

Review

A Critical Review of IEC 61850 Testing Tools

Taha Selim Ustun 

Fukushima Renewable Energy Institute (FREIA), Advanced Industrial Science and Technology (AIST), Koriyama 963-0298, Japan; ustun@ieee.org

Abstract: Smartgrid technologies necessitate the use of information technologies (IT) and communication in power system networks. There are different ways of integrating power system equipment in the communication layer for successful information exchange. IEC 61850 offers standard support object-oriented modeling and standardized parameter declaration. This lends itself to the diverse nature of power systems and supports plug-and-play (PnP) operation in smartgrids. Considering the amount of time that is invested in customizing non-PnP communication networks, this is a huge advantage and the main reason behind the popularity of IEC 61850. In line with this popularity, the body of research regarding this standard is constantly growing. In order to test the developed IEC 61850 models and messages, various tools are required. Researchers operate with a limited budget and have to know the abilities and limitations of such tools before making a procurement decision. This paper provides a critical review of IEC 61850 testing tools available in the market. It compares them in terms of their abilities, technical superiority and customer experience, including delivery time and customer support. Researchers in this field will benefit significantly from this work when making procurement decisions based on their needs.

Keywords: smart grid communication; power system modeling; standardization; simulation; device emulators; research tools; Hardware-In-the-Loop (HIL) testing; system in the loop tests



Citation: Ustun, T.S. A Critical Review of IEC 61850 Testing Tools. *Sustainability* **2021**, *13*, 6213. <https://doi.org/10.3390/su13116213>

Academic Editors: Daisuke Mashima, Subhash Lakshminarayana, Prosanta Gope and Neetesh Saxena

Received: 27 April 2021
Accepted: 28 May 2021
Published: 31 May 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Recently, communication has become indispensable for ensuring the safe operation of power systems [1]. This is due to the fact that new age power systems have more dynamic equipment and operate in a more interactive fashion [2]. Coupled with the novel technologies pushed forward by clean energy policies, e.g., renewable energy-based generators and electric vehicles, the volume of message exchanges has increased significantly [3]. The unprecedented behavior of such new technology disrupts well-established control schemes such as voltage control and protection [4].

One way to tackle this issue is by using communication lines to provide more eyes and ears around the network [5]. More observability paves the way for more efficient and effective control of the power system infrastructure [6]. However, this comes at a cost. Connecting all these devices which belong to different levels of power system operation is a daunting, if not impossible, task [7].

The only viable way to achieve a fully connected smartgrid is by using standardized communication modeling and message exchanges. In this fashion, regardless of the make, model or manufacturer, a device can be successfully modeled with a single projection to exchange the necessary information. This concept is called interoperability and is at the heart of future smartgrid communication systems [8]. There have been different efforts made towards developing a power system communication standard. The most popular and promising is the IEC 61850 communication standard [9]. In spite of being introduced as a substation automation standard initially, it quickly outgrew this mandate and is now utilized to model different equipment such as smart meters, virtual power plants, electric vehicles (EVs) and different novel components [10].

There are several reasons for this uptake. Firstly, IEC 61850 has a robust abstracting system that allows object-oriented modeling. This means complex devices, as well as systems, can be modeled in a fairly simple manner by defining variables and grouping them inside objects [11]. The other reason is that IEC 61850 has efficient message exchange protocols that are developed for the different needs of power system operation [12]. In other words, once a communication model is developed with this standard, the relevant message exchange protocols come ready-for-use. Thirdly, the systematic use and expansion of IEC 61850 standard enables system-wide integration between devices that may be located in very different domains. For instance, a smart load or an EV located in a residential house can be integrated with a frequency control scheme run by the system operator [13]. This holistic development also has advantages in other fields such as cybersecurity. While increased connectivity provided by IEC 61850 creates different vulnerabilities, its complementary standard IEC 623531 provides a full package for mitigating these [14].

This popularity shows itself in the literature. Researchers, companies and standard developers are constantly looking at different ways of using IEC 61850 modeling [15], investigating the performance of its messages for different applications and setups while recommending revisions and extensions [16]. This collective effort is one of the reasons why IEC 61850 standard is very strong and operates at a fast pace in following up with the latest developments [17].

The research work requires development of different IEC 61850 communication models, their implementation in a lab set up, exchange of real information on messages, and observation of communication performance, as well as impact on the power system operation [18]. Due to its intricate nature, this research field requires its own development and testing tools.

There are different testing tools available, ranging from open source to licensed software. Some of these tools are developed by researchers and engineers which come with no warranty [19,20], while others are freshly developed by companies and are still a beta version [21]. This may mean, after spending very limited research funds and countless hours, researchers may find out that a particular tool may not be fit for their needs. Considering that research funds are very limited, and researchers work with tight schedules, it becomes more evident how important it is to find a suitable and operational tool for the right purposes.

There is a consensus in the literature on this issue with many research papers reviewing the abilities of different tools, comparing and contrasting them with different parameters. For instance, in [22] a review of open-source tools for photovoltaic system modeling is given. In [23], a similar comparison is done for tools that model electric vehicles and their impacts on power systems. Focusing on communication between electric vehicles requires the study of ad-hoc networks and tools that can be used in such studies, and are given in [24].

Renewable energy potential estimation is a very popular topic in the literature. Estimating wind profile or the temperature at a certain site is very important. Tools pertaining to these parameters are reviewed in [25] and [26], respectively. Once this is done, it is possible to use different software to design district and urban energy systems. The tools have different capabilities and focus which are studied in [27,28] in great detail. If the research objective is developing a full-fledged renewable energy project with optimization and sizing, it is possible to refer to different publications [29–31] for an in-depth comparison of relevant tools and their capabilities.

Analysis of research tools and their capabilities holds such an important part in the literature and the general body of knowledge. For this reason, there are tool comparison studies, albeit in a very narrow area. For instance, work in [32] gives an analysis of tools directed at studying decarbonization of wastewater plants, while [33] reviews microwave-assisted gasification of biomass. Studying and documenting research tools is such a universal approach that it is done virtually in every field, e.g., for automatic test equipment [34], software for project management [35] and web-based simulation tools [36].

All of these works benefit the literature and researchers by documenting the strengths and weaknesses of different tools. Readers can benefit from the experience of other researchers and judge the suitability of a certain research tool for their purposes with a minimal time dedication. This increases research productivity and efficiency for everyone.

Despite the significance of tools revision for research, there is still no work that critically reviews IEC 61850 testing tools, compares their capabilities and gives insights to researchers about their use. *This paper addresses this very important knowledge gap.* It is based on the extensive research experience of the authors working in the IEC 61850 research field for many years. Different tools pertaining to different aspects of IEC 61850 research are presented and their pros and cons are discussed. New researchers as well as those who are active in this field can benefit from such a review and choose tools for their research in an educated fashion. These aspects are the most important contributions of this paper to the body of knowledge.

The rest of the paper is organized as follows: Section 2 presents a breakdown of IEC 61850 test procedure such as model development and performance studies. Section 3 gives a critical review of IEC 61850 modeling tools along with their advantages and disadvantages. Section 4 presents network emulators that are vital for such kind of research work. Section 5 presents a summarized comparison of all the tools discussed earlier and gives a final verdict. Section 6 concludes the paper.

2. Overview OF IEC 61850 Test Components

The research work in the IEC 61850 field has three distinct steps which require distinct testing procedures for validation. Before going into details of the available software, it is important to understand what these steps consist of and their place within the larger picture. Firstly, the ultimate goal of researchers operating in the IEC 61850 field is to achieve complete connectivity and interoperable communication between different components present in modern smartgrid. As shown in Figure 1, such a connection needs to be established between devices that are very different: power system equipment such as generators, electronics devices such as smart meters or mobile devices such as electric vehicles. In an ideal world, these devices should be able to talk to each other through a common link, using a single language such as IEC 61850. This concept is also called achieving the Internet of Things (IoT) in power systems [37].

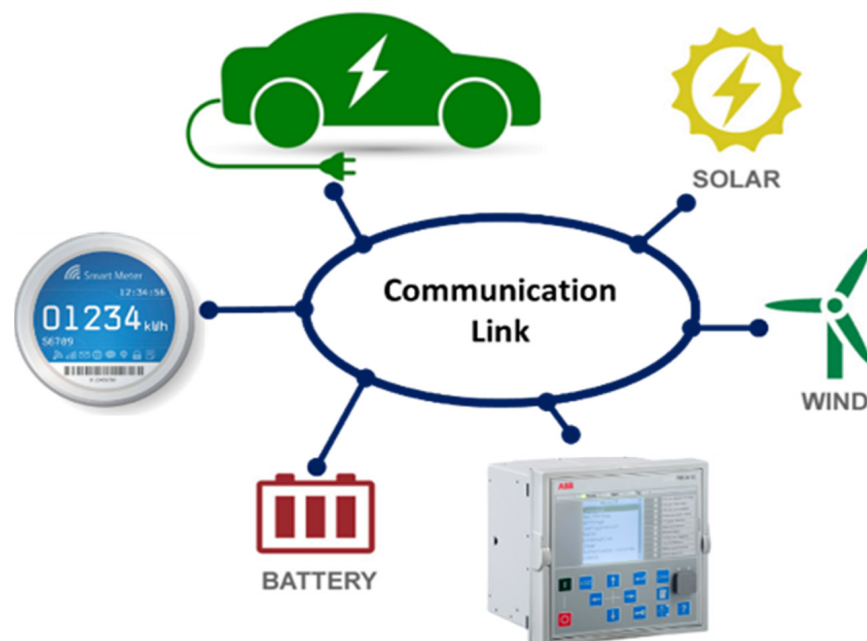


Figure 1. IoT in Smart Grids with different components.

Additionally, again in an ideal world, this should be independent of any particular details of the said device such as their make, manufacturer, model or year. Any EV, be it Tesla, Nissan or Toyota, should be able to connect and exchange information with other devices. Similarly, any PV equipment, smart meter, or power system protection device should *speak* the lingua franca without any issues. This concept is called interoperability [38].

The very first step in IEC 61850 research is developing a communication model for the device(s) that will be part of the experiments. As shown in Figure 2, such a model is used to represent these devices in the communication world. This is achieved by studying the capabilities of the said device and mapping them from the electrical domain to the communication domain within the rules and limitations of the IEC 61850 standard. Some of the devices have ready-made models such as smart inverters [39], while others may need novel models developed for the first time [40–43].

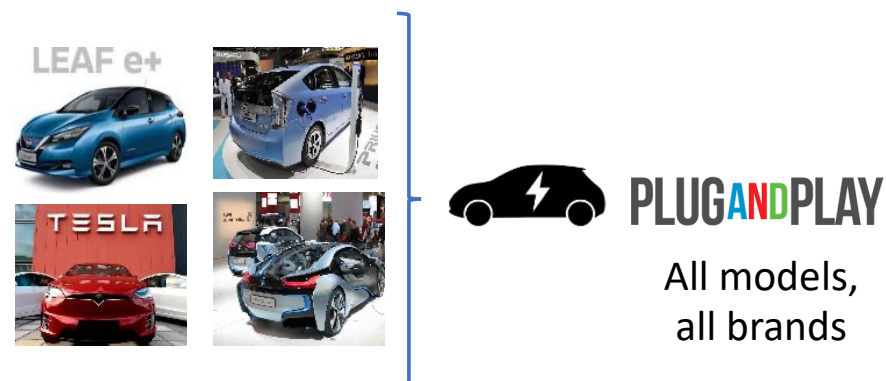


Figure 2. Communication Model development for EVs.

These models are representations of the devices in the communication world where only *information* is exchanged, not power. Therefore, a simple system such as a computer or an embedded system running these models can represent these devices in the communication world. This constitutes the first tool required: an *IEC 61850 model editor*. Due to the complex structure of IEC 61850 models, envisioned message exchanges can also be modeled here.

The next step requires an *emulator* to run these IEC 61850 models, i.e., ICD files, so that the emulated node can represent the intended device and exchanged messages. Full and accurate representation of the model with necessary variables is necessary. In addition, it is important that reporting and messaging preferences embedded in the model operate as required. It is possible to follow successful message exchanges by verifying their impacts on the target nodes, e.g., the status of a particular variable after a message changing it has been issued. As shown in Figure 3, it is also possible to use *network analyzers* that can sniff the packets and capture real messages on the wire, instead of simulated or internal ones only.

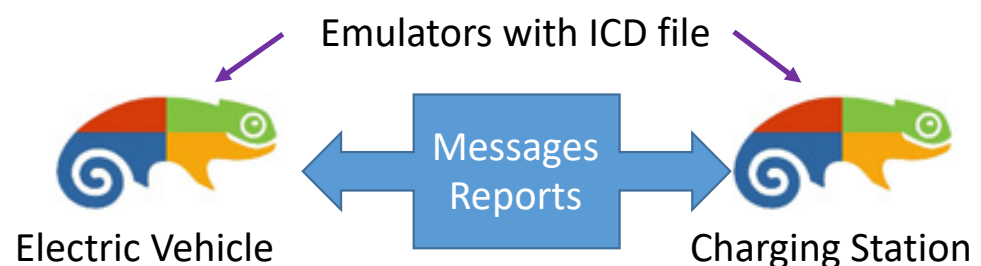


Figure 3. Use of Emulators to run IEC 61850 Models.

The third and final step is about timing considerations. When these messages are exchanged between two emulators that are located side-by-side, modeling and message

exchanges can be verified, but not the timing performance. This is because the messages do not travel over realistic distances as they should in industrial applications. By design, timing is a very important part of IEC 61850 message exchanges and needs to be investigated. While it is not possible to use cables that are several meters or kilometers long in the lab to replicate realistic situations, it is possible to insert *network emulators* in between. As shown in Figure 4, in this fashion, messages travel over long distances through networks with traffic. This ensures that timing performance of IEC 61850 messaging can be examined.

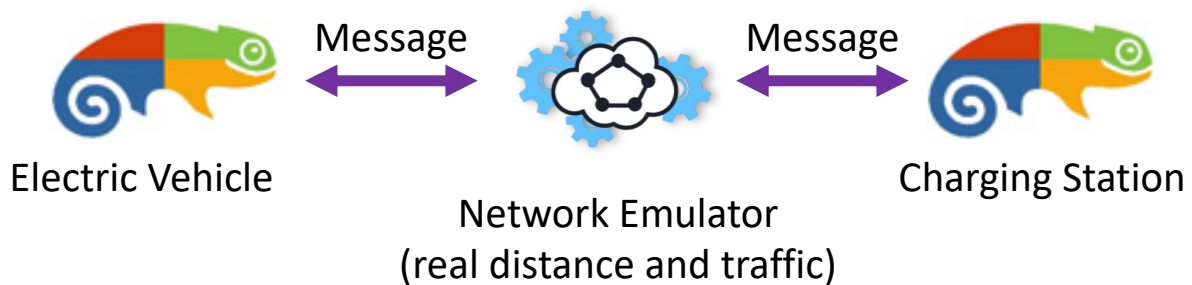


Figure 4. Realistic network and traffic design with Network Emulator.

The software tools for these three distinct steps are reviewed in the next sections. Technically, network sniffers can be classified as another category, but WireShark [44] is so dominant and works so well that it is the only tool used in the trade. It is also freely available. There simply is no reason to try any other tool. It is efficient, effective and comes at no-cost.

3. IEC 61850 Modeling and Emulator Tools

In this section, tools that can develop and run IEC 61850 device models are critically analyzed. Several factors have been taken into account such as capabilities, cost, time required for setup and customer service. The last two factors are especially important for free and paid tools, respectively.

3.1. Infotech

Infotech is a small company that provides several tools for IEC 61850 testing [45]. The software can only be run in Windows which is the biggest limitation of these tools. There are six different components that come with this package, as shown in Table 1. What is most striking about Infotech tools is their *simplicity, ease of use and cost-effectiveness*. These will be explained below.

Table 1. InfoTech IEC 61850 Software Package.

Name	Description
ICD Editor	Create and modify ICD files
61850 Avenue	Emulates Client Devices in IEC 61850 Network
SCL Runner	Emulates Server Devices in IEC 61850 Network
GOOSE Sender/Receiver	Simple interface to send/receive GOOSE messages with different variables and configuration settings
SV Sender/Receiver	Simple interface to send/receive SV (9-2LE) messages with different variables and configuration settings

ICD Editor is a strong editor tool that can be used to generate or modify ICD files. It has a simple user interface as shown in Figure 5. It gives a tree view of any device. Logical

Devices (LDs) and Logical Nodes (LNs) can be added as shown in Figures 6 and 7. It is also possible to add Control Blocks (CBs) and Data Sets (DSs) as shown in Figure 8.

