

A Novel Internet-of-Things Infrastructure to Support Self-Healing Distribution Systems

*Original*

A Novel Internet-of-Things Infrastructure to Support Self-Healing Distribution Systems / Estebasari, A., Orlando, M., Pons, E., Acquaviva, A., Patti, E.. - ELETTRONICO. - (2018), pp. 1-6. (International Conference on Smart Energy Systems and Technologies (SEST 2018) Sevilla, Spain 10-12 September 2018) [10.1109/SEST.2018.8495717].

*Availability:*

This version is available at: 11583/2715493 since: 2020-01-16T17:05:38Z

*Publisher:*

IEEE

*Published*

DOI:10.1109/SEST.2018.8495717

*Terms of use:*

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

*Publisher copyright*

IEEE postprint/Author's Accepted Manuscript

©2018 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collecting works, for resale or lists, or reuse of any copyrighted component of this work in other works.

(Article begins on next page)

# A Novel Internet-of-Things Infrastructure to Support Self-Healing Distribution Systems

Abouzar Estebarsari<sup>‡</sup>, Matteo Orlando, Enrico Pons<sup>‡</sup>, Andrea Acquaviva<sup>§</sup> and Edoardo Patti<sup>\*</sup>

<sup>\*</sup>Dept. of Energy, <sup>‡</sup>Dept. of Control and Computer Engineering

<sup>§</sup>Interuniversity Department of Regional and Urban Studies and Planning  
Politecnico di Torino, Torino, Italy

**Abstract**—In this paper, we present a novel distributed software infrastructure to foster new services in smart grids with particular emphasis on supporting self-healing distribution systems. This infrastructure exploits the rising Internet-of-Things paradigms to build and manage an interoperable peer-to-peer network of our prototype smart meters, also presented in this paper. The proposed three-phase smart meter, called 3-SMA, is a low cost and open-source Internet-connected device that provides features for self-configuration. In addition, it selectively runs on-board-algorithms for smart grid management depending on its deployment on the distribution network. Finally, we present the experimental results of Hardware-In-the-Loop simulations we performed.

## I. INTRODUCTION

Transition from conventional systems with traditional outage management to smart self-healing grids with advanced Outage Management Systems (OMS) is being widely encouraged. This transition aims at improving system reliability and ensure continuity of supply, which eventually reduces outage costs and improves customer satisfaction. In advanced OMS, outage detection is supported by the alarms generated by some network devices (e.g. circuit breakers) and retrieved by Supervisory, Control And Data Acquisition (SCADA) system. This information goes to outage database to support fault diagnosis along with conventional trouble calls. Fault location algorithms may be also applied to use measurements collected by SCADA and find the geographic location of the fault so that crew can be dispatched more efficiently. Integrating smart metering infrastructures enhances performance of advanced OMS by reducing the outage duration and the outage costs [1]. It provides additional information (dashed lines in Figure 1) to OMS such as outage notification (e.g. last gasp messages), outage verification (i.e. reducing number of false or fake reports) and network status for grid reconfiguration-based restoration.

Although integrating smart metering infrastructure per se improves outage management process, but there are still two factors that affect the interruption time: fault location process which depends on communication latency and algorithm execution in distant control centers, and the time needed for dispatching crew to isolate fault. The latter could be accelerated if manual switches are replaced with automatic and/or remote controlled switches, however the efficiency of this scheme, so called Fault Detection, Isolation, Restoration (FDIR), also depends on communication latency and per-

formance of a centralized agent coordinating switchings [2]. Therefore, investigating new approaches addressing the former factor becomes very important.

In this paper, we propose a novel software infrastructure based on Internet-of-Things (IoT) approaches, developing a prototype of a low-cost three-phase smart meter (a.k.a. 3-SMA) to address both aforementioned factors, and eventually improve system reliability by accelerating restoration process. Literature solutions are focused on how to exploit information coming from smart meter to enhance smart grid's performance with particular emphasis on management and on optimizing customers' experience [3]. In this view, the main objectives of future smart meters can be classified in three main categories: i) performing demand response events [4], ii) acting as interface between the smart grid and electric vehicles [5], iii) performing state estimation [6], [7]. Minchala-Avila et al. [4] presented an Internet-connected mono-phase smart meter to monitor and send only power measurements. Qiao et al. [5] discussed an IoT-enabled mono-phase smart meter to perform services for vehicle to grid interactions. Finally, Angioni et al. [7] presented a three-phase meter to monitor distribution grids running also state estimation algorithms. These three meters are low cost and they are ready to be integrated into a smart metering infrastructure (SMI). Regarding commercial Internet-connected three-phase meters, we analysed the following relevant solutions: EMP 7000 [8], G3-PLC [9] and MTR 3000 [10]. All these solutions (both research prototypes and commercial devices) lacks on offering features for self-configuration. Moreover, with the only exception for EMP 7000, they do not provide multi-functions, i.e. they cannot execute different on-board-algorithms to manage the distribution network.

