Real-Time Hardware-in-the-Loop Testbed to Evaluate FLISR Implemented with OpenFMB

Neil Shepard, P.E.
Grid Components and Controls
Oak Ridge National Laboratory
Oak Ridge, TN United States
0000-0003-3234-007X

Aditya Sundararajan
Grid Components and Controls
Oak Ridge National Laboratory
Oak Ridge, TN United States
0000-0003-3577-8544

Maximiliano Ferrari
Grid Components and Controls
Oak Ridge National Laboratory
Oak Ridge, TN United States
0000-0002-5884-7664

Ben Ollis
Grid Components and Controls
Oak Ridge National Laboratory
Oak Ridge, TN United States
0000-0001-9016-4268

***Abstract*—With the increasing complexity of the distribution smart grid architecture, algorithms such as the fault location, isolation, and service restoration (FLISR) scheme rely on robust communications that are resilient to natural and man-made adverse conditions and exhibit robustness. Existing communications infrastructure for information exchange are centralized at the distribution management system, with very little autonomy or intelligence at the grid-edge. As a first step towards achieving grid-edge self-healing, this paper aims to bridge this shortcoming by implementing a centrally coordinated rules-based FLISR scheme and integrating it with Open Field Message Bus (OpenFMB), which is a flexible publish-subscribe architecture with the potential to enable point-to-multipoint communications and is more robust and resilient to natural and man-made adverse conditions. A proof of concept is developed to validate the centrally coordinated FLISR and OpenFMB mounted on an SEL-3360 computer that interacts with a simple feeder network of five SEL-651R relays, an SEL-3530 RTAC, and a hardware-in-the-loop testbed. Results demonstrate the efficacy of this approach in enabling direct, low-latency information exchange. OpenFMB's publish-subscribe data model also opens new ways to enable grid-edge interoperability among devices of different vendors interacting with different protocols.**

***Index Terms*—FLISR, Fault Location, Point-to-multipoint communications, Publish-subscribe, Resilience, Hardware-in-the-loop simulation**

I. Introduction

For a grid modernization technology module such as the fault location, isolation, and service restoration (FLISR) scheme, resilience is a key priority to ensure reliable operation against natural disasters or cyberattacks. FLISR has been proven to reduce the number of customers impacted during a system outage by up to 45% while reducing the customer outage time by up to 51% [1]. But such improvements require an increased communication infrastructure of remote grid systems and must be made resilient to successfully operate when the distribution network is malfunctioning. Modern distribution systems typically utilize a centralized distribution management system (DMS), which operates in a hub-and-spoke logical structure [2]. Such a system allows for a more optimized method of control for the system operator but at the cost of time delays.

ACKNOWLEDGMENT: Contributions to this project were achieved through the Grid Modernization Laboratory Consortium (GMLC), through support from the DOE Office of Electricity Delivery and Energy Reliability, and the Solar Energy Technologies Office. This manuscript has been authored by UT- Battelle, LLC, under contract DE-AC05- 00OR22725 for the U.S. Department of Energy. The United States Government retains and the publisher, by accepting the article for publication, acknowledges that the United States Government retains non-exclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this manuscript, or allow others to do so, for United States Government purposes.

978-1-6654-6618-9/22/$31.00 ©2022 IEEE

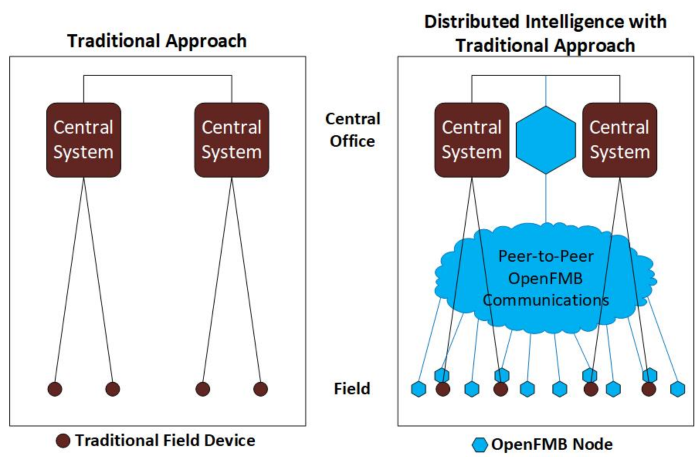

Fig. 1. Traditional vs. Distributed Approaches (Adapted from [3])

One option for improving the communication robustness of any FLISR scheme is to utilize the Open Field Messaging Bus (OpenFMB) architecture, which has the ability to implement distributed peer-to-peer (P2P) control while coordinating with a central controller [3]. Figure 1 shows both a traditional, centralized approach (left) compared with the OpenFMB distributed approach (right) in coordination with a central DMS.