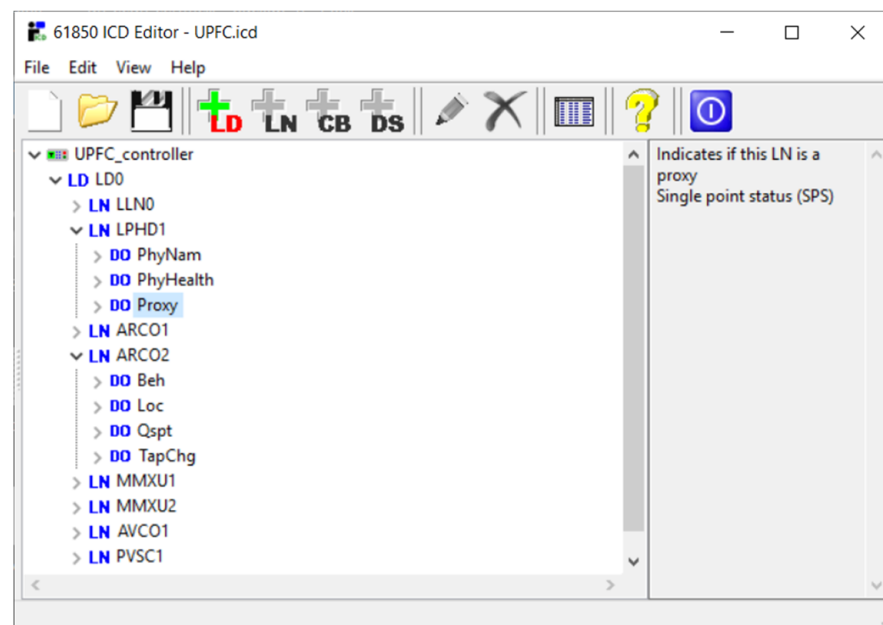


Figure 5. General view of ICD Editor.

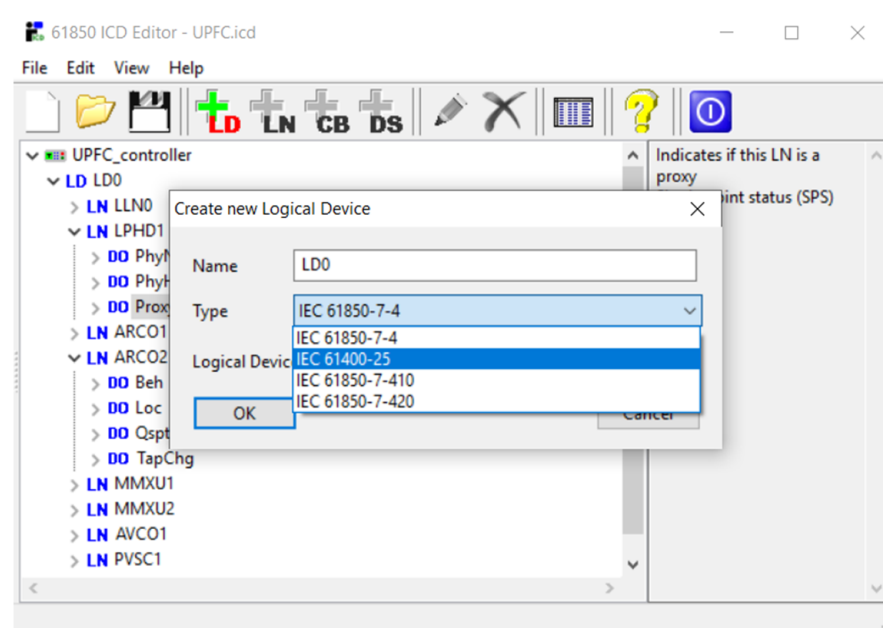


Figure 6. Adding Logical Device from IEC libraries.

The only limitation of this tool is that LDs can only be added from published IEC 61850 libraries so that they will be compliant with the standard. When developing novel IEC 61850 models for devices that are not yet included in the standard [46–48], this becomes a barrier. One possible way to work around this is by using the XML editor capability of the ICD editor tool. This shows the XML rendition of the ICD file created as shown in Figure 9. Following the XML rules, it is possible to add novel LDs and LNs into an IEC 61850 model. This approach provides all the necessary capabilities for ICD editing in the IEC 61850 research field.

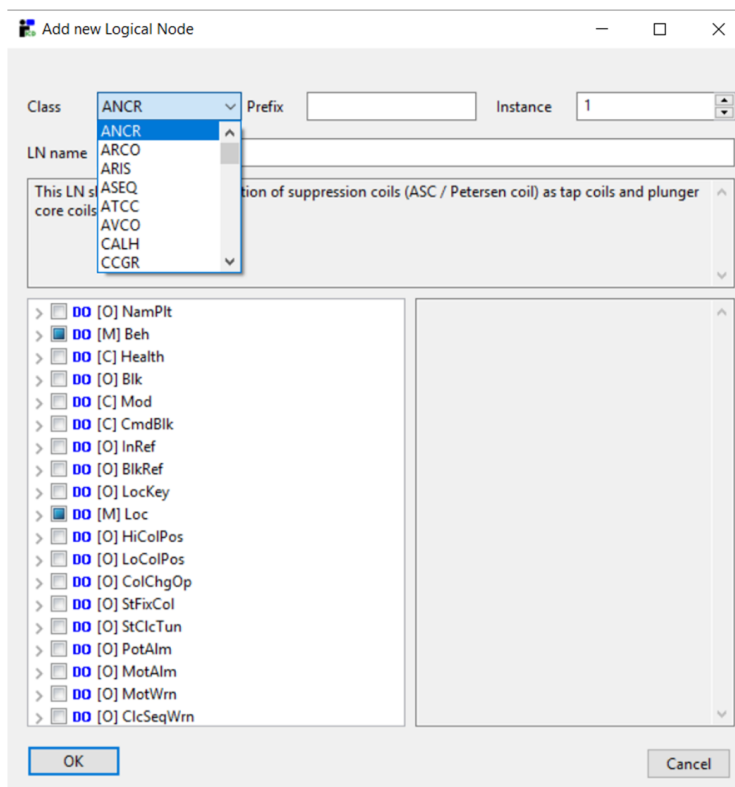


Figure 7. Adding Logical Node from IEC libraries.

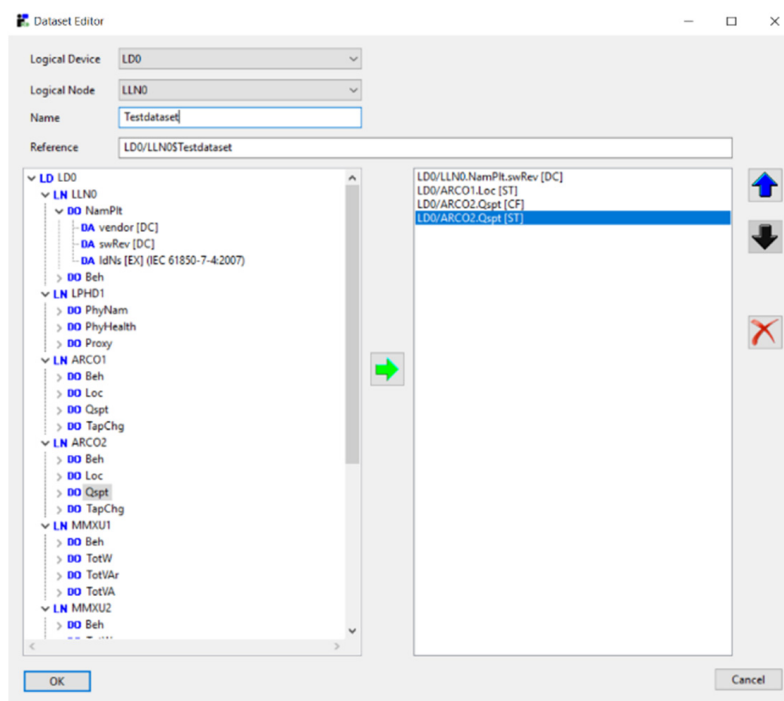


Figure 8. Creating a DS within IDS based on LDs and LNs.

Once necessary ICD files are prepared, 61850 Avenue and SCL runner tools are utilized to emulate client and server devices, respectively. Proper connection can be established, local data changes can be reported to the remote terminals and necessary instructions can be sent. All the actions can be observed in real-time and messages can be captured in the

network. Figure 10 shows how one or more clients are emulated in 61850 Avenue. One device (laptop or pc) can emulate several clients but only a single server.