To overcome these limitations, we propose 3-SMA, a three-phase smart meter, that works together with our proposed software infrastructure. The combination of both 3-SMA and software platform support and provide the self-healing distribution systems. Our contribution with respect to literature solutions is described in the following. The software platform implements the rising Internet-of-Things (IoT) paradigms to orchestrate an interoperable peer-to-peer network of 3-SMAs. 3-SMA is a novel low-cost and open-source smart meter that is a self-configurable Internet-connected device. It also implements the IoT paradigms to bidirectionally communicate with other actors in the system. It selectively runs different

on-board-algorithms for smart grid management (e.g. Fault Location and Outage Detection for advanced OMS) according to its deployment along the distribution network and according to configuration settings retrieved remotely via the software platform. Thus, the same 3-SMA can operate differently depending on the portion of the distribution network it monitors. This happens automatically in a plug-and-play fashion, without any manual software or hardware update. 3-SMA samples at 6.4 kHz; however since it is an open source device, the sampling rate can be modified by developers according to new algorithms requirements. Finally, we designed both 3-SMA and the software platform to be compatible with other Advanced Metering Infrastructure, such as [6]. Since this solution is flexible in adding, updating and removing algorithms, its future application scenarios for grid management are different, spanning from state estimation, integration of renewable energy sources and electric vehicles, as depicted in [11].

To highlight our contribution, Table I reports a comparison of our 3-SMA with reviewed literature solutions, pointing out its main features: i) three-phase or mono-phase; ii) IoT-enabled; iii) sampling rate; iv) multi-functions; v) integration with smart metering architecture; vi) self-configuration and vii) low cost. Finally, to demonstrate the performance of 3-SMA, we run a Hardware-In-the-Loop experiment using an Opal-RT digital real-time simulator. A real urban distribution network is modelled to create some fault scenarios and test the device.

To support self-healing process, two layers of systems should be reinforced: distributed networked control devices (e.g. meters and actuators), and communication infrastructure. Deploying our software infrastructure with 3-SMA would allow for more efficient real-time monitoring and measurements to support OMS since OMS in this case does not need to rely only on its own fault detectors and IEDs, but it can also exploit low cost widely distributed 3-SMA. Moreover, not all real-time data is required to be communicated with a central agent for calculations, but 3-SMA itself runs some algorithms to provide some post-proceed information (e.g. exact location of fault) to the rest of OMS. The proposed infrastructure can also support a more efficient fault isolation and restoration thanks to the orchestration of an interoperable peer-to-peer network, in which one of 3-SMAs (we set the one at the primary substation, at beginning of MV feeder, or the one at the secondary substation at the beginning of an LV feeder) becomes an agent to coordinate status of switches along the feeder. Once 3-SMA has collected fault measurements and has calculated the location of the fault, it sends commands to the nearest switches to the faulted branch to open them and isolate the fault. Once the fault has been isolated, 3-SMA sends commands to circuit breaker of the feeder, as well as available normally open switches for reconfiguration, to eventually restore the rest of the feeder.

The rest of this paper is organized as follows. Section II introduces the proposed three-phase smart meter prototype and the IoT software infrastructure to manage the resulting

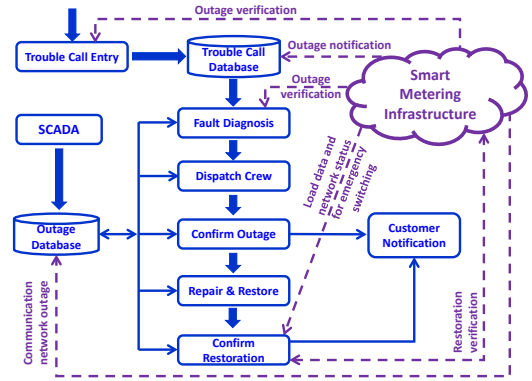


Fig. 1. Integration of Smart Metering infrastructure to OMS.