When the central controller is too far away, OpenFMB allows for a quick response to the changing grid conditions. When communication is delayed or lost with the central DMS, OpenFMB enabled devices may provide an additional layer of resilience. Such flexibility is key with any FLISR method, which has not been implemented and validated in the existing literature.

To bridge this gap in the literature, this paper proposes to implement a centrally coordinated FLISR scheme supported by a robust communications network employing the OpenFMB data model to ensure the model is agnostic to the underlying communications protocol and, hence, promote grid interoperability.

The rest of the paper is organized as follows. Section II-A discusses the architecture of the OpenFMB framework and its publish-subscribe model for asynchronous data exchange. Section II-B presents a brief review of the different FLISR implementations currently prevalent. The proposed rules-based FLISR logic is described in Section IV and the validation use case is defined. Section III describes the hardware-in-the-loop (HIL) system used for the framework validation and the OpenFMB communications configuration. The results are presented and discussed in Section V and the concluding remarks are in Section VI.

II. FLISR and OpenFMB

This section introduces and describes the OpenFMB architecture and a comprehensive overview of the different state-of-the-art FLISR implementations.

A. OpenFMB Architecture

The Open Field Message Bus (OpenFMB) framework enables distributed intelligence at the grid's edge. Its platform independent model (PIM) message profiles are used to exchange information in an interoperable manner between devices irrespective of the protocols they use. The PIM builds upon parts of IEC 61850 and the common information model (CIM) which can be converted to protocol buffers (Protobuf) to enable open source, cross-platform data serialization.

The OpenFMB architecture depicted in Fig. 1 uses a publish-subscribe model complete with structured topics that resolve to a device-specific module, a message profile, and a unique device identifier. It provides plug-and-play [2] functionality with devices utilizing DNP3, Modbus, DDS, or MQTT, etc. or through the use of adapters, like Open Energy Solution's OpenFMB Adapter Toolset [4]. Each device can publish its data and subscribe to data that is relevant to its own needs without the need for vendor specific language protocols. Such flexibility allows for dynamic network configurations without scalability limits in addition to resiliency and security.

B. Fault Location, Isolation, and Service Restoration

FLISR is a major component for improving electrical system reliability by minimizing the number of affected customers during fault events or maintenance work. There are three methods for implementing FLISR: local or non-communications-based, distributed or communications-based, and centralized or regional-based [5].

The simplest implementation method of FLISR is performed at the local relay level without communications. This method utilizes internal relay logic and local meter readings to determine if there is a fault, switch to an alternate group setting, and execute sectionalizing. This scheme is typically limited to a radial loop system that utilizes a normally open (N/O) tie-point, must close into the faulted section, and will require extensive settings changes for future topology changes.

Distributed or communications-based sectionalizing needs relay-to-relay communications infrastructure like fiber, radio, or cellular. Communications methods like P2P-based IEC 61850 (GOOSE) [6] or proprietary SEL MIRRORED BITS protocol [7] allow fast sectionalizing through internal relay logic across multiple relays on a feeder. The distributed scheme typically needs a radial loop system with N/O tie-points but will not close into a faulted section. Like the local implementation, extensive settings updates are required with any future system changes.

The most complicated FLISR implementation is based on a regional or centralized control. The regional control can be located at the substation level on a field computer such as an SEL RTAC 3360 or a DMS located at a distribution control center (DCC). This method has the benefits of scalability with no topology limitations and will allow for future FLISR plan maintenance to be easily implemented. This method of FLISR requires high-speed communications like DNP3 which can be a limiting factor depending on equipment used and will likely introduce larger time delays into this sectionalizing method.

OpenFMB as part of a distributed automation system like FLISR, can combine the benefits of centralized control with high-speed distributed peer-to-peer communications at scalable levels along with implementation of future planning changes. Additionally, OpenFMB will allow for interoperability between various communications protocols or to initiate load shedding via microgrids.

III. Testbed Setup

This section describes the HIL test system and the OpenFMB communications interfacing the reclosers and the central coordinating computer.