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <SCL xmlns="http://www.iec.ch/61850/2003/SCL" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" version="2007" revision="
3 <Header id="" revision="1" version="0" toolID="INFO TECH ICD Editor" nameStructure="IEDName">
4 </Header>
5 <Communication>
6 <SubNetwork name="S1">
7 <ConnectedAP iedName="UPFC_controller" apName="P1">
8 <Address>
9 <P type="IP">127.0.0.1</P>
10 <P type="IP-SUBNET">255.255.255.0</P>
11 <P type="OSI-AP-Title">1,3,9999,23</P>
12 <P type="OSI-AE-Qualifier">23</P>
13 <P type="OSI-PSEL">00000001</P>
14 <P type="OSI-SSEL">0001</P>
15 <P type="OSI-TSEL">0001</P>
16 </Address>
17 </ConnectedAP>
18 </SubNetwork>
19 </Communication>
20 <IED name="UPFC_controller" configVersion="1.0" type="" manufacturer="" desc="">
21 <Services>
22 <DynAssociation/>
23 <SettingGroups/>
24 <GetDirectory/>
25 <GetDataObjectDefinition/>
26 <DataObjectDirectory/>
27 <GetDataSetValue/>
28 <DataSetDirectory/>
29 <ConfDataSet max="0"/>

```

Figure 9. XML editor available in ICD Editor tool.

The screenshot displays the Avenue interface with the following sections:

- Servers:** A list of servers including "EV127.0.0.11", "New IEC-61850 server [127.0.0.11]", and "Demo [127.0.0.11]".
- Main:** Configuration details for the selected server, including Name, Address, Port, and various OS parameters.
- Substation Configuration:** Details for the IED name, RCS Indication, and SCL File Name.
- Association Table:** A table with columns for Name, FC, and Value, listing various data objects and their associated functions.
- Log:** A message log showing connection events and data set requests.

Figure 10. Avenue Interface with Server Information.

It is possible to add multiple servers on the left to which Avenue should connect as a client. With the recent update, this connection can use Transport Layer Security (TLS) with certificate authentication, and a private key. Once the connection is established, server device data model can be explored, and the current data values can be read. If there is a change in the parameter values this is denoted with blue ink.

As shown in Figure 11, it is possible to create new data sets from the server's data objects. It is also possible to set up reporting conditions based on data sets defined in the server's ICD file or created in Avenue. It goes without saying that it is possible to send control commands as real or test messages as shown in Figure 12.

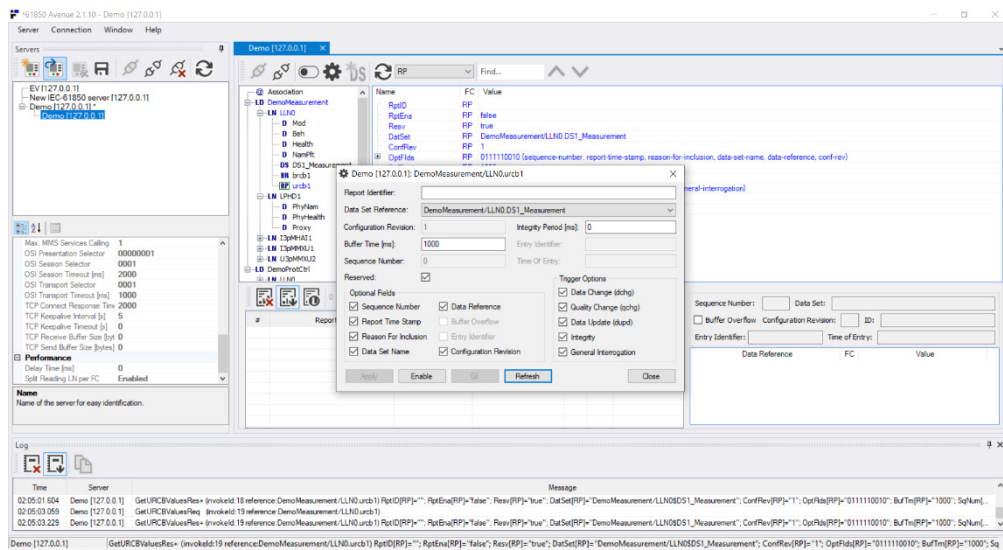


Figure 11. Reporting in Avenue.

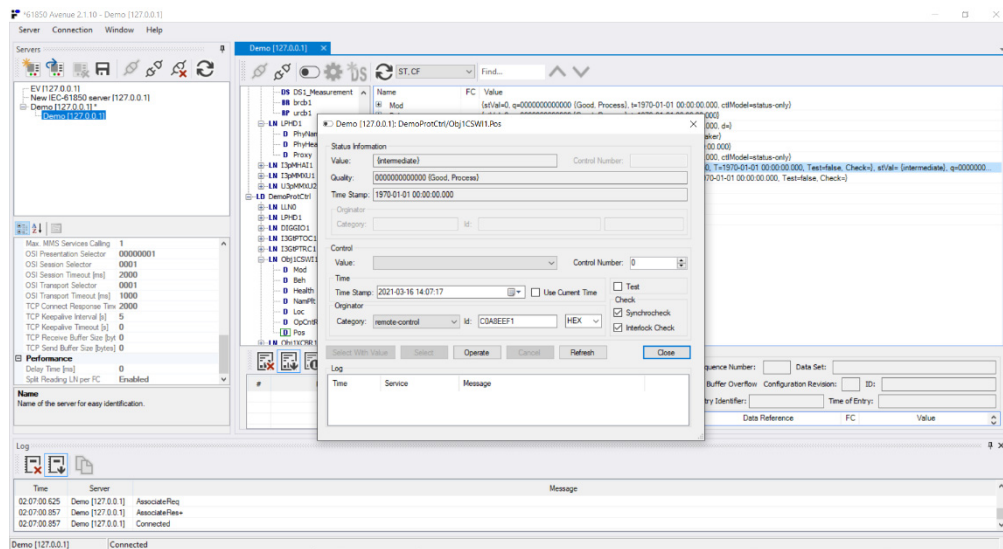


Figure 12. Control Commands sent as real or test messages.

On the server side, SCL Runner is running the ICD file of the server and emulating it. As shown in Figure 13, several servers can be added with their ICD files and IP configurations. At any given time, only one can be launched, and the computer emulates that particular device.

Avenue and the SCL runner can be run on different platforms or as a loopback setup. In either case, when the client and server are connected, the parameters can be edited, and changes can be observed on the opposite side. If the server makes the changes locally, client receives updated information in blue ink, as shown in Figure 11. On the other hand, client can ask for a parameter to be changed in the server, which is reflected inside the server if this modification is allowed.

GOOSE and SV messages can be triggered from Avenue and SCL runner tools as long as the message configurations are done within ICD files, such as goose control block GOCB. However, sometimes researchers would like to focus on the message design and their contents, more than on full-fledged ICD file development. For such cases, there are simple sender and receiver tools for both GOOSE and SV messages. As shown in Figure 14, GOOSE sender has a simple interface where network parameters of the message can be easily configured. Furthermore, GOOSE block data set can be created by adding parameters

from a drop-down menu. When the transmission starts, the tool starts sending GOOSE messages through the chosen network adapter. Initially, only simple GOOSE messages could be sent. However, with the recent update, routable-GOOSE, R-GOOSE, can also be sent with tunneling options.

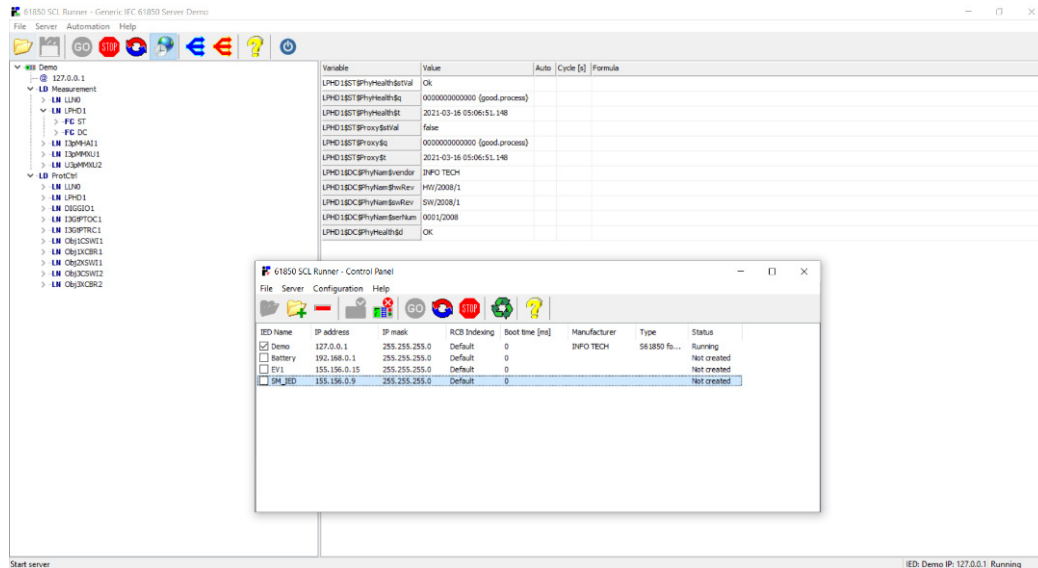


Figure 13. SCL runner emulating a server.

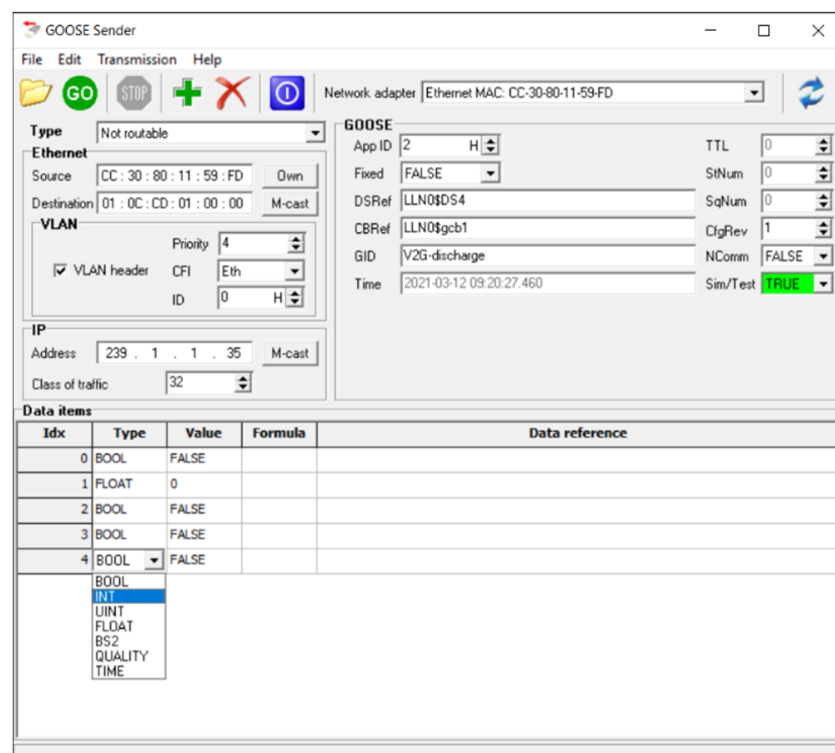


Figure 14. GOOSE Sender Interface.

The receiver tool’s interface, as shown in Figure 15, is almost a mirror image of the sender. The only difference is that everything is read and shown, not editable. The receiver can detect available streams in the network, depending on the selected adapter. Once subscribed, the contents of the GOOSE message are shown in a user-friendly manner. As

shown in Figure 16, the tool also has a parser window for viewing more technical details of the message.

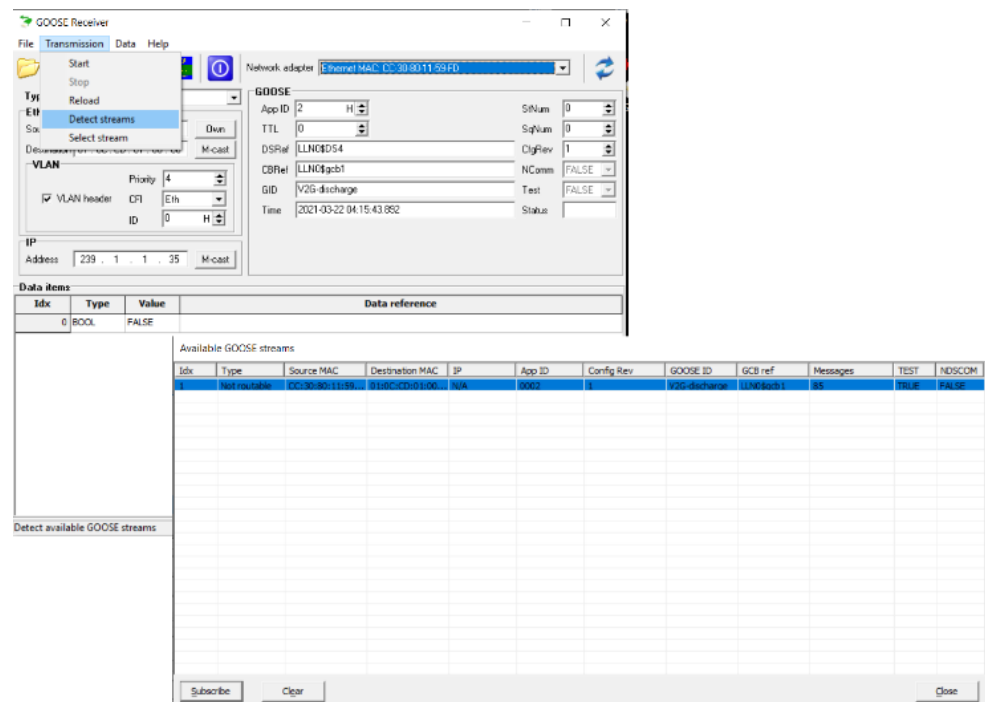


Figure 15. GOOSE Receiver Tool and Stream Detection.

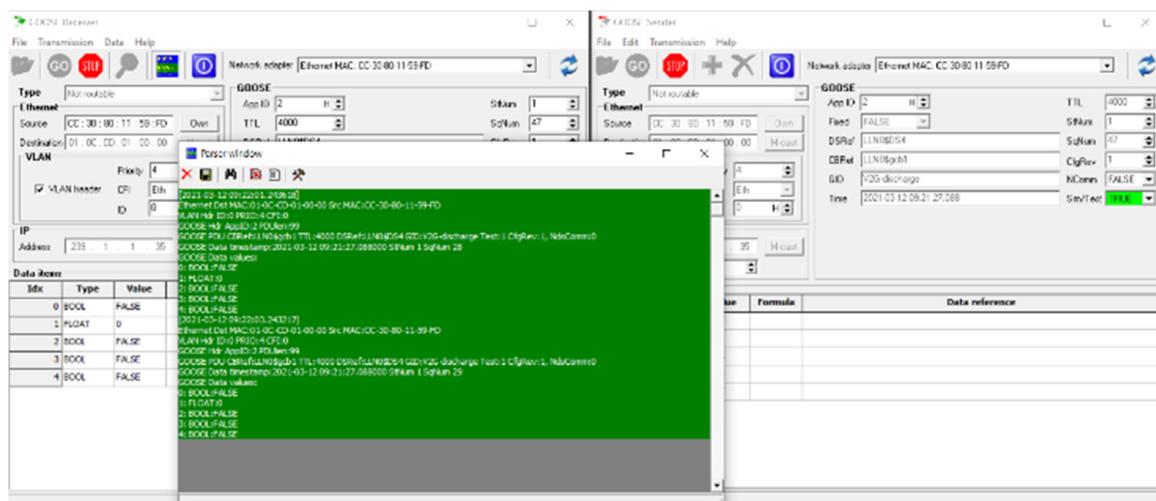


Figure 16. Message Parsing in GOOSE Receiver Tool.

These sender and receiver tools need not be used as a pair. They are interoperable with any valid GOOSE message. Therefore, they offer a very convenient method of testing when the main focus is sending and receipt of messages.

SV Sender and receiver tools, (see Figures 17 and 18), operate in almost the same way, with a significant difference. Industry has agreed on a very limited implementation of SV messages that is called 9-2LE, i.e., limited edition [49]. This is due to the fact that, in their general definition, SV messages are very general, and industrial partners found it hard to implement. Following this trend, SV sender and receiver tools can only deal with 9-2LE messages. For industrial purposes, this ensures capability. However, as in ICD editor tool, for cutting-edge research it creates a handicap. Novel SV message designs and their

operations cannot be implemented or tested, e.g., use of SV for much slower messages such as EV charging signal [50].

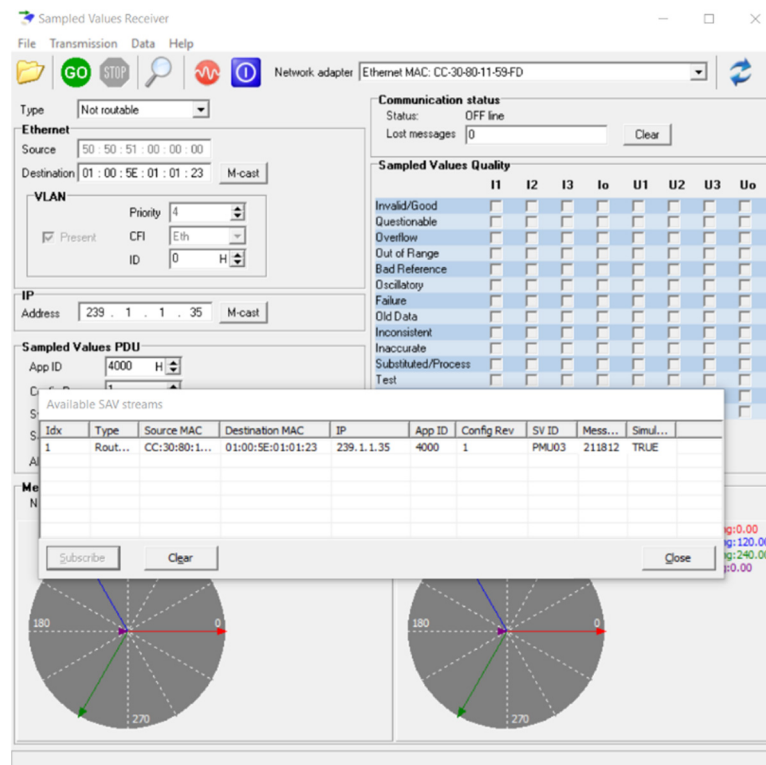


Figure 17. SV Sender Interface.

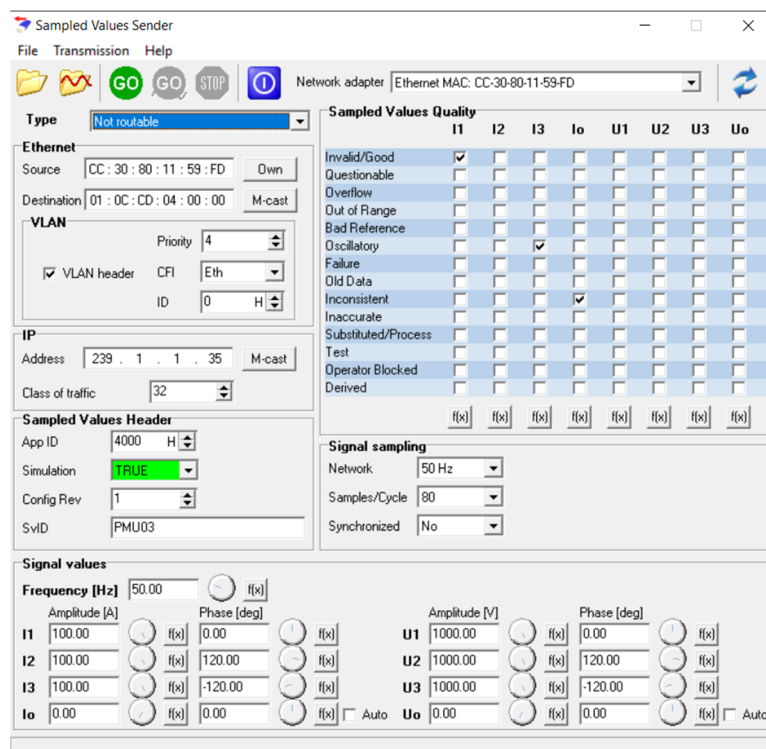


Figure 18. SV Receiver Detecting 9-2LE streams.

In addition to technical properties, this work evaluates these tools in terms of cost, customer service, experience with the company, etc. The overview of Infotech tools evalua-

tion is given in Table 2. From this perspective, it can be easily said that Infotech passed all these fields with flying colors. Firstly, the cost of the entire suite is very reasonable. The company is very responsive and communicates well. They acknowledge the receipt of payment, and then follow up with product delivery, licensing and shipping of physical keys as shown in Figure 19. The only negative aspect of Infotech products is that these keys are the purchased value. In other words, if these USB flash keys are lost or became unresponsive, the customer would bear the cost of another purchase. That being said, dongles have an advantage, where the tools can be installed in a number of computers and keys can be carried with ease, instead of carrying the computers around. This definitely has advantages along with drawbacks.

Table 2. Infotech Tools Review Summary.

Evaluation Criterion	Score	Notes
Tool Capabilities	5—Excellent	SV sender and receiver tools only support (9-2LE) ICD Editor only supports standardized LNs, LDs. XML editor should be used for novel LN/LD designs
Cost	5—Great Value for Money	Infotech costs 10 % of its competitors but does the same job in a reliable and better way
Customer Service	4—Responsive and Active	Online Training is available immediately after purchase, Updates are sent by the company in a proactive way
Licensing	3—Convenient	USB flash keys are used for licensing. Malfunctioning or loss is always a possibility. It also makes the use of several platforms easier.
Time required for Installation and Use	5—Very Robust and User friendly	Installation packages are reliable. Installation takes less than 15 min. Any person with basic IEC 61850 knowledge can master the tools in a day. Online training is also included with the purchase.
Installation is very smooth.	5—Excellent	The tools are robust and reliable. Even after long periods of idle time, they work as expected with no issues. Interoperability tests with other tools also proved their successful operation.

In terms of customer service, Infotech is very responsive and helpful. The company set up online training sessions within one week of purchase. The trainers extensively discussed the capabilities and use of these tools. Most importantly, the use of the tools is intuitive, and they are reliable. The menus, icons and menus are very easy to master and remember. Once this is done, it is convenient to use these tools daily, or only once in a while. Sometimes, research projects go cold, and researchers want to revive them. Infotech tools proved to be easy to remember and reliable to work even after months of non-use. Despite being taken for granted, these qualities are not always present, as will be seen in the next tool.

3.2. Xelas Energy

Xelas Energy software is a scavenged tool from old communication emulation programs. Bootstrap solutions are developed to emulate IEC 61850 servers and exchange IEC 61850 messages. It is also claimed that IEC 62351 cybersecurity requirements are built into this tool. However, it is not possible to over emphasize that Xelas Energy is reuse of an obsolete software. It is not a freshly developed tool and this manifests itself in all aspects such as poor performance, little to no reliability and very low user-friendly interface.

Surprisingly, these are not the worst aspect of Xelas Energy, but customer experience and interaction with the company. Firstly, technical capabilities will be presented.



Figure 19. Infotech Flash Keys.

This tool is provided as a collection of different packages in one installation file. Contents are not clearly listed and there are many dependencies. As shown in Figure 20, the tool is not self-contained but is a collection of different things. The GUI is provided by an Internet Browser, as the tool does not have its own GUI. Energy Management System is not relevant to anything, although it is always mentioned in documentation along with the tool itself (see Figure 21). Its operation depends on the Glassfish [51] server which, then, communicates with IEC 61850 stack to exchange messages. This requires *tp0d* to be utilized. Due to bad design of Xelas Software tool, *tp0d* is not only occupied when the tool runs, but also when it is closed. This means that other software that require access to *tp0d* cannot be run on the same computer. Similarly, glassfish is run in each boot cycle, takes up memory and slows down the computer.

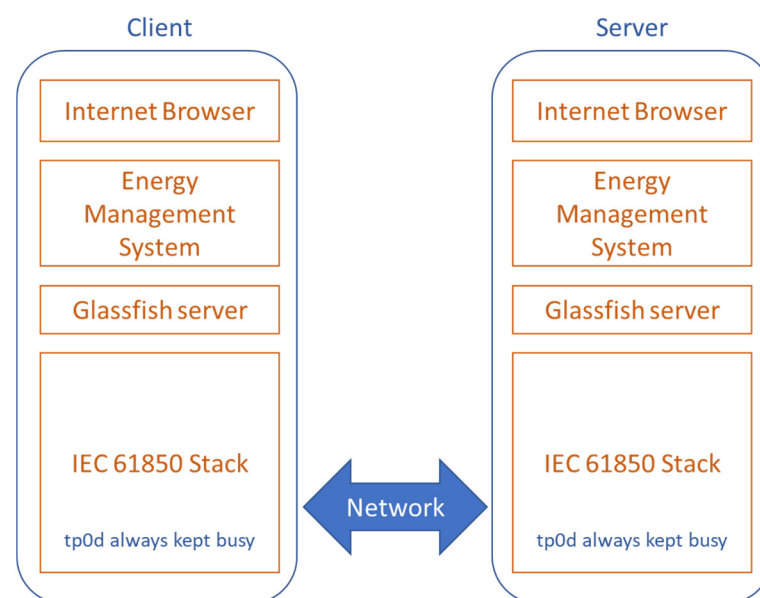


Figure 20. Xelas Energy Architecture.

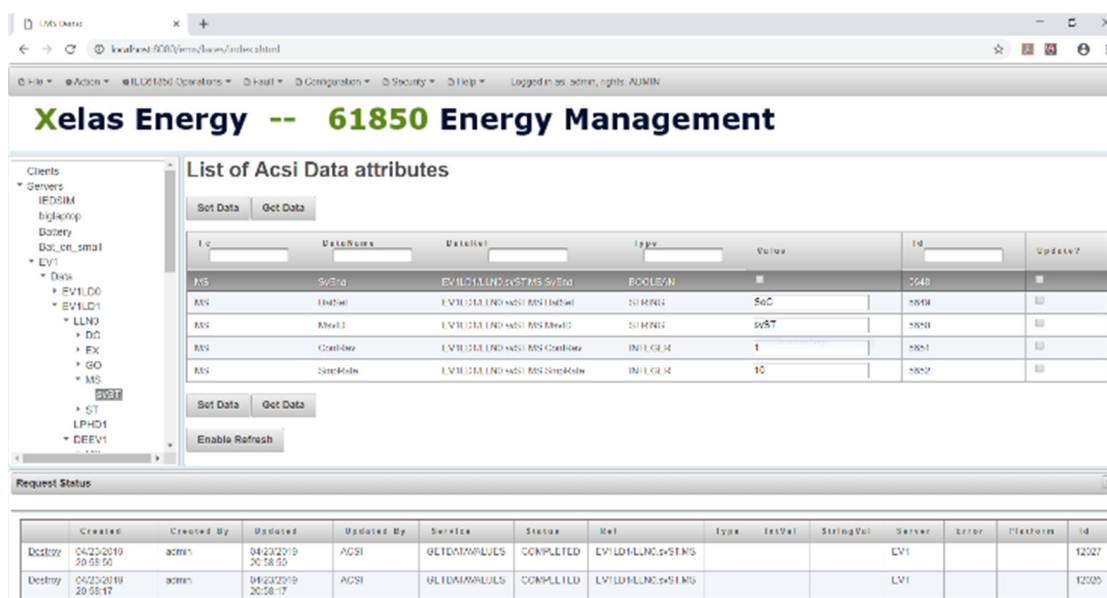


Figure 21. Xelas Energy GUI.

Xelas Energy can run on both Linux and Windows platforms. However, to run on Windows, Cygwin64 [52] is needed as this tool is based on other items originally developed for Unix-based systems. This also increases the installation footprint and impact on the performance of the host computer. Experience shows that it is not possible for Xelas Energy to share the same computer/laptop with other software packages in an efficient or effective way. This means two computers need to be dedicated only for this tool. This is very expensive and inefficient. These are some of the many poor aspects of this tool which stem from the fact that it is salvaged from obsolete software and has a bad design.