TABLE I  
COMPARISON WITH LITERATURE SOLUTIONS

Meter	3-phase	IoT enabled	Sampling rate	Multi-functions	SMI integration	Self-configurable	Low-cost
Our Solution	✓	✓	6.4 kHz	✓	✓	✓	✓
Minchala-Avila et al. [4]	x	✓	0.017 Hz	x	✓	x	✓
L. Qiao et al. [5]	x	✓	n.a.	x	✓	x	✓
Angioni et al. [7]	✓	✓	10 kHz	x	x	x	✓
EMP 7000 [8]	✓	✓	25.6 kHz	✓	x	x	x
G3-PLC [9]	✓	✓	n.a.	x	✓	x	x
MTR 3000 [10]	✓	✓	n.a.	x	x	x	✓

peer-to-peer network. Section III presents both case study and performed laboratory tests. Finally, Section IV discusses concluding remarks.

## II. SMART METER PROTOTYPE AND SOFTWARE INFRASTRUCTURE

In this section, we present our prototype of a three-phase smart meter and the software infrastructure to create an interoperable peer-to-peer 3-SMA network (see Figure 2). 3-SMA is a self-configurable device that can selectively run different on-board-algorithms for smart grid management (e.g. Fault Location and Outage Detection for advanced OMS). It exploits the Internet-of-Things communication models to send sampled measurements and/or post-processed information to the other actors in our infrastructure over the Internet. Thus 3-SMA is a fully connected IoT device. Thanks to the adopted IoT communication paradigms, our solution can be integrated in more complex smart metering infrastructures (such as [12] and [6]), thus enabling additional novel services and applications.

As shown in Figure 2, the main actors in our software infrastructure are two *Message Brokers*, the *Device Catalog* (DeC), the proposed *Three-phase Smart Meters* (i.e. 3-SMA) and the *Services* for smart grid management. The overall infrastructure exploits the two main communication paradigms: (i) request/response through REST Web Services [13] and (ii) publish/subscribe [14] via MQTT protocol [15]. Request/response enables synchronous communication. Whilst, publish/subscribe allows asynchronous communication that enables the development of scalable loosely-coupled event-based systems, services and applications that can react in (near-) real-time to certain events.

The *Message Broker* manages the publish/subscribe com-

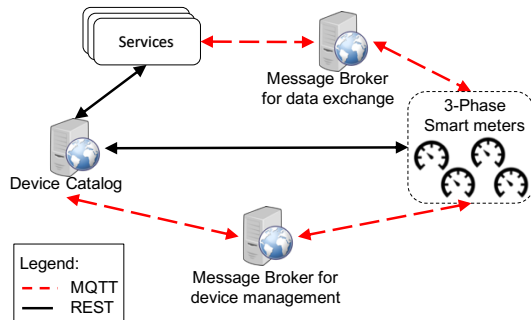


Fig. 2. Scheme of the proposed software infrastructure.

munication among the actors in the infrastructure, either hardware or software. As shown in Figure 2, we use two Message Brokers: the former exchanges information about the status of the smart grid among 3-SMAs and Services; the latter exchanges information about device management between DeC and 3-SMAs.

The *Device Catalog* is a supervising application and it is the core of the proposed software infrastructure. On one hand, it makes available an always updated list (i.e. the catalog) of 3-SMAs really connected to the Internet and their end-points (i.e. IP addresses, REST web services and MQTT topics). In this view, DeC implements a feature to remove from the catalog all the 3-SMAs that cannot communicate any-more with the rest of the infrastructure (e.g. lack on the Internet connection or damaged device). For each 3-SMA, DeC provides also additional meta-data about: (i) which portion of the grid is monitoring (i.e. feeder or lateral) and (i) what on-board-algorithms are running on. This information is also needed by Services to set-up their management engine and by other 3-SMAs to enable a bi-directional and peer-to-peer communication. On the other hand, DeC provides 3-SMAs with configuration settings: (i) list of Message Brokers and their end-points to be used, (ii) information about on-board-algorithms to be executed and (iii) additional information on the specific portion of the grid to be monitored. Hence, DeC is a crucial actor that provides flexibility and scalability to the overall infrastructure. To make available such information and to constantly update the catalog of available devices, DeC provides REST Web Services and exploits the MQTT protocol.

