POLITECNICO DI TORINO
Repository ISTITUZIONALE

A Distributed Multimodel Cosimulation Platform to Assess General Purpose Services in Smart Grids

Terms of use:

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

(Article begins on next page)

13 May 2024

A Distributed Multi-Model Co-simulation Platform to Assess General Purpose Services in Smart Grids

Luca Barbierato, *Student Member, IEEE,* Abouzar Estebsari, *Member, IEEE,* Lorenzo Bottaccioli, *Member, IEEE,* Enrico Macii, *Fellow, IEEE* and Edoardo Patti, *Member, IEEE*

Abstract—Future smart grids with more distributed generation and flexible demand require well-verified control and management services. This paper presents a distributed multi-model co-simulation platform based on Smart Grid Architecture Model (a.k.a. SGAM) to foster general purpose services in smart grids. It aims at providing developers with support to easily set-up a test-bed environment where they can simulate realistic scenarios to assess their algorithms and services. The proposed platform takes advantages of Internet-of-Things communication paradigms and protocols to enable the interoperability among different models and virtual or physical devices that compose a use case. Moreover, the integration of digital real-time simulators unlocks Hardware-In-the-Loop features. To test the functionality of our platform, a novel scheme of fault detection, isolation and restoration is developed, in which communication and interoperability of different functions and devices are crucial. This service is applied on a realistic portion of a power grid in Turin, Italy, where devices communicate over the Internet. Finally, the laboratory experimental results achieved during a real-time co-simulation are discussed.

Index Terms—Smart Grid; SGAM; Distribution Systems; Multi-Model Co-Simulation; Co-simulation Platform; Outage Management

I. INTRODUCTION

Nowadays, we are attending to the transition from a hierarchical electrical power grid with big bulk plants to a smart grid characterised by distributed generation, flexible demand and distributed control policies. Such a new system has to be self-healing and resilient to faults. Thus, new general purpose services need to be developed. For this purpose, the Smart Grid Architecture Model (SGAM) offers a support to design them with an architectural approach by which interoperability viewpoints can be well represented [1].

Similar to any other ex-ante analysis, laboratory facilities and simulation systems are needed to test and verify performance of new methods or tools for different use cases. As a crucial requirement for test and validation of new solutions, a near real-world environment is recommended to be used [2]. In this regard, real-time simulations (RTS) depict a path that can effectively support the research efforts to meet emerging lab test requirements [3], [4].

L. Barbierato, L. Bottaccioli and E. Patti are with the Dept. of Control and Computer Engineering and the Energy Center Lab, Politecnico di Torino, Italy. (e-mail: luca.barbierato, lorenzo.bottaccioli, edoardo.patti}@polito.it)

A. Estebsari is with the School of the Built Environment and Architecture, London South Bank University, London, United Kingdom. (e-mail: estebsaa@lsbu.ac.uk)

E. Macii is with the Interuniversity Dept. of Regional and Urban Studies and Planning, Politecnico di Torino, Italy. (e-mail: enrico.macii@polito.it)

For power system electromagnetic transient analysis (EMT), like fault analysis, RTSs are very reliable with flexibility and scalability due to computation parallelism features. However, to study interoperability in smart grids based on SGAM, combination of control and communication simulators with grid real-time simulator is needed. An overview of existing co-simulation frameworks is discussed in [5]. Some of these frameworks do not run simulations in real-time, some are not scalable in terms of number of interconnected devices (e.g. smart meters, relays, actuators, etc.) and some are quite costly in terms of lab set-up.

On these premises, this paper, which extends our previous work [6], we proposes a distributed multi-model co-simulation platform that takes advantages from Internet-of-Things (IoT) communication paradigms and protocols to allow the interoperability across different models [7] and many virtual or physical devices [8] (i.e. circuit breakers, reclosers, remote terminal units, smart meters, etc.) composing the simulation scenario. It also integrates Digital Real-Time Simulators (DRTS) to enable Hardware-In-the-Loop (HIL) simulations. HIL is very beneficial for testing and evaluating the performance of physical prototypes or newly developed algorithms as embedded software, before deploying them in real-world systems. The proposed platform has been designed to simulate, even concurrently, various general purpose services for smart gird management following an event-driven approach. Nevertheless, a particular emphasis is given on addressing fast transient phenomena in smart grids, which require fast system response, like Fault Detection, Isolation and Restoration (FDIR).

Comparing with most of existing co-simulation platforms, the one proposed in this paper is able to concurrently meet the testing requirements of use cases with the following characteristics: i) when many ICT-enabled devices (i.e. smart meters, actuators, bay controllers, etc.) are distributed widely in the network and coordination of them through communication network is important; ii) when smart functions or management schemes deal with fast transient phenomena; iii) when communication latency would directly impact the performance of an application.