The GUI is a webpage shown on an Internet Browser, as shown in Figure 21. Xelas Energy's design necessitates that everything is done through this browser interface. This proves to be very limiting in practice. In contrast with Infotech tools which can be used separately, Xelas cannot be used to perform different tasks in parallel. All data changes, MMS requests, GOOSE and SV messages are sent through a single interface. Following up these tasks, confirming successful transmission and parameter update are very troublesome.

In addition to this design fault, Xelas has much more serious technical problems. Firstly, it is almost impossible to fully grasp the operation principle of this tool. This means simple steps such as creating an emulator can be troublesome. There are two ways of creating an emulator. One way is by adding an IED first and then importing its properties through Directory update. The second one is creating it by importing its ICD file. Both options should work equally well. However, experience has shown that the former creates a lot of problems during message exchanges. The company cannot provide a satisfactory answer as to why this should be the case or why both options are provided if only one of them works.

Another major problem with Xelas software is reliability. Following the same procedures does not always yield the same results. Emulators are known to become unstable or unresponsive with time. A working set up can become useless later. This is very critical for researchers where the same level of consistency is required to reproduce research results or pick-up from other projects and continue new investigations. In some cases, this time window can be several months, and Xelas Energy has proven to be unreliable. This means researchers need to reinvent the wheel every time which causes frustration and waste of time.

The licenses are fixed to MAC addresses of the host computers. In contrast with the flash key dongle approach taken by Infotech, this seems more convenient at first. However, once a license is fixed, it cannot be transferred in any way. Should there be need to dispose

of the host computer or upgrade to a new terminal, there is no way of transferring the license. Xelas requires an annual service package to be purchased, if such a request is made at any time. This is very limiting for use and an extra cost for research labs that tend to change and evolve over time.

In addition to these terrible technical accolades, Xelas Energy has many other issues. Firstly, it does not provide the simplest of tools required for IEC 61850 research. However, the company is not honest about the contents of a purchase. For instance, even a simple ICD editor is not included in the tool. If this is mentioned in a transparent way, that may not be such a big problem. However, during our preliminary discussions, it is always stated that the tool will include an ICD file editor. This fact is also mentioned in the technical specifications that both parties agreed on for a determined price.

On the last day of bid submission, as shown in Figure 22, Xelas's CEO Koert Bloom sent a different quote with a much higher price. When we inquired about the inconsistency in price, Mr. Bloom claimed that it was his mistake and sent another one with the originally determined price. The catch is that the second quote did NOT include the ICD editor, although it has always been part of the negotiations and it is clearly written in the specifications document. This old quote switching trick is not commonly used in research circles and researchers are oblivious to such tactics. However, with Xelas Energy, researchers do not know what they are purchasing, even until the last minute. Our experience should be a warning for all potential customers. It is also important to note here that, despite its terrible performance, Xelas Energy is much more expensive than its competitors, costing roughly 10 times more than Infotech covered above.

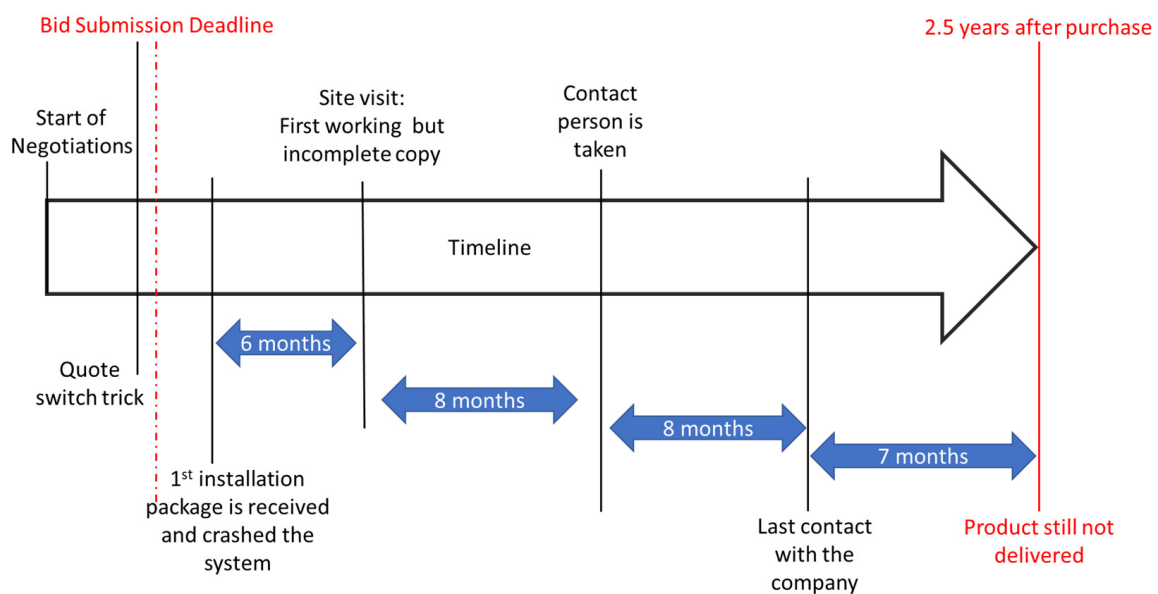


Figure 22. Customer Service Timeline.

Another feature that is clearly missing and has not been provided after almost 3 years is 9-2LE SV messaging capability. Xelas Energy cannot send 9-2LE SV messages in Windows platform. When this was brought up during the site visit, the Xelas representative could only manage to send 1 message per second, which is much less than the stipulated 256 and 80 samples per second in 9-2LE. What is more striking is that, although these sample counts are very well-known and clear, they argued that what they are sending is, in fact, 9-2LE. This unreasonable and aggressive behavior is the modus operandi for Xelas Energy and researchers should be very cautious of this.

A summary of customer service issues is given in Table 3 while Table 4 summarizes overall experience with Xelas Energy software from technical, economic and reliability aspects.

Table 3. Customer Service Issues with XELAS Energy.

Problem	Example
Lack of Communication	<p>There is no clear communication about anything:</p> <ul style="list-style-type: none"> - Delivery Date - Next Update Release Date - Which issues are being fixed - What can the client expect in the next release, next month or next year
Unproductive Communication	<ul style="list-style-type: none"> - Emails are mostly ignored. On average, 1 reminder needs to be sent to receive a response on any topic. For serious topics, 2 or 3 reminders is typical. - Intentionally ignoring email contents, responding to a portion of requests. On average, 30%–40 % of correspondence is taken into consideration. Clients have to repeat the remaining 60%–70% in separate emails. - Inability to comply with simple requests such as maintaining a certain list of recipients in emails. This is also intentionally done to leave clients out of the loop.
Ghosting	<ul style="list-style-type: none"> - Hard to believe for a professional company. All personnel, including the CEO, practice this.
Belligerent and aggressive behavior	<ul style="list-style-type: none"> - Being super defensive whenever a bug or problem is brought up (there are many) - Accusing the client. instead of investigating the issue. - Common responses include: “It works in my system.” “You must have made a mistake.” “The error is caused by 3rd party software we included. That’s not our fault.” “This feature may be promised in the contract. But the dependencies we use do not support it.”
Dishonesty	<p>Clients have to be on guard constantly. Discussions, quotes, new releases or any topic, in general, is subject to manipulation.</p>

Table 4. Xelas Energy Review Summary.

Evaluation Criterion	Score	Notes
Tool Capabilities	−3—Insufficient and Incomplete	<p>Company claims it can emulate several clients simultaneously, perform all IEC 61850 actions with IEC 62351 security</p> <p>In reality, the software is very sloppy, unreliable, full of bugs, cannot perform IEC 61850 tasks, e.g., sending 9-2LE messages. ICD editor is not provided.</p>
Cost	−5—Too expensive	<p>More expensive than its competitors, roughly 10x more expensive; Company is not honest about the final price, it is not delivered fully</p>
Customer Service	−5—Belligerent and Accusatory	<p>The biggest reason to avoid Xelas Energy. Company has serious attitude issues. They are unresponsive, disrespectful and belligerent. Any complaint or request is met with aggression or indifference.</p>
Licensing	−2—Bad	<p>The licenses are tied to MAC addresses of the laptops. In case of laptop replacement, Xelas asks for extra payment for license transfer.</p>
Time required for Installation and Use	−5—Terrible	<p>Installation packages cause system crashes. Installation is not sufficient; a lot of customization is required for bare minimum operation. It is difficult for user to remember operation.</p> <p>The product is not fully delivered even after 2.5 years.</p>
reliability	−5—Terrible	<p>It never works as expected. Working setups become unresponsive and unstable after some time. Since it has many dependencies, any slight change in the system creates serious issues.</p>

This belligerent and unreasonable behavior has been the hallmark of Xelas Energy's customer service. As shown in Figure 22, after the payment was made, it took six months for a working copy to be delivered, albeit incomplete and full of bugs. The first installer sent was not working properly and crashed our testing terminal. Linux installer was such a poorly written script, that it did not even add repositories or update versions. After almost 3 years, the product is still not fully delivered. In spite of this, not once did Xelas personnel or Koert Bloom, CEO, offer apologies or tried to make amends. The default mode of operation is unfazed and unapologetic belligerence. During the first 6 months, no working copy was delivered. All installation packages were faulty and at times caused crashes in our host computer. Again, in all these cases, Xelas Energy's customer service either blamed the user, the dependencies which they included, or any other item. Owning up or taking responsibility has never been experienced with this company.

3.3. libIEC61850

In addition to the paid tools listed above, there are freeware available. These are developed by researchers in the field and they are used not only by researchers but also by some of the companies listed above. libIEC61850 [20] is one of the earliest in this area. It is a robust tool with extensive documentation [53] which puts some of the paid tools to shame.

libIEC61850 is the implementation of libraries required to run IEC 61850 clients and servers. It is not only free but also open source, which is why it is a building block for almost all paid tools. C language is utilized to allow portability. It can be run on Windows, Linux or embedded systems. It has a Java-implemented version, although this requires more resources to run [54].

The general library structures for server and client are given in Figures 23 and 24, respectively. As shown, the respective server or client provided sits on top of the relevant Application Programming Interfaces (APIs) and stacks. All of these can be mapped onto different implementations such as Linux, Windows, Mac OS or user provided. Different portions of the library are utilized to develop different tools such as stand-alone GOOSE publisher/subscriber code, stand-alone SV publisher/subscriber code or conversion tool that can convert SCL files to IED models [55]. Once the server and client models are deployed, a services layer is used to run relevant services such as *association*, *read/write*, *reporting services*, etc.

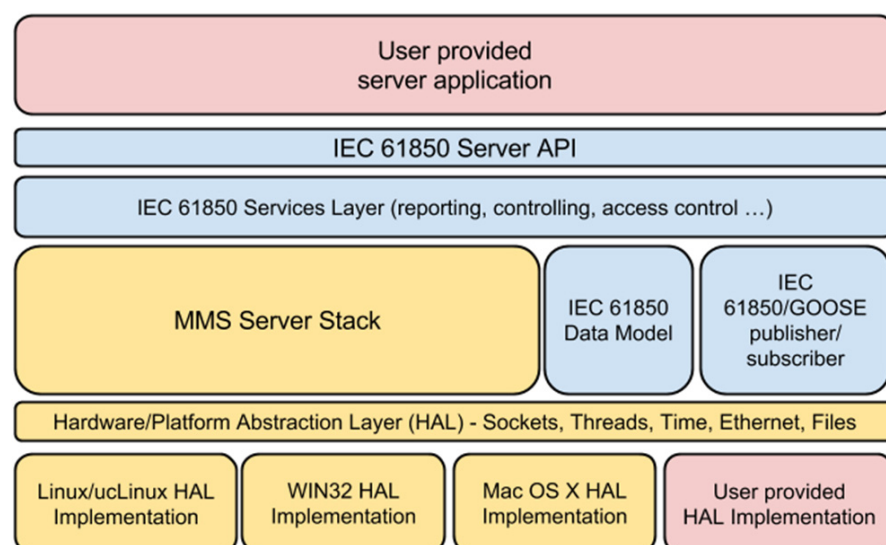


Figure 23. IEC61850 Server library.

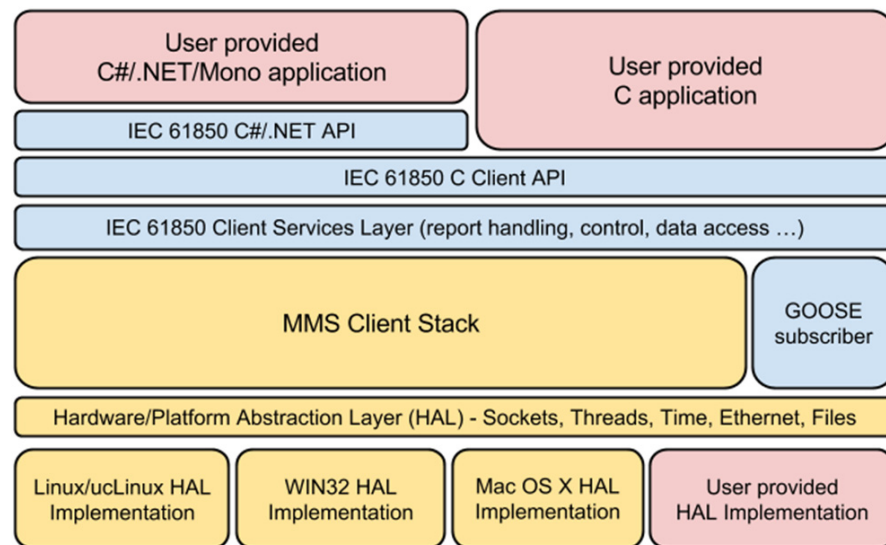


Figure 24. IEC61850 Client library in libIEC61850.

Despite being very strong and providing a complete implementation of IEC 61850 library, libIEC61850 has a serious drawback. It is not a testing tool, per se. It is a building block or stack development for further use by developers. Therefore, it has a very steep learning curve and is not user friendly, by any means. Everything is code based, either in C or Java. For instance, the coding in Figure 25 is required for the IEC 61850 server model.

