the-loop.

Due to these schedulers are sequential, there are some problems with synchronization between simulator and the real network device, and delays sending or receiving messages could make the results unrealistic.

A new simulation core has been created in order to achieve more realistic, reliable and scalable simulation, solving the above problems. Besides, it can be modified easily with the aim of adapt its behavior to real necessities of systems.

This simulation core uses two processes working in parallel, so a computer with two processors is required, but it can be extended to use more cores. The first process manage the main simulation, it means, the event list and the main scheduler. The second process manages the connection with the real network. In Fig. 2 is shown the structure of this simulation core.

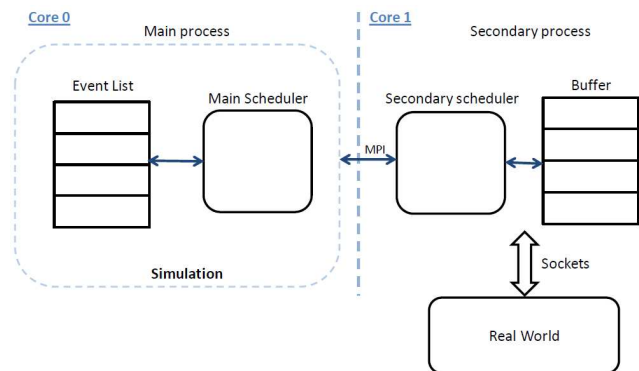


Fig. 2. Simulation core structure.

The communication between cores has been made with Message Passing Interface (MPI) that allows processes of different cores to communicate with other ones by sending and receiving messages.

The main scheduler behavior is similar to cRealTimeScheduler, but also it monitors if there are messages to send to real network (when an event is inserted in the event list and its destination is the real network, it is extracted from the list and passed to the secondary process by an MPI send call), and if a new message arrives from the secondary process (by an MPI receive call), it is inserted in the event list and executed at specific simulation time.

The secondary scheduler monitors if a message arrives from raw sockets or from MPI, and process it. When a message arrives from real network, it is send to the main process at real time and it is inserted in the event list with the specific simulation time. If a message arrives from main process to be sent to real network, the scheduler look if the execution time is the current time and send it; if not, the message is stored in a buffer and sent at correct time, and no delay is present.

The unique restriction (due to it could be a delay between simulation time and real time as seen in Fig. 1) in order to ensure the synchronization between real world and simulation is that the creation time of an event (time in which it is inserted in event list) have to be less or equal than the real time to be scheduled.

## V. CASE STUDY

A simplified case study is analyzed here in order to demonstrate the powerful of the simulation platform, and to give an idea about how simulation techniques can be used. This case is incomplete from the point of view of IEC 61850 functionality (functions like automatic reclosing, breakers failure or alarms, for example, are out of the case), but with this simple example, the potentiality of the platform can be understand easily.

A concrete substation to simulate is presented and a possible Logical Node (LN) configuration explained. A fault case will be presented from the point of view of Generic Object Oriented Substation Events (GOOSEs) exchange.

### A. Substation architecture and LN configuration

The substation under study (Fig. 3) will be composed by two transformers working in parallel (T1 & T2) feeding three loads (F1, F2 & F3); two with high priority and one with low priority, which means that in an anomalous condition of operation (one transformer inoperative for example), the low priority load could be disconnected.

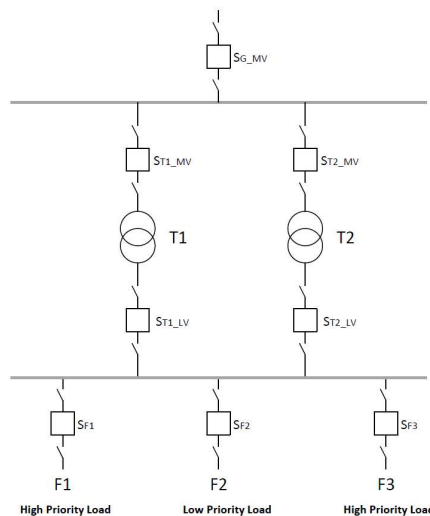


Fig. 3. Substation to simulate.