In its core, the platform integrates different network simulators for data transmission. Thus, realistic Metropolitan Area Networks (MANs) can be modelled to evaluate the impact of new IoT devices and services in communicating over the Internet in terms of congestion and messages delays. Indeed, such delays in data transmission can impact on the operational status of the smart grid [9]. We believe our

platform can provide valuable supports with minimum amount of required efforts to set-up the test-bed environment and to migrate from a fully simulated environment to the real-word.

The rest of the paper is organised as follows. Section II reviews relevant literature solutions highlighting our contribution. Section III presents the proposed distributed multi-model co-simulation platform. Section IV introduces our use case for fault detection, isolation and restoration. Section V describes the case study and its test-bed setup. Section VI presents the experimental results. Finally, Section VII discusses the concluding remarks.

II. RELATED WORKS

This section reviews literature solutions on co-simulation platforms for Smart Grids that are complex networks of systems where different entities cooperate by exchanging heterogeneous information.

In [10], Hopkinson et al. present a co-simulation environment, called EPOCHS, that is based on High Level Architecture (HLA) standard [20]. EPOCHS is a breakthrough technology, since it is the first known co-simulator for realising simulations of power systems and communication networks. To evaluate electromagnetic scenarios involving communication networks, EPOCHS integrates i) ns-2 [21], a communication network simulator; ii) PSCAD/EMTDC, a commercial electromagnetic transient simulator; and iii) PSLF, a commercial electro-mechanical transient simulator. INSPIRE [11] is another co-simulation platform based on HLA technology. It interconnects a power system simulator, called DIgSILENT PowerFactory, with a communication network simulator, called OPNET [22], and continuous time-based application modelled in MATLAB, JAVA, GNU R and C++. It is based on both the IEC 61850 standard and the IEC 61968/61970 common information model. Thus, INSPIRE is limited on simulating the interactions between both *Components* and *Communication Layers* in SGAM. Shum et al. [12] presented a more complex ad-hoc co-simulation platform based on HLA and JADE, a popular multi-agent platform. The platform integrates OPNET with the commercial electromagnetic transient simulator PSCAD/EMTDC. The framework aims at easing to understand, evaluate and debug distributed smart grid software.

Compared to [10], [11], [12], GECO [13] exploits a different approach to create a co-simulation environment between PSLF and ns-2. In GECO, co-simulation is achieved through a global event-driven mechanism to manage the co-simulation orchestration instead of HLA technology. Yang et al. [14] presented a co-simulation framework for validating distributed controls in Smart Grids, based on both Hardware- and Software-In-the-Loop approaches. Controls are designed exploiting a model-driven approach based on the IEC 61499 standard. The communication among power plants and the controllers, that are developed in MATLAB/Simulink, is achieved through UDP (User Datagram Protocol) and TCP (Transmission Control Protocol) sockets. Mosaik [23] is a framework for modular simulation of active components in Smart Grids that also follows an event-driven approach. Mosaik has been used by Nguyen et al. [15] to integrate DRTS (i.e. RTDS) with Omnet++ (a communication network simulator) and perform Power Hardware-in-the-Loop simulations.

For analysing the performance of Smart Grid technology, Bian et al. [16] presented a real-time co-simulation platform that integrates Opal-RT and OPNET. The platform provides the possibility to interface the OPNET simulation environment with the real world through Internet Protocol (IP). Venkataramanan et al. [17] developed an ad-hoc co-simulation platform that can be used as test-bed for microgrid cyber-physical analysis. It combines: i) an RTDS, ii) the Common Open Research Emulator (a.k.a. CORE) and iii) Open Platform Communication based on FreeOPCUA. OOCoSim [18] is another co-simulation framework that integrates both OPNET and OpenDSS (a power system simulator). OOCoSim framework has been applied to study only demand response application in a smart-grid scenario. Finally, Mirz et al. [19] proposed a co-simulation architecture for power system communication and market analysis in smart grids. Authors proposed a data-model to integrate and allow the communication among i) market models in python, ii) power system simulations in Modelica and iii) network models in ns-3.

The presented literature solutions, with the exception of [19], do not follow SGAM that supports the design of services highlighting the interoperability among the actors in the smart grid. Moreover, [10], [13], [17] are designed and implemented for a specific use-case scenario with particular requirements. Whilst, these kind of co-simulation platforms must be designed to be flexible as much as possible to simulate general purpose services, even concurrently, in smart grids. Thus, a multi-model approach is needed to allow a cooperation among different simulation models, possibly distributed over the Internet and communicating by exploiting standard protocols and data-formats. This make easier the definition of more complex smart grids scenarios. Finally, literature solutions lack on the integration of IoT devices and third-party platforms that can feed the algorithm to be tested with real-world data, even in (near-) real-time. Furthermore, the integration of IoT communication paradigms and protocols enable the required flexibility and scalability in performing HIL co-simulations. Thus, a platform that integrates IoT protocols can be ready to integrate next generation devices (e.g. real-world smart meters).