The proposed 3-SMA prototype exploits both communication approaches: (i) request/response is used to periodically retrieve configuration settings from DeC; (ii) publish/subscribe is needed to exchange information about the status of the grid in (near-) real-time and to periodically send control messages to keep the *Device Catalog* updated. 3-SMA consists of an analog to digital converter, namely MCC-USB-201, connected to a Raspberry Pi 3 model B through the USB port. The MCC-USB-201 is equipped with three voltmeters and three amperometers connected to its analog inputs. Measurements can be sampled with a sample rate of 6.4 kHz; however in a laboratory test that we performed on the model of a real distribution system, due to the computation constraint of each

simulator's core dedicated to simulation of each feeder, a time-step of 250 microseconds (i.e. 4 kHz) is set to avoid real-time simulation overruns. The signal generated by the simulator remains constant during a time-step, so it is not a true but digitized sinusoid wave. Thus, in order not to exceed the refresh rate, the sampling rate of the smart meter is set to 3.2kHz (64 samples per cycle). Regarding the fault location algorithm, it should be noted that 32 samples per cycle during pre-fault and 32 samples during post-fault measurements are sufficient, therefore this sampling rate of the meter greatly fits.

Sampled measurements are then sent to the Raspberry Pi via USB interface. The Raspberry Pi is the computational core of 3-SMA. It manages the Internet connection and the routines for self-configuration. It also hosts the on-board-algorithms for smart grid management (e.g. Fault Location and Outage Detection for advanced OMS) that are then executed according to the configuration settings retrieved by DeC. The outputs of the running algorithms are sent to the other actors (i.e. services or other 3-SMA) in our infrastructure via the Message Broker for data exchange. More details about the implemented fault location algorithm running on 3-SMA are given in our previous work [16].

Finally, the *Services* are the algorithms for smart grid management. As shown in Figure 2, they can exploit either REST or MQTT protocols (i) to retrieve the updated catalog of alive 3-SMAs and their end-points and (ii) to exchange data about the status of the grid, respectively. To support an advanced OMS, there are some Services which can exploit our scheme to perform more efficiently. These Services mainly intend to promptly isolate the faulty part of the grid and restore the healthy part in order to reduce interruption time and number of interrupted customers. For example, FDIR coordination program is one of these Services; as soon as it received outage detection alarm and fault location message from meters, it sends commands to closest switches to fault to open them and isolate the fault portion. Then, it sends commands to reclose circuit breaker of the feeder as well as closing a Normally Open Switch (NOP) which reconfigure the grid to quickly re-energize the rest of healthy line and laterals. As shown in Figure 1, some other Services could also benefit from this scheme, such as outage verification and outage mapping (i.e. outage location), event analysis and recording (e.g. customer notification or reliability assessment), supply restoration of faulty part through route optimization for repair crew.

### III. LABORATORY TESTS

In this section, we present some results of running an impedance-based fault location algorithm [16] embedded in the proposed 3-SMA. A portion of a real distribution system in Turin city in Northwest of Italy is considered as the case study. This portion of the grid is in medium voltage (22 kV), and it consists of 11 MV feeders derived from a primary substation. The fault location algorithm used for demonstration purpose requires pre- and post-fault measurements at the beginning of MV feeder. So for each feeder there should be at least

one 3-SMA meter installed at the primary substation. In order to test the functionality and performance of our meter, we connected it to the second feeder from left, indicated with red solid color in Figure 3. We trigger a single-phase to ground fault on different branches (line sections) along this feeder which are always unknown to the algorithm.

#### A. Grid Modelling

We used an OPAL-RT real-time simulator to perform an electromagnetic transient analysis (EMT). This required a specific configuration of simulation solver named eMEGAsim whose development environment is MATLAB Simulink. Therefore, the model of the distribution system is implemented in MATLAB Simulink. An interfacing software named RT-LAB compiles Simulink model and load it to the digital simulator to be executed in real-time with 250 microsecond fixed time-step. This grid has 415 secondary substations (i.e. buses) with 42 MV loads and 220 LV aggregated loads, and 403 MV branches. Implementing such a large system in Simulink with conventional manual drag-and-drop approach using libraries would be very time-consuming and less efficient. To overcome this problem, we developed a MATLAB script, which can interpret a network description and automate the creation of the network (Figure 4).