Due to flexibility and adaptability that IEC 61850 provides, there are a lot of solutions for the same problems from the point of view of LN configuration, and to choose one option or another depends on functionality expected, physical constraints and other variables with respect to design issues. There are some works [25], [26] about how to design IEC 61850 systems and how to configure devices. This is not the purpose of this section; only some LNs involved in this example are exposed in order to understand better how the platform can be used.

Fig. 4 and Fig. 5 show the LN configuration of devices in the feeder and the transformer bay.

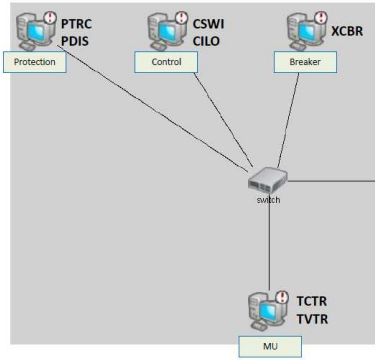


Fig. 4. LN configuration for a feeder bay (F1, F2 & F3).

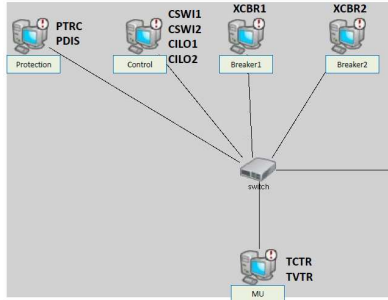


Fig. 5. LN configuration for a transformer bay (T1 & T2 bays). The LNs involved are:

- TVTR: Current transformer.
- TCTR: Voltage transformer.
- XCBR: Circuit breaker. This LN is used for modeling switches with short circuit breaking capability.
- PTRC: Protection trip conditioning. This LN shall be used to connect the “operate” outputs of one or more protection functions to a common “trip” to be transmitted to XCBR.
- PDIS: Distance protection. Distance relay is a relay that functions when the circuit admittance, impedance, or reactance increases or decreases beyond a predetermined value.
- CILO: Interlocking. This LN shall be used to “enable” a switching operation if the interlocking conditions are fulfilled.
- CSWI: Switch controller. This LN shall be used to control all switching conditions above process level.

### B. Example

With the above configuration, one fault case will be analyzed from the point of view of what switches will be open and what messages exchanged.

In this simulation case, switch  $S_{F2}$  (Breaker in Fig. 4) is a real device connected to the simulator platform through an Ethernet connection. From the point of view of this device, it sees the substation in a transparent way, so it does not know if substation is simulated or real. Therefore, the behavior of this device can be tested in any situation, and reaction of other devices in response to its actions can be evaluated too.

The fault is located between  $T1$  and  $S_{T1\_MV}$ , and causes that switches  $S_{T1\_MV}$  and  $S_{T1\_LV}$  open in order to isolate the fault, so only transformer  $T2$  will be operative to ensure the continuity of supply. Besides this and in these conditions,  $S_{F2}$  will be open to disconnect the low priority load.

The logic of the LNs, in order to detect the fault and actuate against it, is not detailed here; just message exchange is analyzed:

- Initially, in  $T1$  bay, the Merging Unit (MU) is sending Sampled Values (SVs) (at sampling rate of 960 samples/s) to Control & Protection devices with the measurements of the current and voltage transformers.
- When the fault occurs, PDIS detects it and PTRC generates a trip message (GOOSE message) that is sent to XCBR1 in Breaker1 and XCBR2 in Breaker2.
- The breakers open and their status information changes from ON to OFF. Two new GOOSEs per breaker are sent to Control and Protection devices with the aim of notify them the new states of the breakers.
- Then, Control sends one GOOSE to Control device in  $F2$  bay in order to indicate that the breaker should be open (to disconnect the low priority load).
- In  $F2$  bay, CSWI, in control device, generates a GOOSE message, and sends it to XCBR in Breaker device.
- The breaker opens, and its status information changes from ON to OFF. This new state is sent (by a GOOSE message) to Control and Protection devices with the aim of notify them this change.