This paper proposes a distributed multi-model co-simulations platform to asses and evaluate general purpose services in Smart Grids by implementing an event-driven approach. It exploits communication paradigms peculiar of IoT platforms, to implement a flexible framework where different power grid scenarios and simulations can be executed. In its core, it implements i) different software simulation environments, ii) Digital Real-Time Simulators, and iii) three different communication network simulators (to be used alternately according to co-simulation requirements). Thanks to the integration of DRTS, the platform is also ready to perform HIL simulations. Whilst by using IoT protocols, it allows the interoperability with real-world IoT devices and

	Co-Simulation Purpose	SGAM Compliant	Multi model				IoT
			Network Simulators	DRTS	Software Framework	HIL	
EPOCHS [10]	Investigating electromagnetic scenarios involving communication	No	ns-2	No	PSCAD/EMTDC PSLF	No	No
INSPIRE [11]	Analysing real-time performance of wide area monitoring, protection and control application	No	OPNET	No	DIgSILENT PowerFacory MATLAB Java C++ GNU R	No	No
Shum et al. [12]	Facilitating to understand, evaluate and debug distributed smart grid software	No	OPNET	No	PSCAD/EMTDC JADE DecompositionJ	No	No
GECO [13]	Investigating wide area measurement and control schemes	No	ns-2	No	PLSF	No	No
Yang et al. [14]	Validation distributed controls in smart grids considering communication networks	No	None	No	Matlab Simulink Functional Blocks	Yes	No
Nguyen et al. [15]	Holistic evaluation of smart grid solutions addressing mainly power and ICT domains	No	Omnet++	Yes	Matlab	Yes	No
Bian et al. [16]	Analysing performance of smart grids in real-time	No	OPNET	Yes	Java Eclipse	Yes	No
Venkataramanan et al. [17]	Co-simulation test-bed for cyber attacks on micro-grids	No	CORE	Yes	Python	Yes	No
OOCoSim [18]	Simulating the integrated scenario on power and communication networks to evaluate the effects on networked control in the smart grids	No	OPNET	No	OpenDSS	No	No
Mirz et al. [19]	Co-simulation of smart grid scenarios by integrating power grid, network and market models	Yes	ns-3	No	Python Modelica	No	No
Our solution	Co-simulation of general purpose service in Smart Grids by using a multi-model approach	Yes	MiniNet Omnet++ ns-3	Yes	Python Matlab Simulink	Yes	Yes

TABLE I: Comparison among our co-simulation infrastructure and literature solutions.

third-party platforms.

The strength of this platform comes from its capability to host new different use cases, input data, or models by means of its modules in a plug-and-play fashion, with a minimum effort to change the overall setup. As an example, several distributed generations (DGs) can be integrated as either independent simulators or standalone physical generators thanks to our IoT-based architecture which is a novel contribution to the co-simulations of electric energy systems with respect to conventional setups. The proposed platform is designed for general purpose services, but its distinguished capability, comparing to many existing platforms, is related to co-simulation and real-like analysis of the use cases in which low communication latency is crucial and electromagnetic transient analysis is required. Coordination of switching of converters connecting emerging DGs to low inertial systems for voltage or frequency control, and improving reliability of network by advanced outage management system are two main examples of such use cases. In these applications, a large number of devices are also involved, which makes the fast coordination and connectivity of them a challenge. We believe IoT-based solutions would tackle this challenge and improve the performance. Regarding advanced outage management systems, which is the context of the use case discussed in this paper, a newly developed self-healing scheme named Decentralised Fault Detection and Isolation, and Centralised Restoration (De-FDI-CR) has been simulated by using the proposed platform. Successful deployment of such schemes creates self-healing distribution grids with high reliability and customer satisfaction.

To highlight our contribution, Table I reports a comparison of our solution with the reviewed smart-grid co-simulation environments. It highlights: i) what kind of co-simulation is performed; ii) if the framework is compliant with SGAM; iii) if and what network simulator has been integrated in the framework; iv) if a Digital Real-time simulator (DRTS) is used; v) which are the integrated software frameworks; vi) if HIL simulation are supported; vii) if the simulation environment interacts with IoT devices.