The Simulink working sheet can be considered as a Cartesian plane that has the values of the ordinates increasing to the right, and the values of the abscissa increasing downwards. Within this sheet, two pairs of coordinates identified the exact position of each block. More precisely, they indicate the upper left corner's position and the lower right corner's position of the blocks. Instead of manually entering the blocks from the library, Simulink components are objects that can be manipulated programmatically from MATLAB. The basic idea of the algorithm is to insert firstly the blocks, which represent the network nodes, in order to create the structure (the "skeleton") of the network itself. For this reason, the first operation to perform is to obtain preliminarily the necessary coordinates for these blocks. Then, using the coordinates of the nodes as references, all the other components (lines, loads, etc.) are also collocated in the model. The network information have been written in an Excel file. Regard to the information about the connected nodes, it should be noted that not all nodes are defined as upstream node and downstream node. In other words, the information is not always given in the format "from - to", but for some couples of nodes the direction of the link is "to - from" type. Our developed program is flexible with respect to the network data tables and input information.

To test the performance of 3-SMA, we run Hardware-In-the-Loop simulations. We connected a 3-SMA meter to the OPAL-RT real-time simulator via the analog output provided by the OPAL-RT itself, as shown in Figure 5. Thus, this 3-SMA meter monitors the feeder indicated by a solid red line in Figure 3 (the second derived line from the left of the MV network). We also deployed the *Device Catalog* in a computer to provide with 3-SMA configuration settings to set-up both the communication over the Internet and the fault

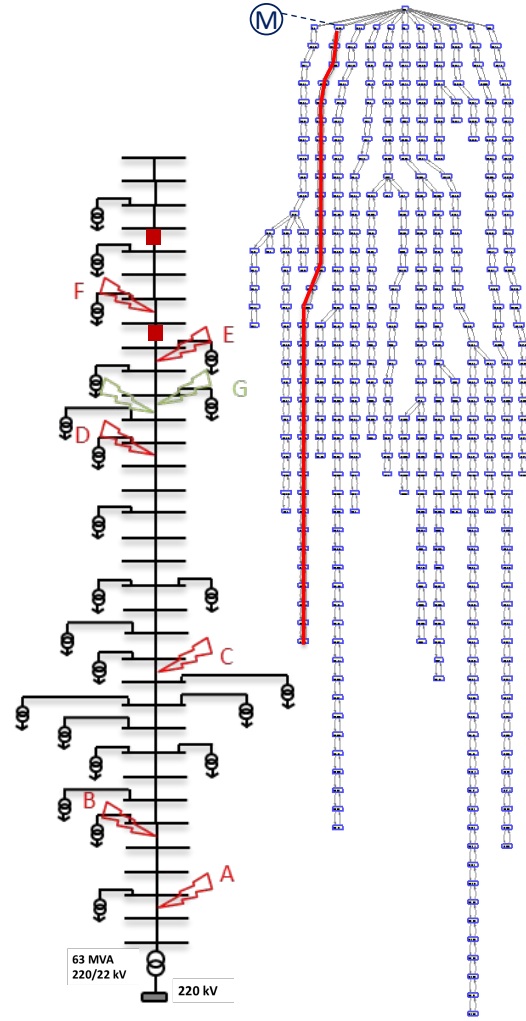


Fig. 3. Topology of the distribution grid with 11 MV feeders and locations of triggered faults along the selected feeder.

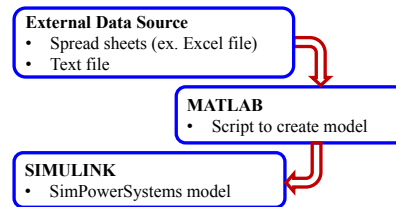


Fig. 4. Process of implementing the grid model.

location algorithm, as described in Section II. The *Device Catalog* gives also additional meta-data about the feeder to monitor. Then, following the scheme depicted in Figure 2, 3-SMA starts i) exchanging information about the status of the grid and ii) sending periodically control messages to keep the *Device Catalog* updated.