A. Hardware-in-Loop System

Figure 2 is a high-level diagram for a traditional paired with OpenFMB hybrid communications structure. This testbed is designed to provide backhaul communications to a control center through an SEL-3530 RTAC using DNP3 while having an OpenFMB harness providing the FLISR control operating on local SEL-3360 industrial computer.

The testbed, depicted in Fig. 3, is composed of a Typhoon HIL 604 running a model of the feeder and along with emulating each physical recloser switch and metering system. The Typhoon 604 is connected through a Typhoon HIL Connect and signal breakout boards to send $\pm$10V signals to the low

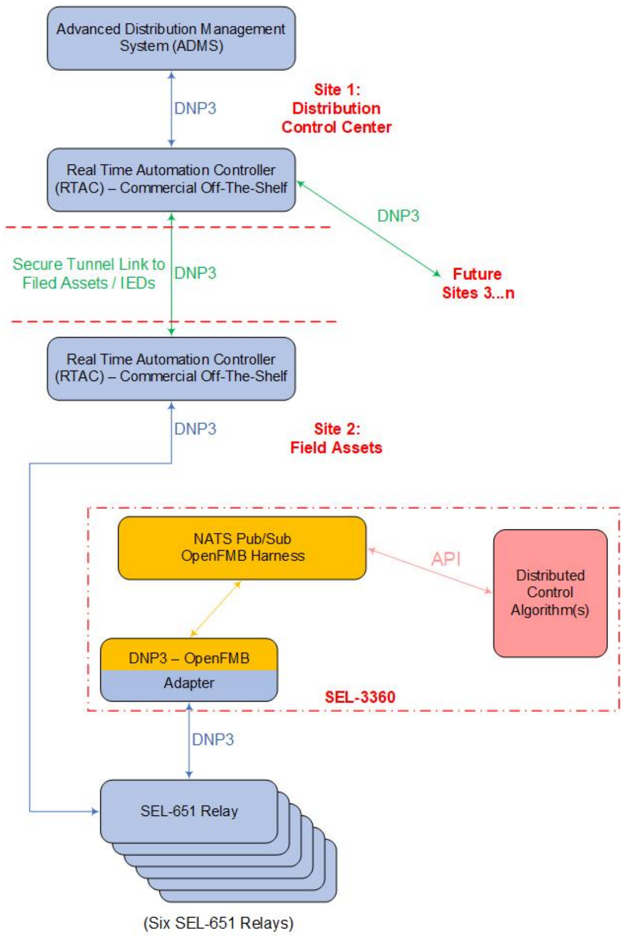

Fig. 2. Diagram of a Multi-Site Implementation of the OpenFMB Testbed (adapted from [8])

energy analog (LEA) inputs of five SEL-651R relays. These signals represent the analogue voltage and current inputs to the relays, the binary outputs from the relay for opening and closing each recloser switch inside the model, and a binary status input to the relay to indicate the state of the recloser switch only. There is no communication aspects within the Typhoon HIL that is utilized by OpenFMB, nor is OpenFMB controlling any part of the model itself. There is also an SEL-3530 Real-Time Automation Controller (RTAC), which is setup for a connection to a remote DMS system [8], and an SEL-3360 industrial computer.

B. OpenFMB Communications Configuration

This work utilizes NATS as the publish-subscribe middleware to facilitate seamless exchange of information among five recloser devices that use DNP3, a legacy protocol. Recloser device modules comprising message profiles for reading (to read the analog values), discrete control (to write changes to settings groups and issue Trip/Close command), and status (to read the 52A3P status bit and the single-phase fault status bits). Each of these profiles are delineated by the OpenFMB harness