III. DISTRIBUTED MULTI-MODELS CO-SIMULATION PLATFORM

Figure 1 shows the proposed distributed multi-model co-simulation platform, which has been designed to simulate and assess general purpose services for a smart management of power grids by exploiting an event-drive approach. It follows the interoperability and inter-dependency among layers depicted in SGAM [1]. Its modularity takes advantage of the microservices software design pattern, which consists on developing software *as a suite of small services, each running in its own process and communicating with lightweight mechanisms* [24]. This increases flexibility and maintainability because services are *small, highly decoupled and focus on doing a small task* [25]. The platform implements Internet-of-Things (IoT) communication paradigms and technologies to allow a fast bidirectional communication among its entities, either hardware or software. In particular, it exploits the two main communication paradigms: i) *publish/subscribe* [26]
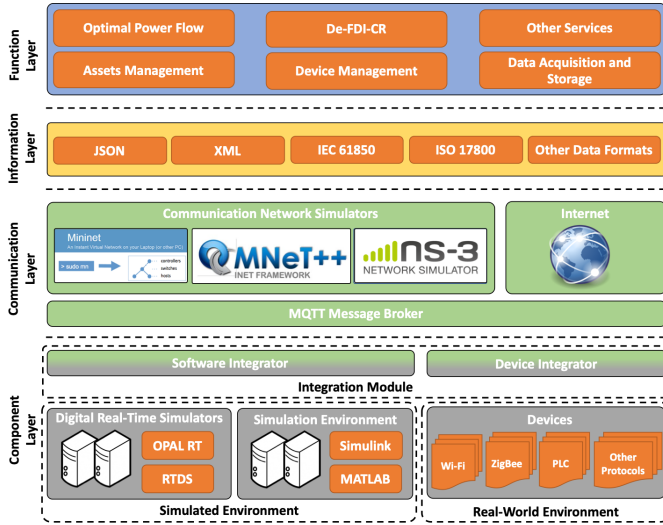
Fig. 1: Scheme of the proposed SGAM-Based Co-Simulation platform.

through MQTT protocol [27] and ii) *request/response* offering REST (Representational State Transfer) web services [28]. Publish/subscribe allows asynchronous communication that complements request/response. Furthermore, publish/subscribe enables the development of scalable loosely-coupled event-driven systems, services and applications that can react in (near-) real-time to certain events [29]. All these features ease developers in creating new simulation scenarios in a plug-and-play fashion also integrating third-party software. Moreover, our solution provides support to migrate from a fully simulated environment to the real-word. The rest of this section describes all the modules for each layer in the platform.

A. Component Layer

The **Component Layer** represents all the interacting entities, either hardware or software, needed to build a services operating in a smart grid scenario following a multi-model approach. As show in Figure 1, it is made up of three macro blocks: i) *Simulated Environment*, ii) *Real-world Environment* and iii) *Integration Module*.

On the one hand, the *Simulated Environment* consists of both digital real-time simulators (e.g. OpalRT and RTDS) and different software simulation frameworks (e.g. MATLAB and Simulink), where models are executed. Thanks to real-time capabilities provided by this macro-block, no significant latency is introduced to simulations. On the other hand, the *Real-world Environment* consists of heterogeneous hardware devices that exploit different protocols, either wireless or wired, to communicate (e.g. Wi-Fi, ZigBee, PLC).

The components of these two macro blocks are integrated into our co-simulation platform by exploiting the *Integration Module*. To allow a bidirectional communication, it acts as bridge between the underlying technologies and the rest of the platform following a methodology in [30]. In this prospective, each component in the Simulated or in the Real-world Environment needs its own *Software-* or *Device-Integrator*,

respectively. Once the corresponding integrator receives a new data from its low level technology, this data is translated into a common data-format and sent to the rest of the platform by exploiting either MQTT or REST, and vice-versa.

In a nutshell, the *Component Layer* provides features to enable a modular multi-model approach, where the different models, running in their own simulation frameworks, are combined together to build smart grid scenarios. Integration of DRTS allows simulations that realistically reproduce electromagnetic transient in distribution and transmission networks enabling also HIL. Finally, devices deployed in the real-word can be integrated in our platform to feed and asses novel services with real data.

B. Communication Layer

Future smart grids will be equipped with several internet-connected devices that will communicate, even each other, to provide a service [31]. Thus, evaluating the impact of data transmission in existing communication networks is of paramount importance to address service requirements even in terms of data transmission latency [9]. In this view, the **Communication Layer** helps in connecting the different entities, either hardware or software, in our co-simulation platform.

As shown in Figure 1, the first module is the *MQTT Message Broker*, which is the main actor to allow a bidirectional and asynchronous communication based on the MQTT protocol. Thus, it routes data from publisher to subscribers.