#### B. Fault Location Experimental Results

The fault location algorithm firstly calculates phasor values of pre- and post-fault measurements, then the phasors are

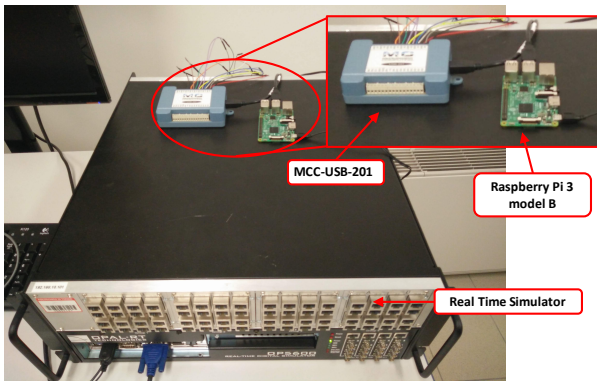


Fig. 5. Hardware-In-the-Loop test bed.

used during the impedance-based fault location process as well as forecasted load data, the system topology and line data from its database. The result of this process contain information of the location of the fault in terms of distance from nearest upstream secondary substation, name of the two nearest substations which are connected through faulty cable, and the fault impedance.

Typically, for an accurate phasor conversion calculated by Discrete Fourier Transform (DFT), the minimum number of samples required is 32 samples per one cycle before fault occurrence and 32 samples per one cycle after fault occurrence and before protection. However, the use of a greater number of samples makes the phasor conversion more reliable and affected by less uncertainty. Hence, an observation window of 1 second including instantaneous values of current and voltage waveforms with 3.2 kHz sampling is captured by the meter to pass to the algorithm (Figure 6).

The simulated fault occurrences were characterized by values of resistance ( $R_{on}$ ) equal to  $0.001 \Omega$  and ground resistance ( $R_g$ ) equal to  $0.001 \Omega$ . We triggered this fault on 8 points of the feeder independently (Cases A to G). In test cases of fault scenario from A to F, the faults were triggered in the middle of faulty sections, while the fault G occurs not in the middle of a section, but along its different other points (one fault was triggered at the 20 percent of its length and another one at the 80 percent of its length). Figure 6 shows the received voltage and current waveforms for case F.

However, for the characterization of the fault point location it was decided to use the distance of the fault point from the beginning of the line. This choice is due to the fact that for distribution system operators this kind of information is more practical. The results obtained by the fault location algorithm have been reported in Table II.

$$\Delta e \% = \frac{\Delta e}{X_s} \cdot 100 = \frac{|X_0 - X_s|}{X_s} \cdot 100 \quad (1)$$

Where  $X_s$  [km] indicates the distance of the fault point, triggered during the real-time simulation from the hypothetical main relaying point (at the beginning of the feeder);  $X_0$  [km] indicates the distance of the fault point, located by the fault

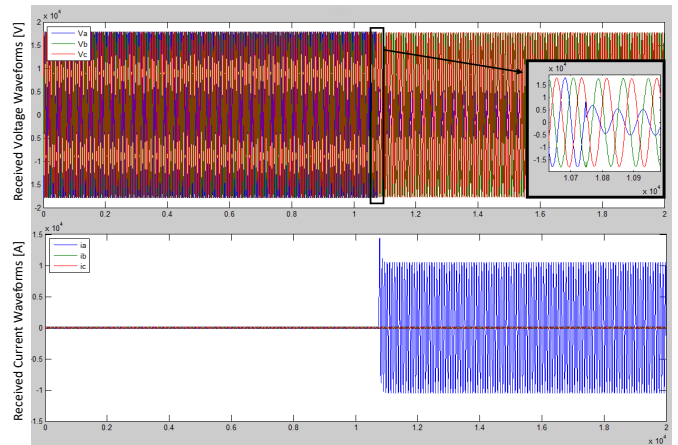


Fig. 6. Captured voltage and current waveforms during fault occurrence.