```

1 | IedServer iedServer = IedServer_create(&iedModel);
2 |
3 | IedServer_start(iedServer);
4 |
5 | while (running) {
6 |     Thread_sleep(1);
7 | }
8 |
9 | IedServer_stop(iedServer);
10 |
11 | IedServer_destroy(iedServer);

```

Figure 25. IEC61850 Server Creation [55].

Here, *iedServer* is manually created and started in lines 1 and 3, while the loop corresponds to the time when the server is alive and operational. *Stop* and *destroy* functions clean the memory allocated to the server once it is not used anymore. It is also possible to do this step by importing an ICD file into *create* function. Nevertheless, there is no GUI or any interface to make these steps clearer and easier. Similarly, feeding values that are needed in regular operation requires defining these parameters as shown in Figure 26. If there is a GUI, this would be done by changing the parameters and a relevant MMS message would be sent. libIEC61850 needs to be used with a high-level of programming knowledge. It needs to be run through the command line. For instance, the following command should be run to create a server from an ICD file: `java -jar genconfig.jar Example.icd Example.cfg`.

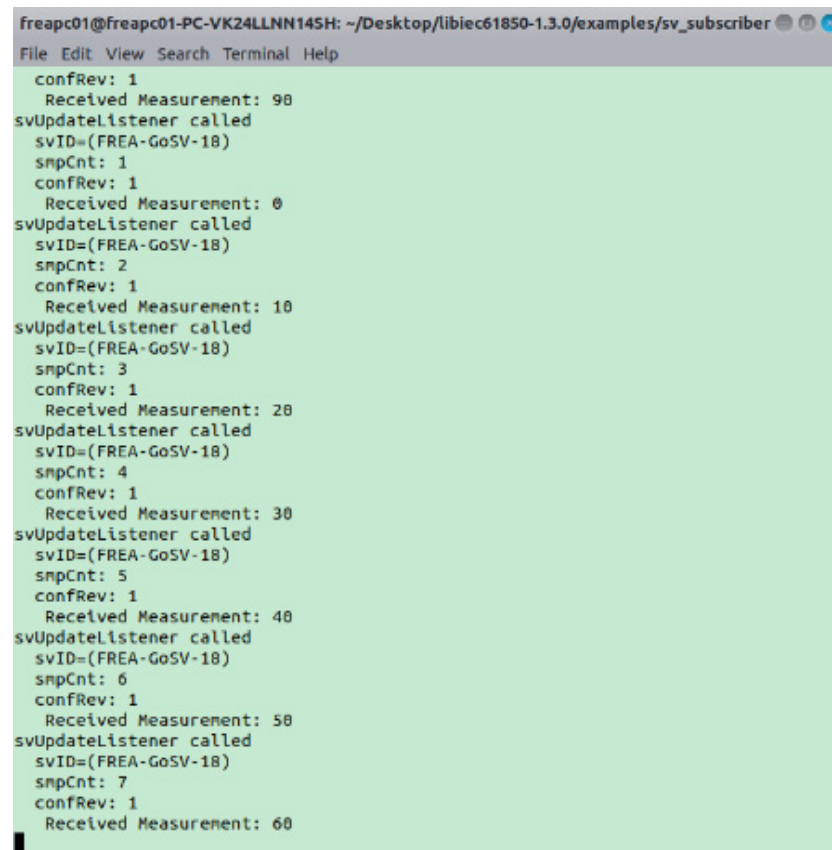
Servers and clients are created with commands and several actions can be performed when they are alive, depending on the code used to create that particular client or server. Figure 27 shows an SV receiver tool implemented in libIEC61850. Data visualization is not very user-friendly, and it is cumbersome to perform control and monitoring with these tools.

```

1 float power = 500.f;
2
3 MmsValue* powerValue = MmsValue_newFloat(power);
4 MmsValue* powerTimestamp = MmsValue_newUtcTime(time(NULL));
5
6
7 while (running) {
8     MmsValue_setUtcTime(powerTimestamp, time(NULL));
9
10    IedServer_lockDataModel(iedServer);
11
12    IedServer_updateAttributeValue(iedServer,
13        IEDMODEL_Inverter_MMXU1_TotW_mag_f, powerValue);
14    IedServer_updateAttributeValue(iedServer,
15        IEDMODEL_Inverter_MMXU1_TotW_t, powerTimestamp);
16
17    IedServer_unlockDataModel(iedServer);
18
19    power += 0.1f;
20
21    MmsValue_setFloat(powerValue, power);
22
23    Thread_sleep(500);
24 }

```

Figure 26. Parameter Update via MMS Messages [55].



```

freapc01@freapc01-PC-VK24LLNN145H: ~/Desktop/libiec61850-1.3.0/examples/sv_subscriber
File Edit View Search Terminal Help
confRev: 1
  Received Measurement: 90
svUpdateListener called
svID=(FREA-GoSV-18)
snpCnt: 1
confRev: 1
  Received Measurement: 0
svUpdateListener called
svID=(FREA-GoSV-18)
snpCnt: 2
confRev: 1
  Received Measurement: 10
svUpdateListener called
svID=(FREA-GoSV-18)
snpCnt: 3
confRev: 1
  Received Measurement: 20
svUpdateListener called
svID=(FREA-GoSV-18)
snpCnt: 4
confRev: 1
  Received Measurement: 30
svUpdateListener called
svID=(FREA-GoSV-18)
snpCnt: 5
confRev: 1
  Received Measurement: 40
svUpdateListener called
svID=(FREA-GoSV-18)
snpCnt: 6
confRev: 1
  Received Measurement: 50
svUpdateListener called
svID=(FREA-GoSV-18)
snpCnt: 7
confRev: 1
  Received Measurement: 60

```

Figure 27. SV receiver tool in libIEC61850.

For this reason, there are two primary uses for libIEC61850. Firstly, developers use it to develop tools on top of it. Researchers, on the other hand, tend to use it as an easy interoperability testing tool for applications they have developed [56,57]. In these cases, not much modification is required since libIEC61850 tools are used to validate the operation of other tools such as successful receipt of a GOOSE or SV message. Table 5 shows evaluation scores and associated notes.

Table 5. libIEC61850 Review Summary.

Evaluation Criterion	Score	Notes
Tool Capabilities	5—Excellent	The library is thorough and constantly updated with IEC 61850 revisions
Cost	N/A	Free
Customer Service	3—Acceptable	Developer is active in their website and responds to questions, but the tool comes with no guarantee or support.
Licensing	N/A	No license is required.
Time required for Installation and Use	1—Challenging	It has a steep learning curve. Especially for larger projects, long-time should be invested. Easy testing is possible with example tools.
Reliability	5—Excellent	The tools are robust and reliable. Even after long periods of idle time, they work as expected with no issues. Interoperability tests with other tools also proved their successful operation.

3.4. *rapidIEC61850*

As the name implies, this tool is supposed to accelerate IEC 61850 modeling and testing procedures. The developer claims [58] that this tool can be easily set up on embedded systems such as raspberry pi. Once this is done, it is possible to import any ICD file and emulate it on the platform that is running RapidIEC61850.

At first, it seems that the instructions are clear and the setting up should be straightforward. However, the authors have spent countless hours trying to set up the project and debugging it. The files on github project site cannot be configured and compiled. Authors have contacted the developer [59], and correspondence showed that there were missing files and steps. Even after following the updated instructions, the tool cannot be successfully configured and executed.

One of the issues is that the step where rapidIEC61850 should recognize projects and import them, i.e., step 5 of the instructions, does not work. No project can be successfully converted or imported. This issue has been reported to the developer, but no solution is provided, and nothing is changed in the github package. Different combinations, such as use of Eclipse on Ubuntu or Windows, have been tried to no avail.

Another important point is that authors have visited the lab at the University of Strathclyde where the developer, Dr. Steven Blair, works as a part of ERiGRID Transcontinental access project [60,61]. The design of the collaboration required use of RapidIEC61850 for hardware-in-the-loop tests for a smart EV charging scheme based on IEC61850 [62]. Overall, this project was active for a year with 2 weeks of active lab tests in UoS. During this time, the developer failed to provide a working copy of RapidIEC61850. This clearly casts doubt on the validity of the claims about this tool as the results cannot be duplicated and developer's claims cannot be verified. Table 6 summarizes the review results for this tool.

It is important to note here that the loss of time and effort caused by RapidIEC61850 and its false claims are some of the main motivations for this review work. It is extremely time-consuming and demotivating to work on such non-operational tools which claim to work. A lot of resources are dedicated in research labs to set up and test the tools when, in reality, they are not operational. It is vital for the researchers to share their experiences to avoid such wastes of resources, both time and money, as well as inefficiencies.

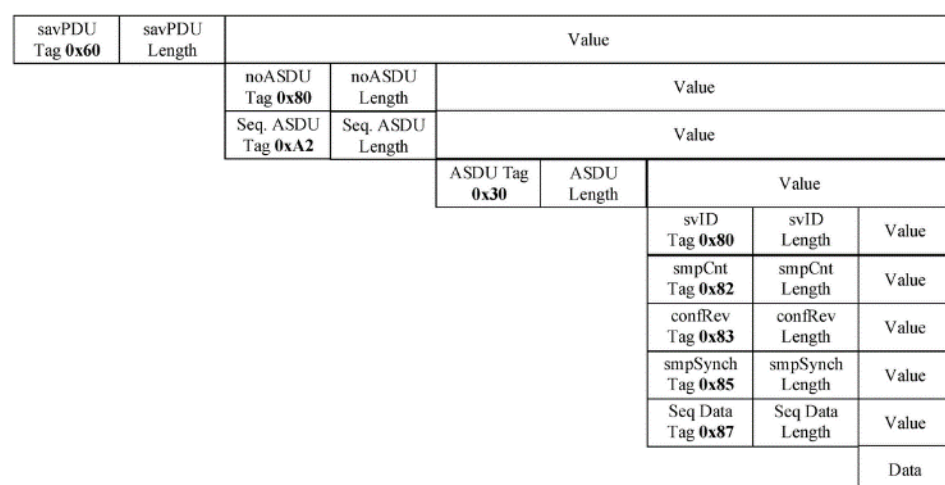
Table 6. rapidIEC61850 Review Summary.

Evaluation Criterion	Score	Notes
Tool Capabilities	0—Very poor	Tool does not work.
Cost	N/A	Free.
Customer Service	0—Very poor	Developer does not answer questions or help solve encountered problems, but they claim the tool works, which is more confusing.
Licensing	N/A	No license is required.
Time required for Installation and Use	−5—Terrible	Due to developer’s false claims and poor documentation, it is expected to work. Researchers waste a lot of time and effort trying when it clearly does not work.
Reliability	−5—Terrible	The tool does not work but it requires long investigations to discover this.

3.5. GoSV, S-GoSV and R-GoSV

GoSV framework has been developed to build GOOSE and SV messages from scratch, hence the name [63]. Manually building these messages with a low-level detail enables cutting edge research where novel approaches are tested. Such novel ideas may include adding new fields into the message body which may not yet be a part of IEC 61850 standard. The tools available in the market, such as Infotech, have to adhere to IEC 61850 standard for compatibility. However, this makes it impossible to try a new proposal in the IEC 61850 research field. This need is addressed by this tool.

The messages are built byte-by-byte following the description of respective payload fields as defined in IEC 61850. Figures 28 and 29 shows structures used to build an SV or a GOOSE message following the Tag, Length, Value (TLV) approach. Technical fields that are required for transmission in the network or identification of the message are also manually configured. Once this is done, the skeleton of the message is ready and it can be transmitted with any payload, i.e., dataset that is sent via GOOSE and SV message. In order to do so, it is required to perform socket programming and ethernet frame creation. The steps required to execute these tasks are shown in Figure 30. The tool is developed and run on Linux as it requires low-level access to network devices.

**Figure 28.** Byte-by-byte structure of an SV message.

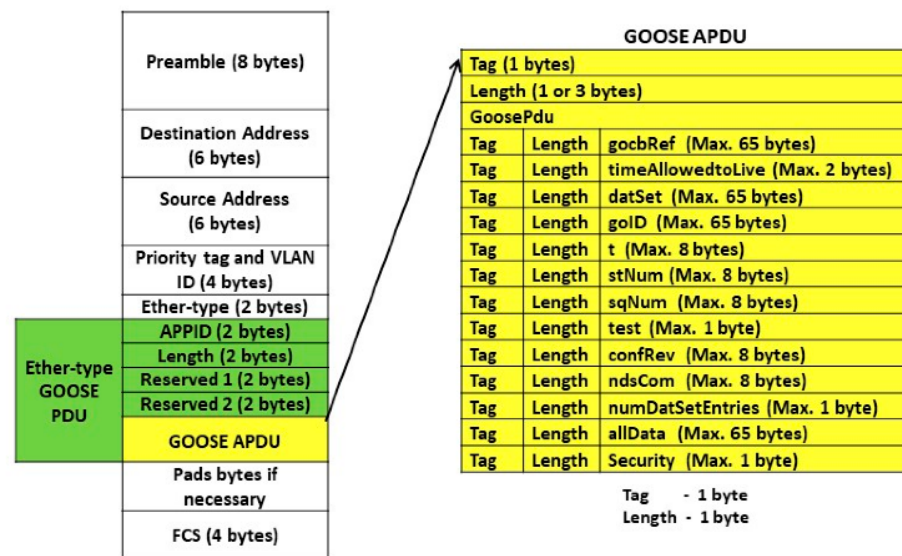


Figure 29. Internal structure of a GOOSE message.

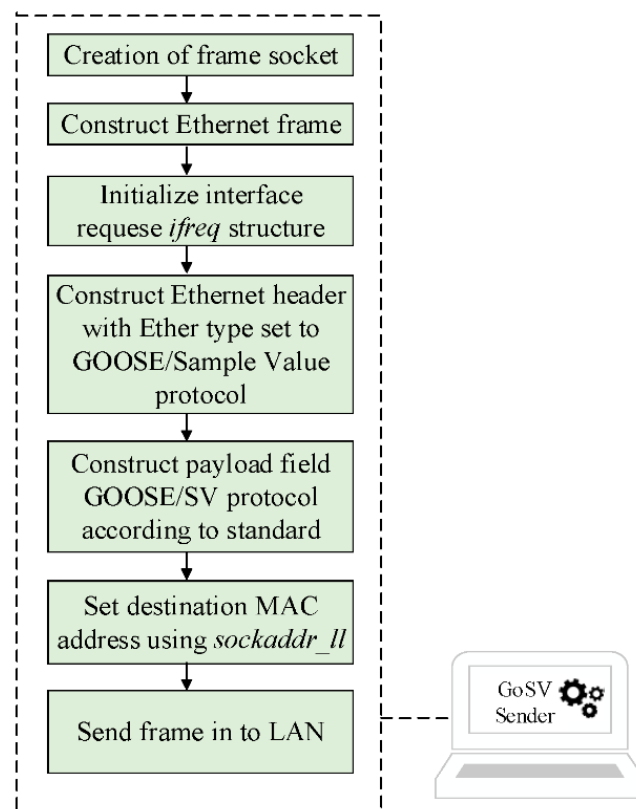


Figure 30. Socket Programming for message transmission.

The motivation behind building the GoSV tool is not to send regular GOOSE and SV messages. As it is covered above, there are different tools that can achieve that. Instead, the real focus is to be able to modify GOOSE and SV messages in such a way that different cybersecurity techniques can be added and tested. There is no other tool that has the ability to do this, other than S-GoSV. Some of these cybersecurity techniques are recommended in newly revised IEC 62351 standard [64]. However, the tool is not limited by the contents of the standard. Non-standard techniques can be implemented and tested [65]. Results of such cutting-edge research work are utilized to revise and modify the standards. For example, IEC 62351 standards recommended use of RSA digital signatures for securing

GOOSE messages. However, research has shown that the RSA based digital signatures results in larger computational delays and cannot meet the strict timing requirement of GOOSE messages. Hence, researchers proposed symmetric key based MAC algorithms to secure the GOOSE messages. The revised editions of IEC 62351 standards have adopted MAC algorithms.

S-GoSV has the ability to establish secure connection between subscriber and publisher by means of digital signatures. As shown in Figure 31, RSASSA-PKCS1-v1_5 is utilized for this purpose, i.e., public key cryptography standard 1, version 1.5. To be able to carry the new digital signature with the message, GOOSE ethernet frame is modified as shown in Figure 32. With these changes, subscribers can check the authenticity of the publishers and may opt to process or discard incoming GOOSE messages.

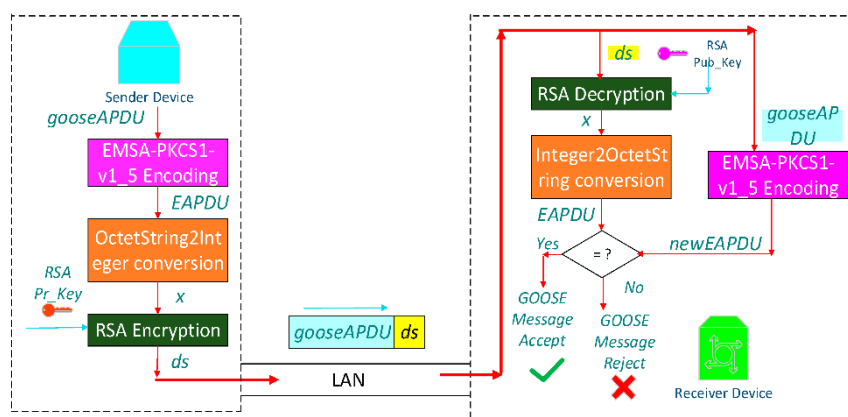


Figure 31. Authentication mechanism with digital signatures.

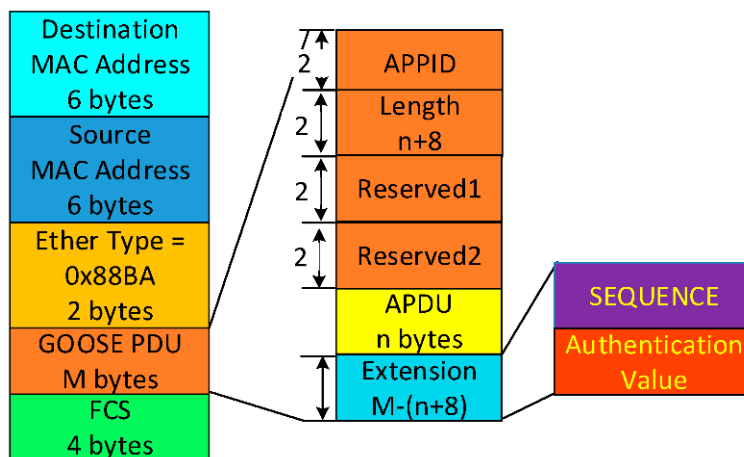


Figure 32. Modification of GOOSE message contents for security.

Another security feature that is available in S-GoSV is message integrity. This is performed with hash functions, e.g., Keyed Hash based Message Authentication Code (HMAC) as shown in Figure 33, where the hash value is appended to the message body. The hash value is also encrypted with a shared key between publisher and subscriber. The subscriber can check whether the received message corresponds to the appended hash value. If not, it shows that the contents have been changed during transmission. HMAC is implemented with openssl libraries which makes it possible to select other hashing techniques. Researchers may modify the code and investigate different digital signature or hashing techniques, their performances and operation. The main use of S-GoSV is for answering such cybersecurity questions.

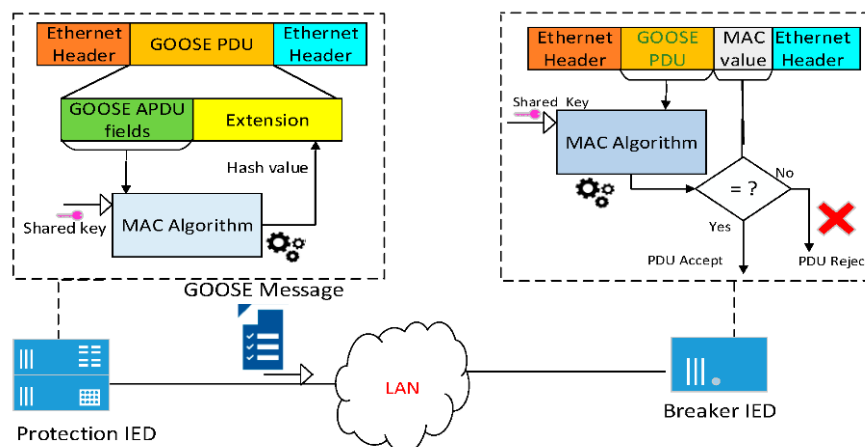


Figure 33. HMAC for ensuring message integrity.

As mentioned above, these tools are developed in Linux and their use requires some programming knowledge. This is especially true if the researchers would like to modify the contents to try different techniques or approaches. Programs need to be compiled with GCC compiler and run in terminal window as shown in Figure 34. These messages are represented in bytes which are detected successfully by other tools such as Wireshark as shown in Figure 35. Different messages sent by S-GoSV with security features are shown in Figures 36 and 37. Interoperability tests have been performed with different tools across Windows and Linux operating systems [63].