This layer optionally provides different simulators to realistically emulate communication networks. Thus, developers can optionally switch from the real-word MAN to a virtual MAN to study their simulation scenarios also from the data transmission viewpoint, i.e. transmission latency, bandwidth, low level protocols and access media (e.g. fibre optic, LTE, NBIoT, and WiMax). Following a methodology in [32], [33], we embedded three main *Communication Network Simulators* that can be alternatively chosen by developers according to their assessments: i) *Mininet* [34], ii) *Omnet++* [35] and iii) *ns-3* [36]. Mininet is a network emulator for building virtual networks consisting of hosts, switches, controllers and links exploiting both real kernel Linux and real network stack. It is recommended to evaluate end-to-end time latency and maximum bandwidth of communicating devices. Whilst, Omnet++ and ns-3 are two frameworks for in-depth simulations of different physical layers and protocols in communication networks. They also allow assessments of the overall Internet Protocol suite. Both Omnet++ and ns-3 are suitable for performing stack message exchange analysis to design innovative protocols. These simulators implement real-time software schedulers, hence also this layer does not introduce significant latency to whole simulation scenarios. In a test-bed co-simulation environment, the selected communication network simulator can be deployed in a server working as a virtual router where the virtual MAN runs. Thus, all the communication traffic generated by the entities in the platform is routed through this virtual router to analyse delays, congestion and packet losses.

This is valid also for communication flows either based on MQTT or REST.

C. Information Layer

The **Information Layer** in Figure 1 defines data-formats and information models for the data exchanged between the entities, either hardware or software, in our co-simulation platform. By default, it proposes open-standard formats that are completely language-independent, such as JSON and XML. Both data-formats are widespread used in distributed software architectures and web applications. In particular, JSON is becoming a standard for data exchange, even among IoT devices, because it is i) lightweight, ii) easy to understand, manipulate and generate, and iii) self-describing and human readable. Nonetheless, our platform is opened in integrating other data-formats according to requirements of the different simulation scenarios.

Regarding the information models, our solution is compliant with both IEC 61850 [37] and ISO 17800 [38] that are the two main standards for making communication between Intelligent Electronic Devices (IEDs) and control centres. The former is used for information exchanged between electrical substation and IEDs. Whilst, the latter is used for data exchanged between control systems and end use IEDs.

D. Function Layer

The **Function Layer** in Figure 1 includes the main functionality that is commonly used by the other entities in the co-simulation platform, i.e. i) *Asset Management* to manage information related to smart grid equipment and their geo-referenced locations; ii) *Device Management* to manage interactions between devices and other entities in the platform (e.g. services); and iii) *Data Acquisition and Storage* to store real and/or simulated data coming from other entities in the platform into scalable non-relational database (e.g. MongoDB). This layer defines also the functionality that a general purpose service provides for a smart management of electrical distribution network, such as i) *Optimal Power Flow* and ii) *Decentralised Fault Detection and Isolation, and Centralised Restoration*, a.k.a. De-FDI-CR (see Section IV). This paper is focused on services to make smart grids support self-healing and resilient to faults. However, thanks to its modularity, the platform is flexible in defining and including additional new services that can either play concurrently or be replaced in a plug-and-play fashion.

E. Sequence diagram of a generic smart grid scenario

This Section describes the communication flow in our platform for a generic smart grid scenario to be simulated. Figure 2 reports a sequence diagram with the interaction among the possible actors in the co-simulation, each of them running a different model.

In this example, we suppose that the actors are running in four different servers and a Digital Real-Time Simulator (DRTS) that are connected in a real dedicated local area network designed to minimise network latency (e.g. by exploiting gigabit Ethernet technology).
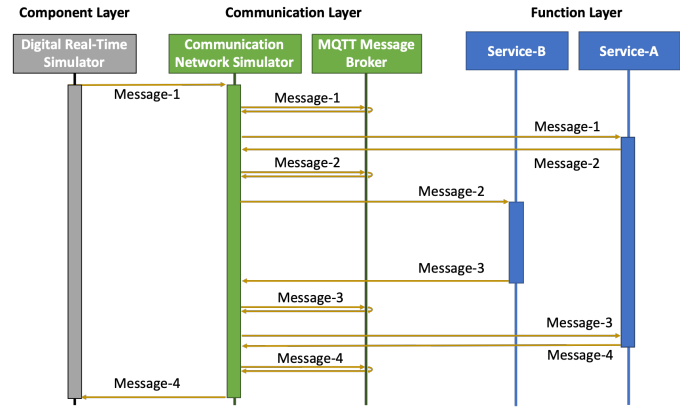

Fig. 2: Sequence Diagram of a generic smart grid scenario.

As depicted in Figure 2, DRTS runs a model reproducing the behaviour of a distribution network that exchanges information with the other actors in the system by exploiting the MQTT protocol. When DRTS publishes the *Message-1* with, for instance, some alerts or measurements to MQTT Message Broker, this message is routed through the virtual MAN simulated in one of the integrated communication network simulators to realistically assess the delay in the transmission till reaching the subscribed Service-A. To fulfil its computation, Service-A invokes Service-B by sending the *Message-2* again via MQTT routed through the virtual MAN. After a post-process performed by Service-B, it replays to Service-A sending the resulting *Message-3*. Finally, Service-A ends its computation and sends the *Message-4* with, for example, an actuation command to DRTS.