TABLE II  
RESULTS OF FAULT LOCATOR FOR DIFFERENT CASES

Fault Case	Section length [Km]	$X_s$ [Km]	$X_0$ [Km]	$ \Delta e $	$\Delta e \%$
A	0.115	1.1133	1.0851	0.0282	2.53 %
B	0.1123	1.48575	1.4584	0.02735	1.84 %
C	0.173	2.6084	2.5662	0.0422	1.61 %
D	0.2095	5.16665	5.1154	0.05125	0.99 %
E	0.152	5.9977	5.9608	0.0369	0.61 %
F	0.1347	6.15305	6.1204	0.03265	0.53 %
G	0.2389	5.54868	5.5116	0.03708	0.67 %
		5.69202	5.6549	0.03712	0.65 %

locator, from the hypothetical main relaying point (at the beginning of the feeder);  $|\Delta e|$  indicates the absolute error between  $X_s$  and  $X_0$ ; and finally  $\Delta e \%$  indicates the percentage error. Figure 7 shows the trend of  $\Delta e$  percentage compared to the distance expressed by  $X_s$ .

The algorithm running on the 3-SMA meter has always localized correctly the nodes of the faulty section and the distance of the fault point has been given with a reasonable error. Furthermore, as regards the impedance-based fault location method used in this work, the analogue to digital conversion

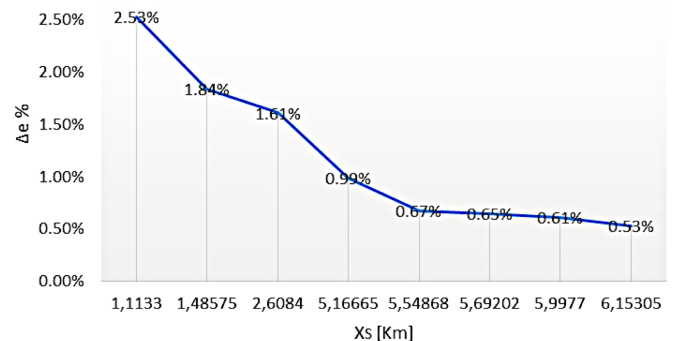


Fig. 7. Chart of e percentage in function of  $X_s$ .

of the measured signals in phasor calculation contributes to increase this small and unavoidable margin of error. That is, the process of sampling the continuous signals obtained by the meters of voltage and current and the process of their conversion into phasor values affects the accuracy of the responses. Anyway, the studies carried in literature about the correct use of fault location algorithms agree in considering that a result is reasonably acceptable if it is affected by a margin of error respect the real fault position less than 100-200 meters. The maximum value obtained for  $\Delta e$  during our tests is 51.25 meters.

Finding the exact location of the fault is important for the repair crew to access the point efficiently and quickly, however prior to any crew dispatching for fixing the issue, a self-healing system should be able to isolate the faulted portion of the grid from the rest, and restore the healthy part of the feeder. If automation agent is informed of the location of the fault, it can shortly send commands to the nearest switches to isolate the branch.

In our case study, for the case F, we ran the simulation with both fault location and FDIR algorithm. In this case, FDIR does not look for detecting the fault, yet it awaits the result of fault location algorithm. The result includes the name of the upstream and downstream secondary substations, the distance of fault, and the impedance of fault. FDIR looks into its data base to find the two nearest switches before and after those two substations detected by fault location (shown in Figure 3). It subsequently sends opening commands to them, for fault isolation, followed by another command to the circuit breaker of the feeder to close it back (i.e. restoration). This grid has no normally open switches for reconfiguration.

#### IV. CONCLUSION

In this paper, we introduced a software infrastructure to foster novel services for advanced outage management in smart grids. It exploits the Internet-of-Things paradigms to build and manage an interoperable peer-to-peer network of smart meters, called 3-SMA. In the paper, we also presented this novel smart meter which is an IoT device, open-source and low-cost. 3-SMA is self-configurable and selectively run different on-board-algorithms for smart grid management, such as Fault Location and Outage Detection for advanced OMS, according to its deployment along the distribution network.

Finally, to demonstrate the performance of the proposed solution, we performed Hardware-In-the-Loop simulations by connecting a 3-SMA meter to an OPAL-RT real-time simulator where a model of a portion of a real distribution system was running. Captured pre- and post-fault measurements including voltage and current samples are fed into the embedded algorithm to locate the fault. A single-phase to ground fault was triggered in several locations of a selected feeder, and the error percentage between the location of the triggered fault and the location found by the algorithm was calculated as a performance indicator of the meter in sense of fault location. The results showed the maximum error was 51.25

meters, which is much below usual acceptable range of 100-200 meters. To test a full-cycle self-healing process which also includes fault isolation and restoration, we enabled FDIR algorithm of the meter to communicate with the nearest switches to the fault, after Fault Location algorithm has located it.