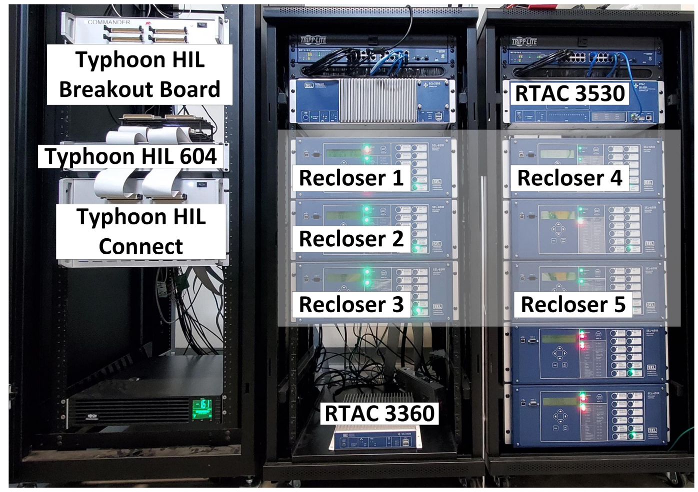

Fig. 3. Testbed setup showing the HIL, SEL reclosers and RTAC

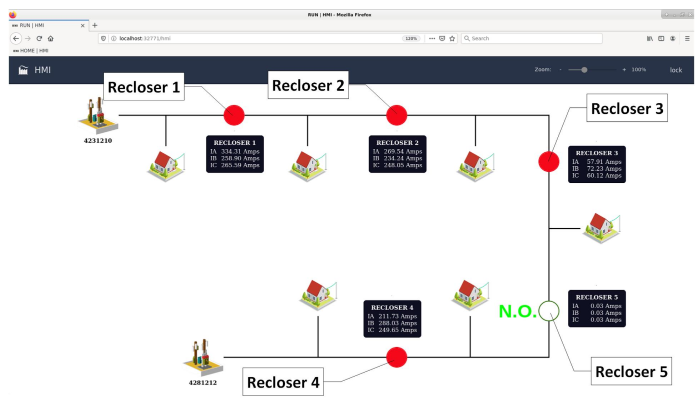

Fig. 4. OpenFMB HMI Screen

based on the unique identifier assigned to each recloser in the network.

The OpenFMB harness comprises a configurable adapter and a programmable human machine interface (HMI), as shown in Fig. 4. Both modules of the harness are containerized and run on a dedicated SEL 3360 Linux-based machine. They are designed to exchange information through the NATS message bus, also containerized. Whereas they are subscribed to the discrete control and status profiles, they publish to the reading message profile. Fig. 6 shows Wireshark capture from a Windows-based machine on the same local network, an independently running script developed in Python uses the Protobuf libraries to interact with the adapter container through the NATS message bus.

The latter functionality of this test setup demonstrates the flexibility of OpenFMB harness and mimics closely a field system where operators can interact with the harness and the NATS bus in a plug-and-play manner once they are in the same local network.

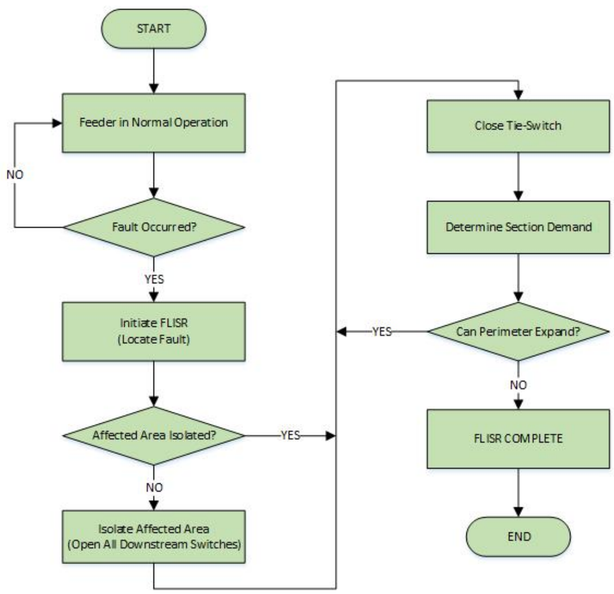

Fig. 5. Typical Rules-Based FLISR Process

IV. Operational FLISR Logic and Test Case

Many methods of system restoration can be distilled down to a manageable course of action, like that in Fig. 5, whereby a restoration process is initiated based on a fault event. Fig. 5 lays out a simple framework for isolating the affected area, reducing the number of impacted customers, and restoring the power efficiently.

A. The Test Case

This project evaluated a simple rules-based FLISR test case and assumed all protective relays operated correctly per their respective settings to isolate the system fault. The restoration process was initiated when a fault forced Recloser 1 to trip open. Due to the fault, Recloser 1 used DNP3 to communicate the presence of a fault, as well as, update the 52A status to OPEN.

When the system running the FLISR script received an indication of a fault and the change in 52A status, the script, communicating through NATS, immediately opened the remaining downstream relays on the feeder as part of a stepped restoration process. Next, utilizing OpenFMB, the script forced a settings group change in Recloser 3 to account for the new feeder operating orientation. Once complete, Recloser 5 was closed, followed by Recloser 3.