IV. SERVICE: FAULT DETECTION, ISOLATION AND RESTORATION

In this section, a self-healing scheme for modern distribution systems is described, and it is mapped to SGAM according to the proposed co-simulation platform. The scheme aims to perform Decentralised Fault Detection and Isolation, and Centralised Restoration (De-FDI-CR).

FDIR aims to automatically provide detection and isolation of the faulted portion of distribution system and restore the service for customers on healthy sections of the grid [39]. It makes decisions using over-current protection relays, loss of voltage on any phase, presetting capacity limits, and typically what the other switches see in the system. FDIR does not locate faults [40], however it intends to restore most of customers as fast as possible, which would avoid extra techno-economic costs in emergency [41].

There are several logic architectures for FDIR including centralised logic and distributed logic. In decentralised logic, switch/re-closer devices communicate among each other to locate and isolate the fault, and eventually restore the healthy part of the feeder [42]. The IEC 61850 based peer-to-peer communication technology is often used to deploy this scheme [43]. This logic is usually faster and more reliable, but restoration decisions are not necessarily the best solutions; lack of real time load information from neighbouring or supporting
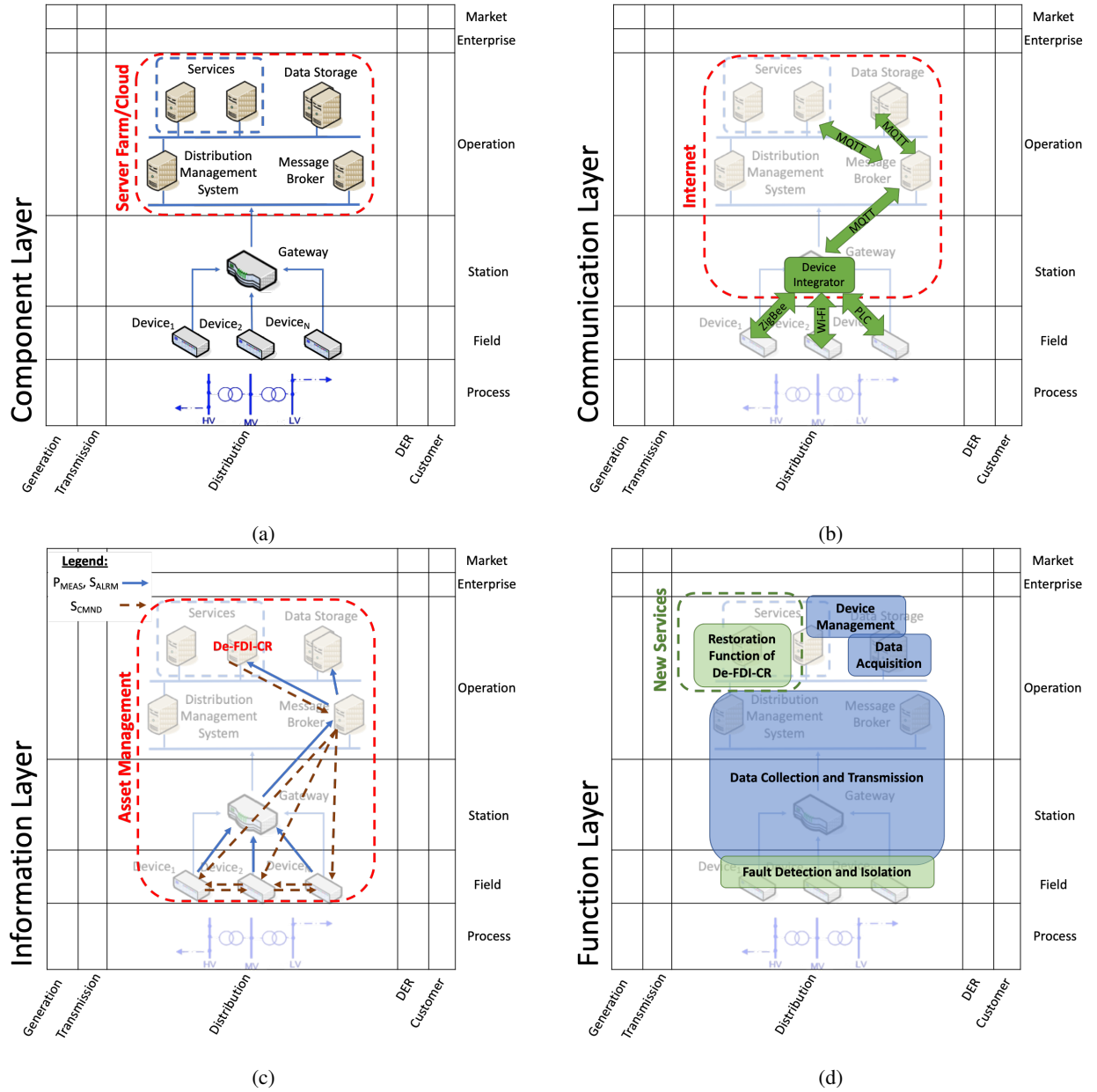
Fig. 3: De-FDI-CR service mapping to SGAM Layers

feeders for reconfiguration may result in line congestion or even over-current causing consequent faults.