As a future work, we would like to embed a state estimation algorithm into the 3-SMA meter to support fault location algorithm with more accurate estimation of network operational status, which eventually improves self-healing process. The idea of exploiting such a distributed software infrastructure is not limited to only support self-healing algorithms, but also foster any new services in smart grids in a decentralized way with less dependency to a central control agent like SCADA of distribution management system (DMS).

#### REFERENCES

- [1] J. B. Ekanayake, N. Jenkins, K. Liyanage, J. Wu, and A. Yokoyama, *Smart grid: technology and applications*. John Wiley & Sons, 2012.
- [2] A. Zidan, M. Khairalla, A. M. Abdrabou, T. Khalifa, K. Shaban, A. Abdrabou, R. E. Shatshat, and A. M. Gaouda, "Fault detection, isolation, and service restoration in distribution systems: State-of-the-art and future trends," *IEEE TSG*, vol. 8, no. 5, pp. 2170–2185, 2017.
- [3] S. S. S. R. Depuru, L. Wang, and V. Devabhaktuni, "Smart meters for power grid: Challenges, issues, advantages and status," *Renewable and sustainable energy reviews*, vol. 15, no. 6, pp. 2736–2742, 2011.
- [4] L. I. Minchala-Avila, J. Armijos, D. Pesántez, and Y. Zhang, "Design and implementation of a smart meter with demand response capabilities," *Energy Procedia*, vol. 103, pp. 195–200, 2016.
- [5] L. Qiao, X. Liu, and B. Jiang, "Design and implementation of the smart meter in vehicle-to-grid," in *Electric Utility Deregulation and Restructuring and Power Technologies (DRPT), 2011 4th International Conference on*. IEEE, 2011, pp. 618–621.
- [6] M. Pau, E. Patti, L. Barbierato, A. Estebsari, E. Pons, F. Ponci, and A. Monti, "Low voltage system state estimation based on smart metering infrastructure," in *Applied Measurements for Power Systems (AMPS), 2016 IEEE International Workshop on*. IEEE, 2016, pp. 1–6.
- [7] A. Angioni, G. Lipari, M. Pau, F. Ponci, and A. Monti, "A low cost pmu to monitor distribution grids," in *Applied Measurements for Power Systems (AMPS), 2017 IEEE International Workshop on*. IEEE, 2017, pp. 1–6.
- [8] EMP 7000. [Online]. Available: <http://www.gegridsolutions.com/multilin/catalog/epm7000.htm>
- [9] G3-PLC. [Online]. Available: <https://www.devalo.com/products/Smart/devolo-G3-PLC-Modem-500k/data/Product-sheet-devolo-G3-PLC-Modem-500k-com.pdf>
- [10] MTR 3000. [Online]. Available: <https://www.networkedenergy.com/en/resources/datasheets-networked-energy-services>
- [11] M. H. Amini, K. G. Boroojeni, S. S. Iyengar, P. M. Pardalos, F. Blaabjerg, and A. M. Madni, *Sustainable Interdependent Networks: From Theory to Application*. Springer, 2018, vol. 145.
- [12] M. Pau, E. Patti, L. Barbierato, A. Estebsari, E. Pons, F. Ponci, and A. Monti, "A cloud-based smart metering infrastructure for distribution grid services and automation," *Sustainable Energy, Grids and Networks*, 2017.
- [13] R. T. Fielding and R. N. Taylor, "Principled design of the modern web architecture," *ACM Transactions on Internet Technology (TOIT)*, vol. 2, no. 2, pp. 115–150, 2002.
- [14] P. T. Eugster, P. A. Felber, R. Guerraoui, and A.-M. Kermarrec, "The many faces of publish/subscribe," *ACM computing surveys (CSUR)*, vol. 35, no. 2, pp. 114–131, 2003.
- [15] Message Queue Telemetry Transport (MQTT). [Online]. Available: <http://mqtt.org/>
- [16] A. Bahmanyar, S. Jamali, A. Estebsari, and E. Bompard, "A comparison framework for distribution system outage and fault location methods," *Electric Power Systems Research*, vol. 145, pp. 19–34, 2017.