```

root@crypto-HP-2000-Notebook-PC: ~/finalprograms/sec-GOOSEPKCS1-v1_5
30:42:10:14:12:15:12:64:11:12:18:22:14:12:17:16:30:42:10:14:12:15:12:64:11:12:18:
:22:14:12:17:16:30:42:10:14:12:15:12:64:11:12:18:22:14:12:17:16:

Sample Value (SV) Message Sent :
01:0c:cd:01:03:ff:a0:b3:cc:c5:77:a1:81:00:80:00:88:ba:40:00:00:6e:00:00:00:00:60
:64:80:01:01:a2:5f:30:5d:80:0c:46:52:45:41:2d:47:6f:53:56:2d:30:31:82:02:0c:a4:8
3:04:00:00:00:01:85:01:00:87:40:30:42:10:14:12:15:12:64:11:12:18:22:14:12:17:16:
30:42:10:14:12:15:12:64:11:12:18:22:14:12:17:16:30:42:10:14:12:15:12:64:11:12:18:
:22:14:12:17:16:30:42:10:14:12:15:12:64:11:12:18:22:14:12:17:16:

Sample Value (SV) Message Sent :
01:0c:cd:01:03:ff:a0:b3:cc:c5:77:a1:81:00:80:00:88:ba:40:00:00:6e:00:00:00:00:60
:64:80:01:01:a2:5f:30:5d:80:0c:46:52:45:41:2d:47:6f:53:56:2d:30:31:82:02:0c:a4:8
3:04:00:00:00:01:85:01:00:87:40:30:42:10:14:12:15:12:64:11:12:18:22:14:12:17:16:
30:42:10:14:12:15:12:64:11:12:18:22:14:12:17:16:30:42:10:14:12:15:12:64:11:12:18:
:22:14:12:17:16:30:42:10:14:12:15:12:64:11:12:18:22:14:12:17:16:
    
```

Figure 34. SV Sender in Go-SV.

```

▶ Frame 1: 128 bytes on wire (1024 bits), 128 bytes captured (1024 bits) on
▶ Ethernet II, Src: HewlettP_c5:77:a1 (a0:b3:cc:c5:77:a1), Dst: Tec-Tc57_01
▶ 802.1Q Virtual LAN, PRI: 4, DEI: 0, ID: 0
  IEC61850 Sampled Values
    APPID: 0x4000
    Length: 110
    Reserved 1: 0x0000 (0)
    Reserved 2: 0x0000 (0)
    savPdu
      noASDU: 1
      seqASDU: 1 item
        ASDU
          svID: FREA-GoSV-01
          smpCnt: 3236
          confRef: 1
          smpSynch: none (0)
          seqData: 304210141215126411121822141217163042101412151264...
0000  01 0c cd 01 03 ff a0 b3 cc c5 77 a1 81 00 80 00  ..w....
0010  88 ba 40 00 00 6e 00 00 00 00 60 64 80 01 01 a2  ..@.n...d...
0020  5f 30 5d 80 0c 46 52 45 41 2d 47 6f 53 56 2d 30  _0]..FRE A-GoSV-0
0030  31 82 02 0c a4 83 04 00 00 00 01 85 01 00 87 40  1.....@
0040  30 42 10 14 12 15 12 64 11 12 18 22 14 12 17 16  0B.....d "....
0050  30 42 10 14 12 15 12 64 11 12 18 22 14 12 17 16  0B.....d "....
0060  30 42 10 14 12 15 12 64 11 12 18 22 14 12 17 16  0B.....d "....
0070  30 42 10 14 12 15 12 64 11 12 18 22 14 12 17 16  0B.....d "....
    