In the Centralised logic, Distribution Management System (DMS) performs a high-level optimisation of feeder reconfiguration in case of occurrence of faults. Power flow algorithms would support an effective load distribution by using existing switches in the system [44]. Currently, the challenge is to make direct and efficient communication between devices in the network with DMS and also accurate load model information for an effective near real-time response.

This paper proposes an FDIR scheme which takes advantages of both centralised and decentralised logic. In this scheme, fault detection and isolation are managed very close to field thanks to distributed intelligence, while the restoration

is done by a reconfiguration decision made by a centralised algorithm. In other words, the scheme performs Decentralised Fault Detection and Isolation, and Centralised Restoration. This hybrid solution aims to minimise the above-mentioned challenges of Centralised logic by enhancing communication among devices using IoT paradigms, and integrating near real-time load data collected periodically from substation smart meters.

Once a fault occurred, the closest upstream device (Device-A) detects it and opens a switch/re-closer towards the fault. This device sends a signal to the next downstream device to request opening its switch/re-closer, and isolate the faulty section. As the effect of fault might trigger main feeder protection relay, at primary substation, circuit breaker (CB) of the feeder would be open. Once the fault is isolated, CB and all

upstream switch/re-closer, which are opened, should be closed. Upstream switches might be open due to their relay settings as reaction to the fault current. In order to close them all in a distributed way without any distant coordination, Device-A sends a command to the next upstream IED to close its switch, if it is opened, and the latter device passes the same signal to the next one upstream. Switch re-closing commands or trip messages are transferred to all next upstream devices to close all, including finally the CB. At this stage, actions of fault detection and isolation are completed, but the rest of feeder is not re-energised yet. In our scheme, a central restoration agent is in charge of restoring supply of the other loads in the feeder. It gets the most updated data of the load and generation which is periodically uploaded to a data storage (e.g. every minute or every 15-minute data). It also accesses the network topology data in the data storage at DMS to check all available Normally Open (NOP) switches which can be closed to supply the rest of affected feeder.

Device-A sends an alarm message to this agent reporting a permanent fault at its downstream section. Since some of other upstream devices might sense the fault and send similar alarms, the agent supposes the fault to be right after the farthest one from the head of the feeder. It, then, creates a set of possible network reconfiguration based on existing NOPs, and then call an Optimal Power Flow (OPF) tool in DMS to eventually rank the countermeasures; different objectives may be considered, however to improve system reliability, the best choices are the ones which would not cause other failures due to over-current or reverse power flow in other lines. In our scheme, we minimise the line loss and the number of NOP switching.

Figure 3a maps the main actors of our service to different zones within distribution domain. The devices include all physical components of the field. Figure 3b describes protocols and mechanisms for the interoperable exchange of data among components in the context of the underlying service. Fault detection and then isolation are performed thanks to peer-to-peer communication among devices in the field. Power measurements to be used for power flow analysis are transferred to the operation zone through data concentrator. MQTT is used for making communication between operation zone and station or field zones. Figure 3c represents the data type exchanged among actors; devices in the field exchange commands to pass the detection alarm and eventually isolate the faulted section. They also send trip alarms of breakers or re-closers directly to message broker, and listen to it to receive actuation commands for network reconfiguration and supply restoration.

Figure 3d maps the specific functions involved in this service to appropriate zones within distribution domain. As discussed earlier, De-FDI-CR would perform fault detection and isolation through a distributed logic using intelligent devices, and restore the supply for healthy, yet interrupted sections of network through central logic.

V. CASE STUDY AND EXPERIMENTAL SETUP

To demonstrate the performance of our developed FDIR scheme as well as the functionality of the co-simulation
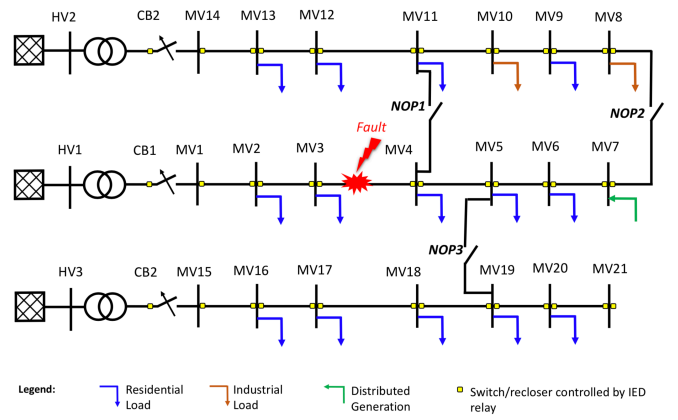