### C. Hardware-in-the-loop simulation and results

The simulation platform is composed by one PC (Intel Core 2 Quad Processor Q8300) with four cores of 2.50 GHz connected with a real device through an Ethernet switch (see Fig. 6). This external device simulates the functions of a breaker in a substation, and it is based on IEC 61850 standard.

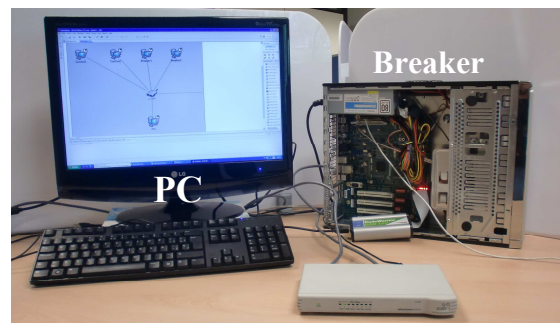


Fig. 6. Hardware-in-the-loop simulation platform.

From the point of view of the results, the messages exchanged with the device have been carried out without any delay and at correct times.

The maximum delay for messages exchanged has been analyzed, and it has been of 0.03ms (GOOSE of 32 bytes) for

the worst case, so the IEC 61850 requirements are fulfilled without problems.

## VI. APPLICATIONS

Due to fast advance of computing capacity, the simulation platform developed will allow to realize more complete, realistic and complex simulations. For example, is possible to use the platform in combination with grid computing in order to simulate very huge models, which will be very useful for Smart Grid simulations.

An interesting case to study and simulate could be the application of the IEC 61850 for Distributed Energy Resources.

Another application is shown in Fig. 7, where a complete substation could be modeled distributing the substation bays in different computers.

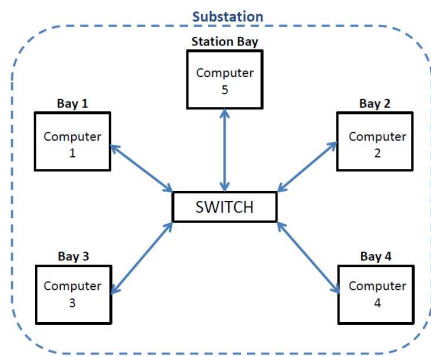


Fig. 7. Grid computing case.

Due to the existence of a large number of communication protocols in Smart Grids, it will be interesting to extend the simulation models with them in order to study integration issues, for example.

## VII. CONCLUSION

An advanced simulation platform for Smart Grid has been presented, and an application case for an IEC 61850 communication system analyzed. The IEC 61850 simulation model used in this application is available on web [24].

The simulation platform developed will allow to realize more complete, realistic and complex simulations, due to it allows to distribute, in a synchronized way, the simulation, by the application of simulation techniques such as parallel simulation, emulation and hardware-in-the-loop.

Its use in Smart Grids will allow to realize modeling and performance analysis, and to study in deep the behavior of Smart Grid protocols, comparison and testing of different applications, and protocols integration.

## REFERENCES

[1] IEC 61850, «Communication networks and systems in substations», 2003.  
 [2] K. Wehrle y J. Gross, *Modeling and Tools for Network Simulation*. Springer, 2010.