```

Figure 35. SV Message transmitted by GoSV.

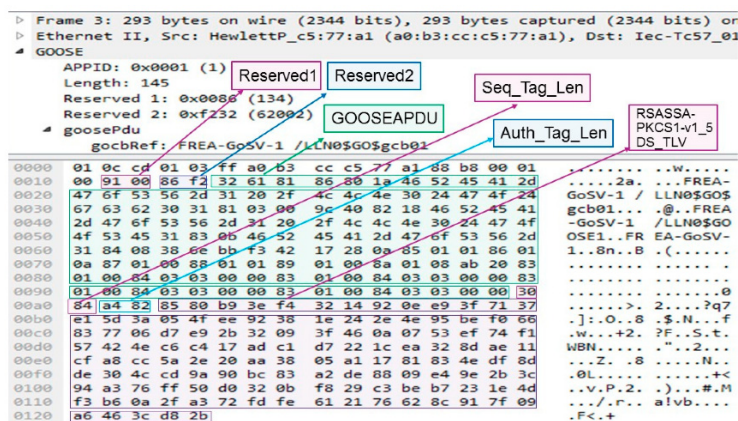


Figure 36. Secure GOOSE Message by S-GoSV with digital signature.

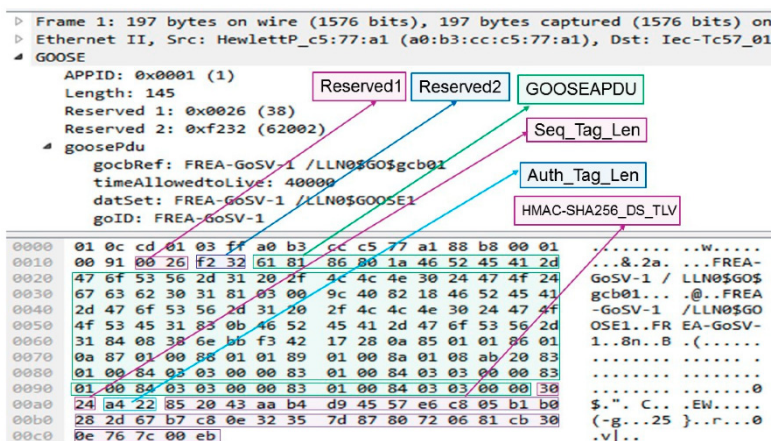


Figure 37. Secure GOOSE Message by S-GoSV with hash value.

The final addition to this tool is routable messages, R-GOOSE and R-SV, as per IEC 61850-90-5 rules. These messages are used to secure phasor measurement unit communication and follow the structure shown in Figure 38. Once implemented, R-GoSV toolbox is able to send regular and secured (encrypted) R-GOOSE/R-SV messages as shown in Figures 39 and 40, respectively. It is important to highlight here that, in the latter case, the message cannot be successfully detected since the contents are encrypted.

In short, the GoSV tool is able to create custom GOOSE and SV messages. These can be secured with S-GoSV and made routable with R-GoSV. Tools are used via the command line, similar to libIEC61850, and modification of the samples requires a high-level of programming knowledge. It has features that are unique for cybersecurity considerations in IEC 61850 communication networks. Table 7 summarizes these findings.

Table 7. GoSV, S-GoSV and R-GoSV Tools Review Summary.

Evaluation Criterion	Score	Notes
Tool Capabilities	2—Limited focus	Only GOOSE and SV messages, but routable options as well as cybersecurity features are available.
Cost	N/A	Free.
Customer Service	3—Acceptable	Developer is active and responds to questions, but the tool comes with no guarantee or support.
Licensing	N/A	No license is required.
Time required for Installation and Use	2—Difficult	Tools are ready for use with default options. However, modification requires learning socket programming in Linux.
Reliability	4—Good	The tools are robust and reliable. However, they have many dependencies such as openssl or GCC and these may cause issues. Interoperability tests with other tools also proved their successful operation.

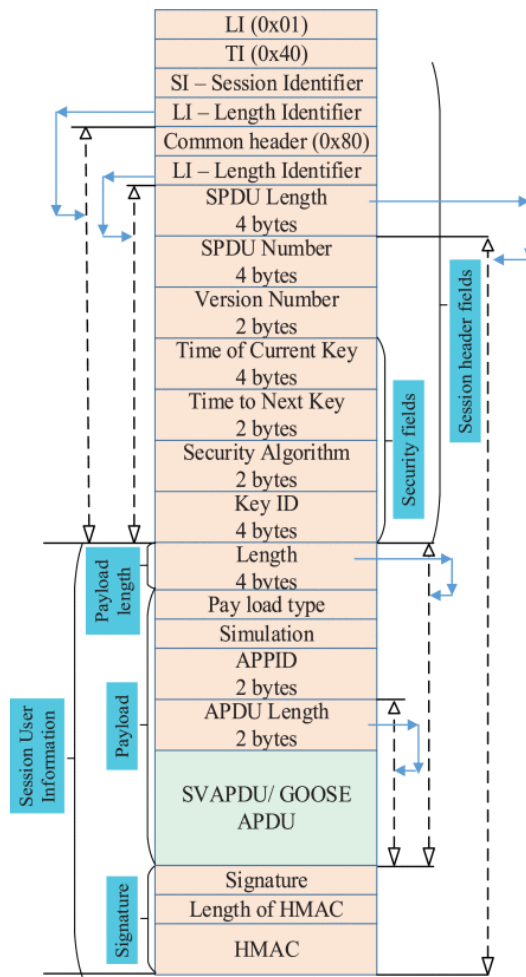


Figure 38. Session Layers in IEC 61850-90-5.

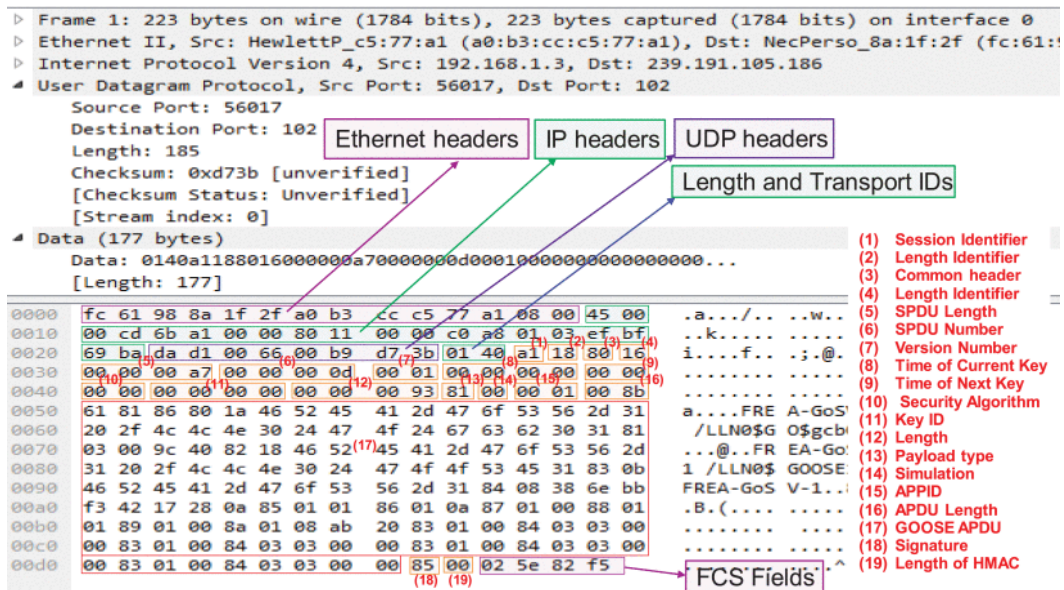


Figure 39. R-GOOSE without security by R-GoSV.

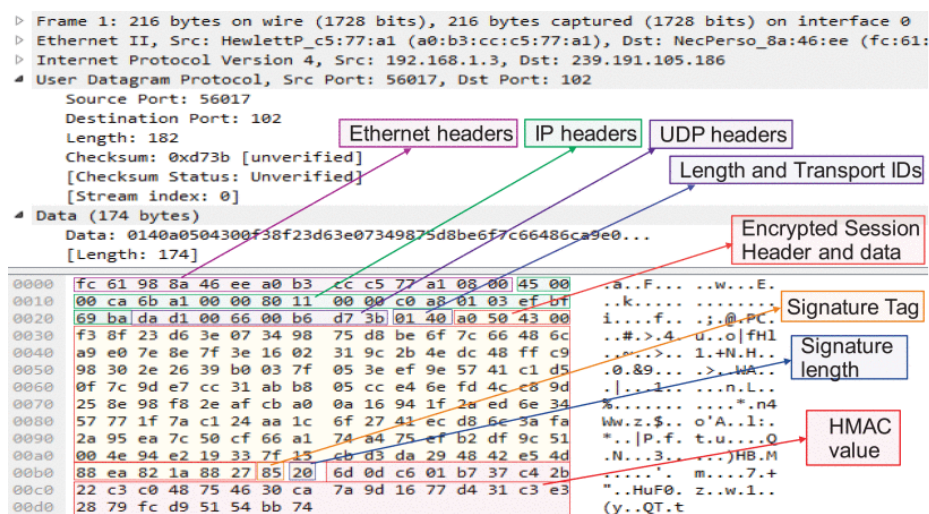


Figure 40. R-SV with security features by R-GoSV.

4. Network Emulators

This section presents an overview of two widely used network simulator tools for IEC 61850 communication networks, namely Riverbed Modeler and a much more cost-effective alternative, Netsim.

4.1. Riverbed Modeler (Formerly Known as OPNET Modeler)

Riverbed Modeler (formerly OPNET modeler) is a Proto-C based object-oriented network simulator software [66]. It provides high-fidelity modeling, scalable simulation, and detailed analysis of a broad range of wired and wireless networks.

Riverbed Modeler contains editors such as Project editor, Node editor, Process editor, and Open model source code for detailed customization of the communication network. Figure 41 illustrates different editors in Riverbed Modeler. The project editor is used to design the overall architecture of the communication network. The Riverbed Modeler has a very rich library with node models for different communication network components such as switches, routers, links, servers, etc. Custom nodes can also be developed and added to the library using the node editor. Node editor consists of different blocks, each depicting a process. The process of each block can be edited in the process editor tool of the software. Thus, using this tier level modeling approach, different customized node models can be developed.

In the literature, Riverbed Modeler has been extensively used to model and evaluate the performance of IEC 61850 communication networks. The library includes a wide variety of wire and wireless communication models. Basically, the rich library of Riverbed Modeler has provided off the shelf nodes to exactly model different IEC 61850 based IEDs. The ‘ethernet_station_adv’ node from library is used to model the Merging Unit (MU) IEDs which generate the GOOSE and SV messages which are layer 2 messages directly mapped to Ethernet layer. Figure 42 shows the node model used for modeling MU IEDs. Similarly, ‘Ethernet_workstation_adv’ node model is used to model other IEDs which exchange MMS messages having complete OSI stack. Figure 43 shows the node model for IEDs generating MMS messages.

The traffic in the Riverbed Modeler is set by using ‘Application Config’, ‘Profile Config’ and ‘Task Config’ nodes. In ‘Application Config’ different traffic attributes such as inter-arrival time, size of packet, etc., for each application can be defined. The different applications defined in ‘Application Config’ are grouped for one particular node in ‘Profile Config’. Figure 44 shows Profile configuration in Riverbed Modeler. Using these options different types of traffic can be configured.

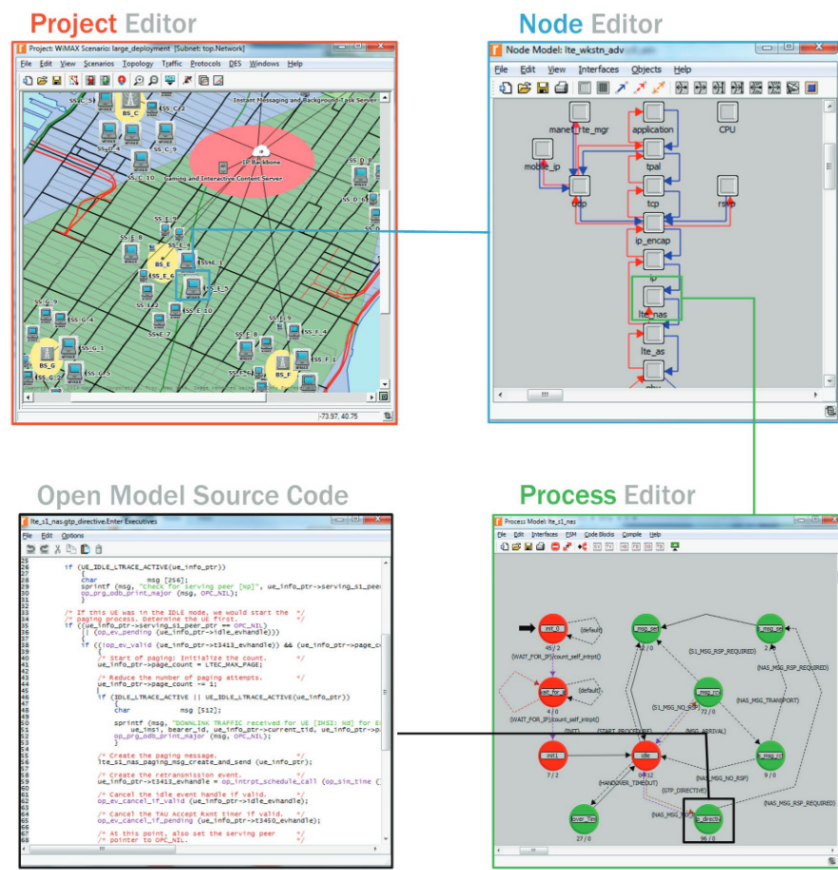


Figure 41. Different Editors in Riverbed Modeler.

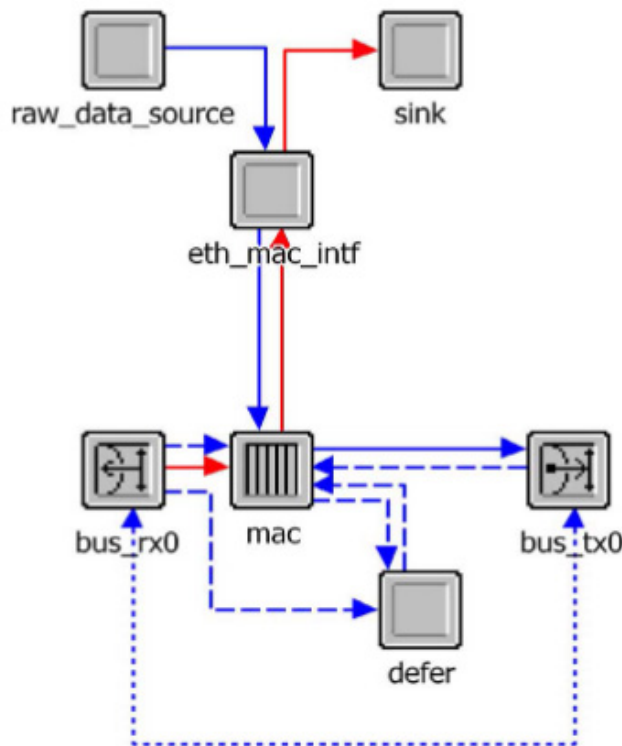


Figure 42. 'ethernet_station_adv' Node model.

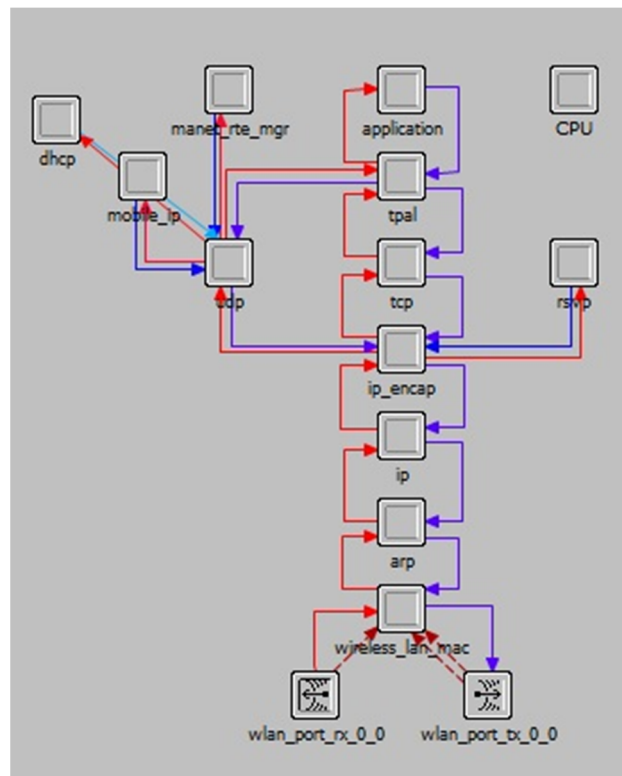


Figure 43. 'ethernet_workstation_adv' Node model.

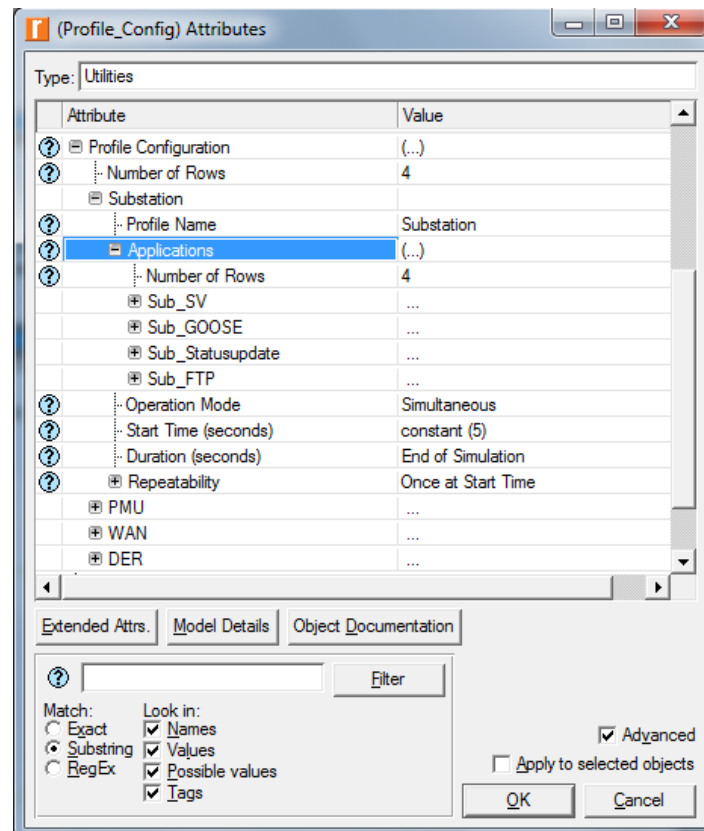


Figure 44. Profile configuration for different messages.

Riverbed Modeler provides a variety of performance statistics for the simulation. It provides the option for global level, node level and link level statistics such as End to End delays, throughput, packet loss, etc., at each protocol level.

One of the important features of Riverbed Modeler is the System-in-the-Loop (SITL) module. SITL provides an interface for connecting real network traffic (hardware and software applications) with discrete event simulation traffic inside Riverbed Modeler Simulation. It extends the scope of Riverbed Modeler functionality by enabling technology testing and training. The SITL functionality of Riverbed Modeler has been exploited by the researchers to interface the real/emulated IEC 61850 IEDs with communication network inside the simulation.

Riverbed Modeler has also an academic edition which is available freely for education use. This version has limited capabilities and limited libraries. The advanced libraries and features such as LTE, Wimax, SITL module are not available. Furthermore, the node and process editors are not accessible, hence custom nodes cannot be created with the academic edition. Nevertheless, the academic edition is still well equipped for carrying out basic simulation studies.

Overall, Riverbed Modeler has been an effective tool for modeling and testing the IEC 61850 communication networks. Table 8 summarizes its features.

Table 8. Riverbed Modeler Review Summary.

Evaluation Criterion	Score	Notes
Tool Capabilities	5—Excellent	Rich library, node and process level editing capabilities
Cost	2—Expensive	Expensive
Customer Service	4—Acceptable	Customer support is good. Responds to all queries.
Licensing	4—Acceptable	Tied to MAC address or floating licenses over a server
Time required for Installation and Use	5—Excellent	Installation packages are reliable. Installation is very smooth.
Reliability	4—Good	Very Good.

4.2. Netsim

Similar to Rivered Modeler, NetSim is also a commercial network simulator tool with similar features [67]. NetSim is also a GUI based simulator tool with a rich library. The user interface is friendly and readily deployable as shown in Figure 45. NetSim has three versions: academic, standard and pro. The academic version only includes basic libraries/components and is limited to 100 node simulation scale. The standard version includes all the libraries/components and simulation scale is 500 nodes, whereas the perversion has simulation scale of 10000 nodes. For the IEC 61850 research evaluations, the standard version is the most suitable option.

The licenses are managed through a license server as shown in Figure 46. The benefit of this approach is that different terminals can run NetSim easily, provided that they do not work simultaneously. For dynamic research labs, this is a very useful feature.

Compared to Riverbed Modeler, the library of NetSim is smaller. However, all the basic and essential node models for different advanced wired and wireless technologies such as 5G, cognitive radio, etc., are available. The NetSim software does not provide any process or node level editors. Thus, the task of creating custom nodes in NetSim is cumbersome. Network emulator feature is add-on in NetSim software. The Network emulator feature allows NetSim simulation to interface with real traffic, i.e., hardware.

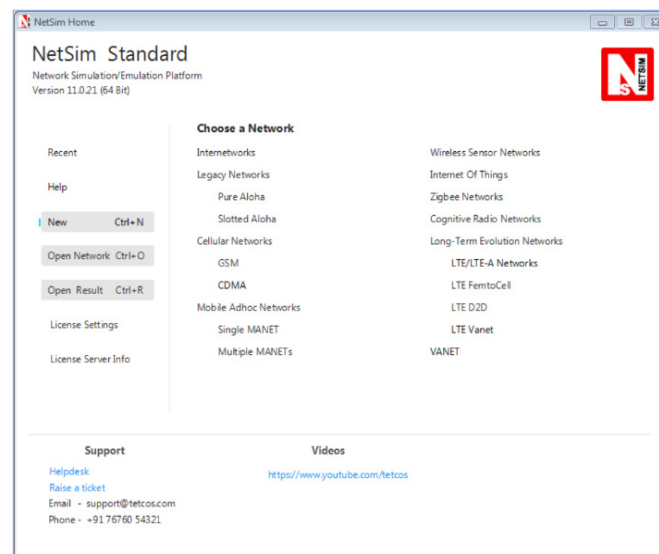


Figure 45. Netsim Standard GUI.



Figure 46. Netsim License Server Configuration.

Despite some lesser features compared to Riverbed Modeler, in the context of IEC 61850 communication network modeling NetSim has all the features and is a very cost-effective alternative. It can be utilized to model and test complex IEC 61850 communication networks such as the microgrid energy management system shown in Figure 47 [68]. Table 9 provides the review summary of the NetSim network simulator tool.

Table 9. NetSim Review Summary.

Evaluation Criterion	Score	Notes
Tool Capabilities	5—Excellent	Rich library, node and process level editing capabilities
Cost	4—Acceptable	Costs 20% of its competitors' price. Libraries are not as rich but worth the cost.
Customer Service	4—Acceptable	Customer support is good. Responds to all queries. Online tutorials are available after purchase
Licensing	4—Acceptable	License server with PC dongle or tied to MAC address
Time required for Installation and Use	5—Excellent	Installation packages are reliable.
Reliability	4—Good	Installation is very smooth.

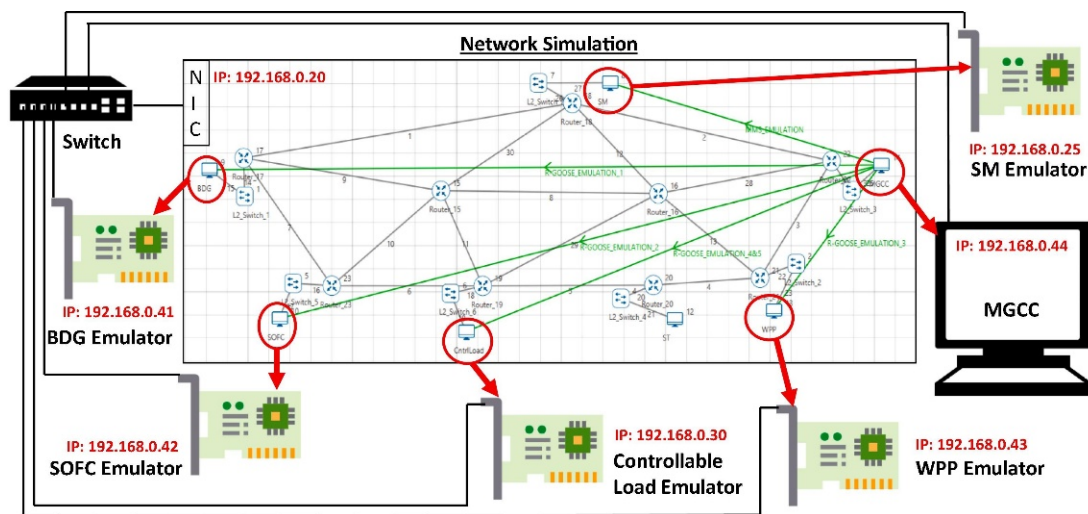


Figure 47. IEC 61850 based microgrid Energy Management System Implemented in NetSim.

5. Critical Comparison of the IEC61850 Testing Tools

Based on the lengthy and detailed review presented above, this section renders a final verdict on every tool covered in this paper. This is intended to help researchers and practitioners make an educated decision while choosing a test tool. Needless to say, this is purely academic and is an effort to contribute the acquired experience into the literature.

As shown in Table 10, there are three major types of IEC 61850 testing tools. The first one is the Network Analyzer. In this field, Wireshark is the pioneer as it is free, has great performance and is easy to use. It also has an active online community. Therefore, troubleshooting rare problems or getting help with novel ideas is straightforward.

Table 10. Critical Comparison of IEC 61850 Tools.

Type	Tool	Final Verdict
Network Analyzer	Wireshark	The one and only. Free too!
	Infotech	Very good investment, strong tool!
	Xelas	Bad tool, terrible company. Avoid at all costs!
IEC 61850 Emulators	libIEC61850	If you are programmer, very strong and flexible
	rapidIEC61850	Do NOT waste time with this useless tool!
	(S-R)Go-SV	Very robust for GOOSE and SV messages
	Riverbed Modeler	Very costly. Trial version is a good option.
Network Emulators	Netsim	As strong as Riverbed Modeler (at least for IEC 61850 communication networks), cost effective, easy to use!

When it comes the IEC 61850 emulator tools, the market certainly has a wide spectrum. Of the paid tools, Infotech is the reliable, cost-effective and researcher-friendly option. Despite being ten times costlier, Xelas Energy does not provide even the most basic IEC 61850 services. The negative attitude of the company and its personnel is definitely a more important reason to avoid them at any cost.

The free tools are generally good, such as libIEC61850 and Go-SV. They require some programming skills, but simple tasks can be performed by tweaking the provided samples. The biggest pitfall in this category is tools with dishonest claims that cause waste of time and resources, such as rapidIEC61850. Despite the claims of the developer, this tool does not work, and researchers should save their patience and energy for more productive tasks.

Finally, both Network Emulators perform very similarly for IEC 61850 communication network evaluations. Riverbed Modeler has a very rich library with a greater number of nodes. The process model and attributes in Riverbed Modeler are superior compared to NetSim. However, for IEC 61850 based communication networks the performance of both the simulators is acceptable. They are easy to use and improve research results significantly. This is due to the fact that they convert in-lab tests to pseudo-real-life tests with traffic and physical distance. Of the two, Netsim is the more cost-effective option.

6. Conclusions

There are an increasing number of research tools available in the market. Some of them are licensed while others are open source. Selecting the correct tool for any research is critical and can save a lot of time and effort. Therefore, benefiting from the experiences of other researchers who are familiar with these tools is very important. There are different works in the literature that provide a comparative review for different, and typically quite narrow, fields. This paper fills a huge knowledge gap by providing a comparative review for IEC 61850 research which is a very popular field.

The authors' experience with IEC 61850 tools available in the market has been a mixed bag. While some have caused either, or both, loss of time and funds while, others turned out to be good investments. This work shares these valuable insights that requires a lot of time, funds and effort to gather. Instead of reinventing the wheel, readers can benefit from this experience. Overall, this will massively increase efficiency of research projects and save a lot of time and money.