V. Results and Discussions

The results of the simple centrally coordinated FLISR test case carried out on the testbed showed a successful integration of OpenFMB into FLISR scheme in combination with a central DMS. The simplicity of deploying containerized OpenFMB harness on existing industrial computers that distribution system operators routinely utilize, along with seamless integration in parallel with the DNP3 communications from a DMS, allows for testing the OpenFMB logic for correct operation and designing and evaluating a plan for future field deployment without the need for vendor specific proprietary protocols.

Fig. 6 shows the packet capture sequence for a specific event in the test case described in Section IV-A. The sequences of interest are highlighted from A through F. The packet capture sequence to the left was taken from the NATS traffic exchanging information between the SEL 3360 computer (IP address 192.168.37.198) that also hosts the OpenFMB harness, the SEL-651R recloser network, and a separate machine where the FLISR scheme runs (IP address 19.2.168.37.10). The sequence to the right was taken from the corresponding NATS traffic facilitating information exchange between the FLISR-running machine and the SEL-3360. With a reference frame identified and marked, the two sequences are locally synchronized in time.

The packet annotated as A represents the CLOSE command triggered by the FLISR scheme at timestamp 0.004732. The receipt of this packet is seen to the left annotated as B on the SEL-3360 machine. Within a millisecond, an acknowledgment is sent back to the FLISR-running machine. The sequences C through E show the communication handshake between the SEL-3360 machine and the Recloser 3 (IP 192.168.37.101). Whereas the communication between the FLISR-machine and the SEL-3360 was purely via the NATS message bus, the exchange between the SEL-3360 and Recloser 3 happens via DNP3. Packet C denotes the transmission of the control message to the recloser, followed by packet D, the corresponding acknowledgment. Packet E denotes the status of Recloser 3 being sent from the device to the SEL-3360 computer, and packet F relays it back to the FLISR-running machine, as read by the control script.

The results demonstrate efficient information exchange using OpenFMB's publish-subscribe model. The asynchronous mode of communication through the NATS message bus provides a level of interoperability across the communication protocols. Though the test case here is limited to devices using DNP3, the authors' ongoing work includes expanding the testbed to include other common protocols such as Modbus, GOOSE, and DDS. Hence, the work presented in this paper acts as the required proof of concept prior to the next steps. Further steps are detailed in the next section.

VI. Conclusion and Future Work

This paper presented a proof of concept for integrating a centrally coordinated rules-based FLISR scheme with the OpenFMB publish-subscribe communications architecture. By leveraging the flexibility in information exchange, interoperability, resilience, and adaptability of this architecture, the first step towards enabling the future vision of distributed FLISR is presented. To validate this integrated approach, a simple HIL testbed paired with a network of five SEL-651R reclosers and a central SEL-3360 computer was used. The results discussed show the efficacy of OpenFMB in facilitating a seamless low-latency information exchange among different reclosers to enable a smooth operation of FLISR.