[3] IEC 61850-5, «Communication networks and systems in substations - Part 5: Communication requirements for functions and device models». 2002.  
 [4] T. Skeie, S. Johannessen, y C. Brunner, «Ethernet in substation automation», *Control Systems Magazine, IEEE*, vol. 22, n° 3, págs. 43-51, 2002.  
 [5] T. S. Sidhu y Yujie Yin, «Modelling and Simulation for Performance Evaluation of IEC61850-Based Substation Communication Systems», *Power Delivery, IEEE Transactions on*, vol. 22, n° 3, págs. 1482-1489, 2007.  
 [6] M. S. Thomas y I. Ali, «Reliable, Fast, and Deterministic Substation Communication Network Architecture and its Performance Simulation», *Power Delivery, IEEE Transactions on*, vol. 25, n° 4, págs. 2364-2370, 2010.  
 [7] H. S. Yang et al., «Gigabit Ethernet Based Substation», *Journal of Power Electronics*, vol. 9, n° 1, pág. 100-108, 2009.  
 [8] U. Premaratne, J. Samarabandu, T. Sidhu, R. Beresh, y Jian-Cheng Tan, «Security Analysis and Auditing of IEC61850-Based Automated Substations», *Power Delivery, IEEE Transactions on*, vol. 25, n° 4, págs. 2346-2355, 2010.  
 [9] P. M. Kanabar, M. G. Kanabar, W. El-Khattam, T. S. Sidhu, y A. Shami, «Evaluation of communication technologies for IEC 61850 based distribution automation system with distributed energy resources», in *Power & Energy Society General Meeting, 2009. PES '09. IEEE*, 2009, págs. 1-8.  
 [10] T. S. Sidhu, S. Injeti, M. G. Kanabar, y P. P. Parikh, «Packet scheduling of GOOSE messages in IEC 61850 based substation intelligent electronic devices (IEDs)», in *Power and Energy Society General Meeting, 2010 IEEE*, 2010, págs. 1-8.  
 [11] C. R. Ozansoy, A. Zayegh, y A. Kalam, «Time synchronisation in a IEC 61850 based substation automation system», in *Power Engineering Conference, 2008. AUPEC '08. Australasian Universities*, 2008, págs. 1-7.  
 [12] NS-2 Home Page, «<http://www.isi.edu/nsnam/ns/>», 2010.  
 [13] OPNET modeler, «[http://www.opnet.com/solutions/network\\_rd/modeler.html](http://www.opnet.com/solutions/network_rd/modeler.html)», 2010.  
 [14] NS-3 Home Page, «<http://www.nsnam.org/>», 2010.  
 [15] OMNeT++ Home Page, «<http://www.omnetpp.org/>», 2010.  
 [16] JIST Home Page, «<http://jist.ece.cornell.edu/>», 2008.  
 [17] SimPy Home Page, «<http://simpy.sourceforge.net/>», 2010.  
 [18] E. Weingartner, H. vom Lehn, y K. Wehrle, «A Performance Comparison of Recent Network Simulators», in *Communications, 2009. ICC '09. IEEE International Conference on*, 2009, págs. 1-5.  
 [19] J. Lessmann, P. Janacik, L. Lachev, y D. Orfanus, «Comparative Study of Wireless Network Simulators», in *Networking, 2008. ICN 2008. Seventh International Conference on*, 2008, págs. 517-523.  
 [20] Xiaodong Xian, Weiren Shi, y He Huang, «Comparison of OMNET++ and other simulator for WSN simulation», in *Industrial Electronics and Applications, 2008. ICIEA 2008. 3rd IEEE Conference on*, 2008, págs. 1439-1443.  
 [21] L. Begg, W. Liu, S. Pawlikowski, S. Perera, y H. Sirisena, «Survey of simulators of Next Generation Networks for Studying Service Availability and Resilience». Department of Computer Science and Software Engineering, University of Canterbury, 2006.  
 [22] A. Varga, «The OMNET++ discrete event simulation system», in *Proceedings of the European Simulation Multiconference*, 2001, págs. 319-324.  
 [23] T. Staub, R. Gantenbein, y T. Braun, «VirtualMesh: an emulation framework for wireless mesh networks in OMNeT++», in *Proceedings of the 2nd International Conference on Simulation Tools and Techniques*, ICST, Brussels, Belgium, Belgium, 2009, pág. 64:1-64:8.  
 [24] J. Juárez, «Webpage of Javier Juárez Montojo». [Online]. Available: <http://www.iit.upcomillas.es/jjuarez/>. [Accessed: 04-Abr-2011].  
 [25] K. P. Brand, C. Brunner, y W. Wimmer, «Design of IEC 61850 based Substation Automation Systems according to customer requirements», in *CIGRE Plenary Meeting, Paris*, 2004, pág. 195-04.  
 [26] A. Apostolov, C. Brunner, y K. Clinard, «Use of IEC 61850 object models for power system quality/security data exchange», in *Quality and Security of Electric Power Delivery Systems, 2003. CIGRE/PES 2003. CIGRE/IEEE PES International Symposium*, 2003, págs. 155-164.