Funding: This research received no external funding.

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

References

- Momoh, J. *Smart Grid: Fundamentals of Design and Analysis*; Wiley-IEEE Press: Hoboken, NJ, USA, 2012; ISBN 9781118156117.
- Kahouli, O.; Alsaif, H.; Bouteraa, Y.; Ben Ali, N.; Chaabene, M. Power System Reconfiguration in Distribution Network for Improving Reliability Using Genetic Algorithm and Particle Swarm Optimization. *Appl. Sci.* **2021**, *11*, 3092. [CrossRef]
- Alhasnawi, B.; Jasim, B.; Sedhom, B.; Hossain, E.; Guerrero, J. A New Decentralized Control Strategy of Microgrids in the Internet of Energy Paradigm. *Energies* **2021**, *14*, 2183. [CrossRef]
- Ustun, T.S.; Aoto, Y. Analysis of Smart Inverter's Impact on the Distribution Network Operation. *IEEE Access* **2019**, *7*, 9790–9804. [CrossRef]
- Yang, Q.; Yang, T.; Li, W. *Smart Power Distribution Systems*; Academic Press: Cambridge, MA, USA, 2019; ISBN 9780128121542.
- Nogaret, E.; Stavrakakis, G.; Kariniotakis, G.; Papadopoulos, M.; Hatziargyriou, N.; Androutsos, A.; Papathanassiou, S.; Lopes, J.; Halliday, J.; Dutton, G.; et al. An advanced control system for the optimal operation and management of medium size power systems with a large penetration from renewable power sources. *Renew. Energy* **1997**, *12*, 137–149. [CrossRef]
- Ahmad, T.; Zhang, D. Using the internet of things in smart energy systems and networks. *Sustain. Cities Soc.* **2021**, *68*, 102783. [CrossRef]
- Global Smartgrid Federation. Smart Grid Interoperability Report, June 2014. Available online: <https://bit.ly/3dFzGbK> (accessed on 22 April 2021).
- International Electrotechnical Commission (IEC). *IEC TR 61850 Standard*; IEC: Geneva, Switzerland, 2013.
- Ustun, T.S.; Hussain, S.M.S. Extending IEC 61850 Communication Standard to Achieve Internet-of-Things in Smartgrids. In Proceedings of the 2019 International Conference on Power Electronics, Control and Automation (ICPECA), New Delhi, India, 16–17 November 2019; pp. 1–6.
- Zhechen, H.; Lei, G.; Yi, Y.; Xiangping, K.; Jinjiao, L. Chapter 2—IEC 61850 Standards and Configuration Technology, *IEC 61850-Based Smart Substations*; Academic Press: Cambridge, MA, USA, 2019; pp. 25–62, ISBN 9780128151587.
- Falk, H. *IEC 61850 Demystified*; Artech House: Washington, DC, USA, 2019; pp. 1–50, ISBN 978-1-63081-329-1.
- Aftab, M.A.; Hussain, S.S.; Ali, I.; Ustun, T.S. IEC 61850 based substation automation system: A survey. *Int. J. Electr. Power Energy Syst.* **2020**, *120*, 106008. [CrossRef]
- Power Systems Management and Associated Information Exchange-Data and Communications Security, Part. 6: Security for IEC IEC 62351-6*; IEC: Geneva, Switzerland, 2007.
- Haffar, J.M.; Thiriet, E.; Savary, M. Modeling of substation architecture implementing IEC 61850 protocol and solving inter-locking problems. In Proceedings of the IFAC Proceedings Volumes; IEEE: New York, NY, USA, 2007; Volume 40, pp. 291–294.
- Aftab, M.A.; Hussain, S.M.S.; Ali, I.; Ustun, T.S. A Novel SCL Configuration Method for Modeling Microgrids with IEC. *IEEE Syst. J.* **2020**, *14*, 2676–2683. [CrossRef]

17. Nimma, K.S.; Faraj, S.N. Modeling Intelligent Control Switch IEC 61850 Based Substation Automation Communication. *Appl. Syst. Innov.* **2018**, *1*, 7. [CrossRef]
18. Ali, N.; Eissa, M. Accelerating the protection schemes through IEC 61850 protocols. *Int. J. Electr. Power Energy Syst.* **2018**, *102*, 189–200. [CrossRef]
19. Ustun, T.S.; Farooq, S.M.; Hussain, S.M.S. Implementing Secure Routable GOOSE and SV Messages Based on IEC 61850-90-5. *IEEE Access* **2020**, *8*, 26162–26171. [CrossRef]
20. Libiec61850, IEC 61850 Library. Available online: <http://libiec61850.com/libiec61850/> (accessed on 16 March 2021).
21. Xelas Energy. Available online: <http://xelasenergy.com/> (accessed on 16 April 2021).
22. Holmgren, W.F.; Hansen, C.W.; Stein, J.S.; Mikofski, M.A. Review of Open Source Tools for PV Modeling. In Proceedings of the 2018 IEEE 7th World Conference on Photovoltaic Energy Conversion (WCPEC) (A Joint Conference of 45th IEEE PVSC, 28th PVSEC & 34th EU PVSEC), Waikoloa, HI, USA, 10–15 June 2018; pp. 2557–2560.
23. Mahmud, K.; Town, G.E. A review of computer tools for modeling electric vehicle energy requirements and their impact on power distribution networks. *Appl. Energy* **2016**, *172*, 337–359. [CrossRef]
24. Stanica, R.; Chaput, E.; Beylot, A.-L. Simulation of vehicular ad-hoc networks: Challenges, review of tools and recommendations. *Comput. Networks* **2011**, *55*, 3179–3188. [CrossRef]
25. Moazami, A.; Carlucci, S.; Geving, S. Critical Analysis of Software Tools Aimed at Generating Future Weather Files with a view to their use in Building Performance Simulation. *Energy Procedia* **2017**, *132*, 640–645. [CrossRef]
26. Naboni, E.; Meloni, M.; Coccolo, S.; Kaempfer, J.; Scartezzini, J.-L. An overview of simulation tools for predicting the mean radiant temperature in an outdoor space. *Energy Procedia* **2017**, *122*, 1111–1116. [CrossRef]
27. Allegrini, J.; Orehoung, K.; Mavromatidis, G.; Ruesch, F.; Dorer, V.; Evins, R. A review of modelling approaches and tools for the simulation of district-scale energy systems. *Renew. Sustain. Energy Rev.* **2015**, *52*, 1391–1404. [CrossRef]
28. Abbasabadi, N.; Ashayeri, M. Urban energy use modeling methods and tools: A review and an outlook. *Build. Environ.* **2019**, *161*, 106270. [CrossRef]
29. Tozzi, P.; Jo, J.H. A comparative analysis of renewable energy simulation tools: Performance simulation model vs. system optimization. *Renew. Sustain. Energy Rev.* **2017**, *80*, 390–398. [CrossRef]
30. Vera, S.; Pinto, C.; Tabares-Velasco, P.C.; Bustamante, W. A critical review of heat and mass transfer in vegetative roof models used in building energy and urban environment simulation tools. *Appl. Energy* **2018**, *232*, 752–764. [CrossRef]
31. Kaur, D.; Cheema, P.S. Software tools for analyzing the hybrid renewable energy sources—A review. In Proceedings of the 2017 International Conference on Inventive Systems and Control (ICISC), Coimbatore, India, 19–20 January 2017; pp. 1–4.
32. Nakkasunchi, S.; Hewitt, N.J.; Zoppi, C.; Brandoni, C. A review of energy optimization modelling tools for the decarbonisation of wastewater treatment plants. *J. Clean. Prod.* **2021**, *279*, 123811. [CrossRef]
33. Zhang, Y.; Ke, C.; Fu, W.; Cui, Y.; Rehan, M.A.; Li, B. Simulation of microwave-assisted gasification of biomass: A review. *Renew. Energy* **2020**, *154*, 488–496. [CrossRef]
34. Barela, S. Software review for automatic test equipment. IEEE Autotestcon, 2005, Orlando, FL, USA, 26–29 September 2005; pp. 30–35. [CrossRef]
35. Desmond, C. Project management tools—software tools. *IEEE Eng. Manag. Rev.* **2017**, *45*, 24–25. [CrossRef]
36. Byrne, J.; Heavey, C.; Byrne, P. A review of Web-based simulation and supporting tools. *Simul. Model. Pr. Theory* **2010**, *18*, 253–276. [CrossRef]
37. Muhanji, S.O.; Flint, A.E.; Farid, A.M. *The Development of the Energy Internet of Things in Energy Infrastructure*, 1st ed.; Springer: Berlin/Heidelberg, Germany, 2019; ISBN 978-3-030-10427-6.
38. National Institute of Standards and Technology. *NIST Framework and Roadmap for Smart Grid Interoperability Standards*; NIST Special Publication 1108; National Institute of Standards and Technology: Gaithersburg, MD, USA, 2010.
39. International Electrotechnical Commission (IEC). *Communication Networks and Systems for Power Utility Automation, Part 90-7: Object Models for Power Converters in Distributed Energy Resources (DER) Systems*; Standard IEC/TR61850-90-7; International Electrotechnical Commission (IEC): Geneva, Switzerland, 2013.
40. Ustun, T.S.; Ozansoy, C.; Zayegh, A. Simulation of communication infrastructure of a centralized microgrid protection system based on IEC 61850-7. In Proceedings of the 2012 IEEE Third International Conference on Smart Grid Communications (SmartGridComm), Tainan, Taiwan, 5–8 November 2012; pp. 492–497.
41. Hussain, S.M.S.; Aftab, M.A.; Nadeem, F.; Ali, I.; Ustun, T.S. Optimal Energy Routing in Microgrids With IEC 61850 Based Energy Routers. *IEEE Trans. Ind. Electron.* **2020**, *67*, 5161–5169. [CrossRef]
42. Miao, D.; Dostanov, J.; Redfern, M. Using IEC 61850 data transfer beyond the substation for enhanced protection for distribution networks. In Proceedings of the 2013 48th International Universities Power Engineering Conference (UPEC), Dublin, Ireland, 2–5 September 2013; pp. 1–6.
43. Apostolov, A.; Brunner, C.; Clinard, K. Use of IEC 61850 object models for power system quality/security data exchange. *Int. Sympos. Qual. Secur. Elect. Power Deliv. Syst.* **2003**, 155–164. [CrossRef]
44. Wireshark, Network Analyzer. Available online: <https://www.wireshark.org/> (accessed on 19 April 2021).
45. Infotech Company. Available online: <https://www.infotech.pl/> (accessed on 19 April 2021).
46. Nadeem, F.; Aftab, M.A.; Hussain, S.S.; Ali, I.; Tiwari, P.K.; Goswami, A.K.; Ustun, T.S. Virtual Power Plant Management in Smart Grids with XMPP Based IEC 61850 Communication. *Energies* **2019**, *12*, 2398. [CrossRef]

47. Ustun, T.S.; Hussain, S.M.S. IEC 61850 Modeling of UPFC and XMPP Communication for Power Management in Microgrids. *IEEE Access* **2020**, *8*, 141696–141704. [CrossRef]
48. Ustun, T.S.; Ozansoy, C.; Zayegh, A. Extending IEC 61850-7-420 for distributed generators with fault current limiters. In *2011 IEEE PES Innovative Smart Grid Technologies*; IEEE: Melbourne, Australia, 2011; pp. 1–8.
49. UCA International Users Group. Implementation Guideline for Digital Interface to Instrument Transformers Using IEC 61850-9-2, July 2004. Available online: http://iec61850.ucaiug.org/Implementation%20Guidelines/DigIF_spec_9-2LE_R2-1_040707-CB.pdf (accessed on 22 April 2021).
50. Aftab, M.A.; Hussain, S.S.; Ali, I.; Ustun, T.S. IEC 61850-Based Communication Layer Modeling for Electric Vehicles: Electric Vehicle Charging and Discharging Processes Based on the International Electrotechnical Commission 61850 Standard and Its Extensions. *IEEE Ind. Electron. Mag.* **2020**, *14*, 4–14. [CrossRef]
51. Glassfish Server. Available online: <https://www.oracle.com/middleware/technologies/glassfish-server.html> (accessed on 7 April 2021).
52. Cygwin. Linux on Windows. Available online: <https://www.cygwin.com/> (accessed on 19 April 2021).
53. MZ Automation. Available online: <https://bit.ly/2NCO6yS> (accessed on 22 March 2021).
54. Open MUC Software Solutions for Monitoring and Control Systems. Available online: <https://www.openmuc.org/iec-61850/> (accessed on 22 March 2021).
55. libIEC61850 Server Tutorial. Available online: <https://libiec61850.com/libiec61850/documentation/iec-61850-server-tutorial/> (accessed on 19 April 2021).
56. Fu, Q.; Chen, J. Design of experiment platform for digital substation based on IEC. In Proceedings of the 2016 5th International Conference on Computer Science and Network Technology (ICCSNT), Changchun, China, 10–11 December 2016; pp. 4–8.
57. Rosch, D.; Ruhe, S.; Schafer, K.; Nicolai, S. Local anomaly detection analysis in distribution grid based on IEC 61850-9-2 LE SV voltage signals. In Proceedings of the 2019 International Conference on Smart Energy Systems and Technologies (SEST), Porto, Portugal, 9–11 September 2019; pp. 1–6.
58. rapidIEC61850 Webpage. Available online: <http://stevenblair.github.io/rapid61850/> (accessed on 16 March 2021).
59. Blair, S. Electronic and Electrical Engineering. Available online: <https://www.strath.ac.uk/staff/blairstevenmacpherson/dr/> (accessed on 16 March 2021).
60. Standard-Charge, Erigrad Transnational Access, Selected User Projects. Available online: <https://erigrad.eu/transnational-access/selected-projects/> (accessed on 16 March 2021).
61. Ustun, T.; Hussain, S.; Syed, M.; Dambrauskas, P. IEC-61850-Based Communication for Integrated EV Management in Power Systems with Renewable Penetration. *Energies* **2021**, *14*, 2493. [CrossRef]
62. Ustun, T.S.; Hussain, S.M.S.; Kikusato, H. IEC 61850-Based Communication Modeling of EV Charge-Discharge Management for Maximum PV Generation. *IEEE Access* **2019**, *7*, 4219–4231. [CrossRef]
63. Farooq, S.M.; Hussain, S.S.; Ustun, T.S. S-GoSV: Framework for Generating Secure IEC 61850 GOOSE and Sample Value Messages. *Energies* **2019**, *12*, 2536. [CrossRef]
64. Hussain, S.M.S.; Ustun, T.S.; Kalam, A. A Review of IEC 62351 Security Mechanisms for IEC 61850 Message Exchanges. *IEEE Trans. Ind. Inf.* **2019**, *16*, 5643–5654. [CrossRef]
65. Farooq, S.M.; Hussain, S.M.S.; Ustun, T.S. Elliptic Curve Digital Signature Algorithm (ECDSA) Certificate Based Authentication Scheme for Advanced Metering Infrastructure. In Proceedings of the 2019 Innovations in Power and Advanced Computing Technologies (i-PACT), Vellore, India, 22–23 March 2019.
66. Riverbed Modeler, Discrete Event Simulator for Networks. Available online: www.riverbed.com (accessed on 21 April 2021).
67. Netsim. Network Simulator and Emulator by Tetcos. Available online: www.tetcos.com (accessed on 21 April 2021).
68. Aftab, M.A.; Hussain, S.S.; Latif, A.; Das, D.C.; Ustun, T.S. IEC 61850 communication based dual stage load frequency controller for isolated hybrid microgrid. *Int. J. Electr. Power Energy Syst.* **2021**, *130*, 106909. [CrossRef]