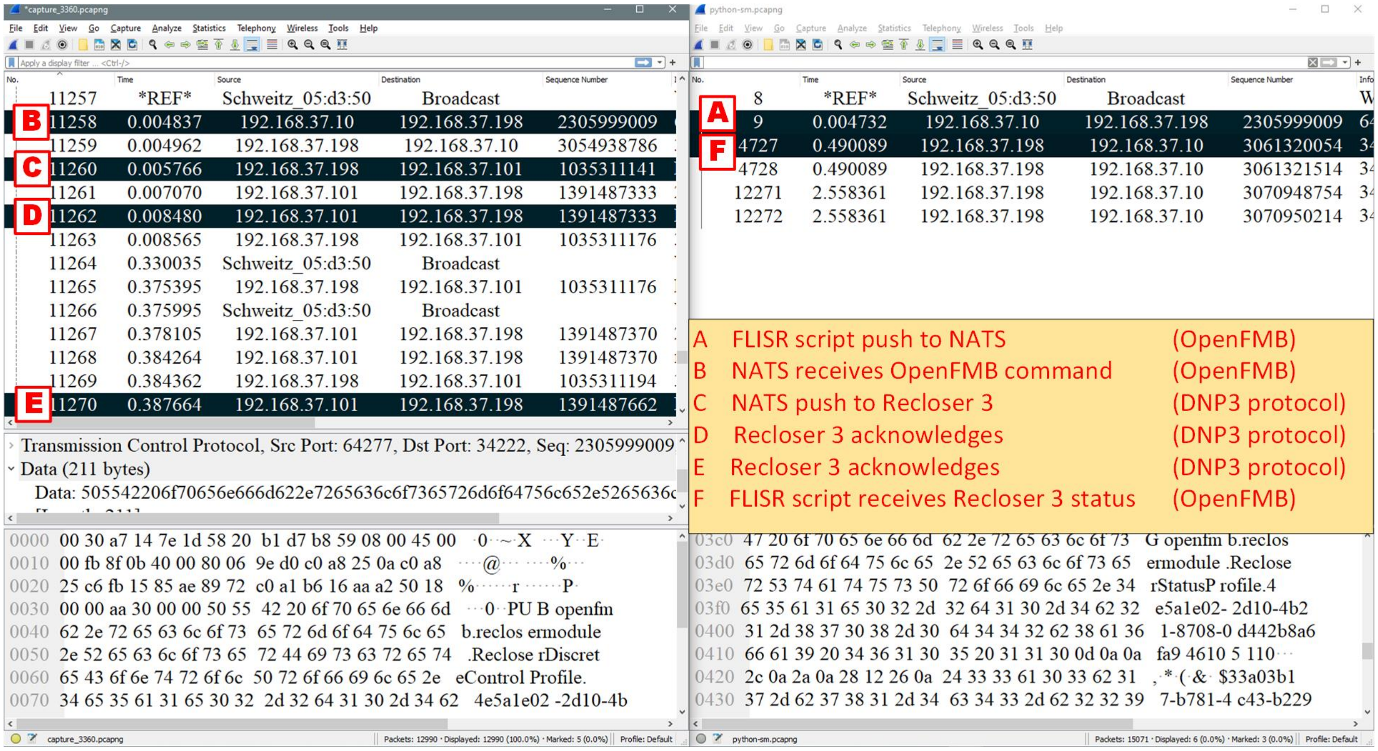

Fig. 6. Wireshark Capture of FLISR Script and NATS Traffic

Future work will look at two additional options for expanding the abilities of OpenFMB as part of a FLISR scheme. The first will be to convert the normally open relay for Modbus to determine the effectiveness of a scheme using multiple protocols and the second will be utilizing meter readings to determine correct relay operation. By adding additional layers of complexity to the FLISR use case, the testbed and OpenFMB are evaluated for functionality that may be encountered on the distribution system. Additionally, future work includes an analysis of the inherent time delays will be evaluated to see what the reduction in delays occur using OpenFMB in a P2P FLISR scheme versus a centralized DMS FLISR scheme.

References

[1] "Fault Location, Isolation, and Service Restoration Technologies Reduce Outage Impact and Duration - Smart Grid Investment Grant Program," Tech. Rep., Electricity Delivery & Energy Reliability, U.S. Department of Energy, Dec 2014.

[2] J. D. Taft, "Comparative architecture analysis: Using laminar structure to unify multiple grid architectures," Pacific Northwest Nat. Lab., Richland, WA, USA, Tech. Rep. PNNL-26089, 2016.

[3] OpenFMB Online Documentation. Available at: https://openfmb.ucaiug.org/ (Accessed Feb 24, 2022).

[4] https://openfmb.openenergysolutions.com/

[5] Liu, Jian Guo, Xinzhou Dong, Xingying Chen, Xiangqian Tong, Xiaoqing Zhang, and Shiming Xu. *Fault Location and Service Restoration for Electrical Distribution Systems.* Hoboken: John Wiley & Sons, Incorporated, 2016. Web.

[6] P. Parikh, I. Voloh and M. Mahony, "Fault location, isolation, and service restoration (FLISR) technique using IEC 61850 GOOSE," 2013 IEEE Power & Energy Society General Meeting, 2013, pp. 1-6, doi: 10.1109/PESMG.2013.6672862.

[7] M. Diehl and A. Dulmage, "Understanding and Effectively Using the SEL MIRRORED BITS Protocol," SEL Application Guide Vol. II, 2018-24

[8] S. Essakiappan, P.R. Chowdhury, K. P. Schneider, S. Laval, K. Prabakar, M.D. Manjrekar, Y.N. Velaga, N. Shepard, J. Hambrick, and B. Ollis. "A Multi-Site Networked Hardware-in-the-Loop Platform for Evaluation of Interoperability and Distributed Intelligence at Grid-Edge." IEEE Open Access Journal of Power and Energy 8 (2021): 460-71. Web.