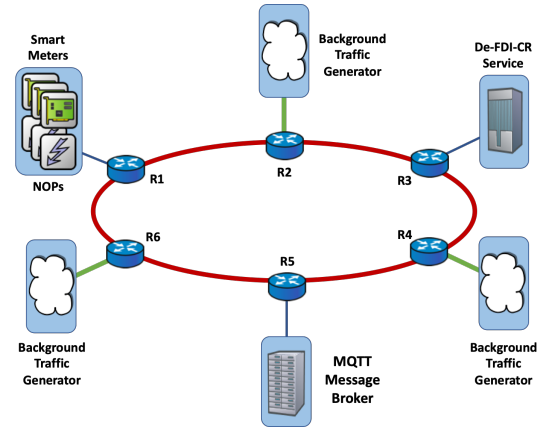Fig. 4: Grid topology for co-simulation tests of De-FDI-CR.

Fig. 5: MAN topology for De-FDI-CR co-simulation tests.

platform, we use two cases as IEEE 34-node radial network [45] and a portion of an urban distribution grid, respectively. The former is used to present what kind of FDIR scheme we integrate to our overall platform, and the latter aims to demonstrate the whole process of self-healing mechanism including communication system and the test platform.

In this section, we introduce the realistic case used for testing the co-simulation platform and we explain the lab setup built according to SGAM to verify the applicability of the proposed method. We use a small portion of an urban distribution grid with three feeders and three normally open switches for different reconfiguration choices. This network is extracted from the real distribution grid of Turin, a city in Northwest of Italy. Figure 4 represents the topology of the case study and the location of the fault. It is composed of 3 primary and 18 secondary substations, with 220 kV at high voltage and 22 kV at medium voltage side. Fourteen nodes are supplying residential loads, two are feeding industrial loads and there is one substation connected to a generation plant. In our tests, a constraint on the radial operation of the grid is considered. Three of the branches (between nodes MV4 and MV11, between nodes MV7 and MV8, and between nodes MV5 and MV19) are structurally connected and considered as normally open. Supply restoration is performed by closing one of these NOP switches once the fault is isolated.
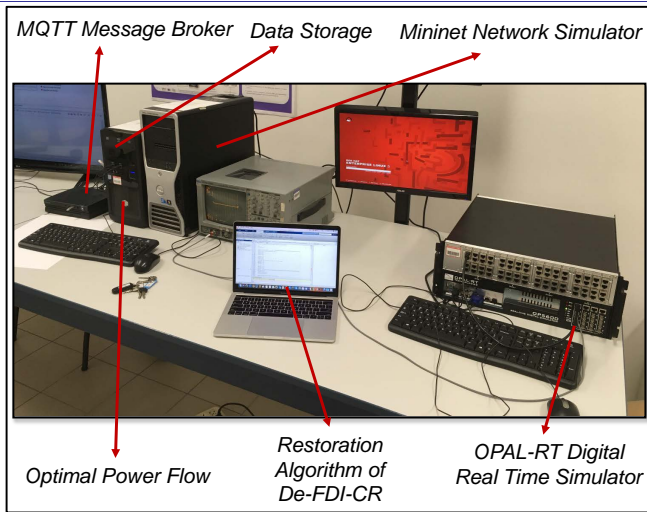
Fig. 6: Lab set-up for De-FDI-CR co-simulation tests.

The model of this grid is implemented in MATLAB Simulink as the development environment of eMEGAsim solver of Opal-RT real time simulator. The model is compiled by RT-Lab and executed on a 5600 Opal-RT digital simulator. For this electromagnetic transient analysis, a 250-microsecond time-step is considered which makes a suitable sampling for modelled over-current relay. A single-phase to ground fault is triggered between substation MV3 and MV4.

Once the fault is triggered, upstream switch/re-closer at substation MV3 opens followed by an IED relay command, and it sends opening command to the downstream switch/re-closer at MV4. In this way, faulted section is isolated.

Once the fault is isolated, IED at MV3 sends alarm signal to the restoration algorithm of De-FDI-CR, and re-closing command to its closest upstream substation (i.e. MV2 in this grid). The latter is to ensure all switches at upstream substations are closed. This is because faults could also trigger relays at other upstream substations, especially the one at the beginning of the feeder. Therefore, all substations from MV3 to the head of the feeder would pass re-closing command one by one through peer-to-peer communication. The last switch to be re-closed - in case it was open - is the main circuit breaker of the feeder. This in-field process ensures restoration of all interrupted loads before the location of fault.

Supply restoration of the rest of feeder requires a high-level or centrally supervised coordination of several existing Normally Open (NOP) switches to achieve an optimal reconfiguration of the network. Since reliability of supply is the main objective of our use case, this case-study does not consider a network reconfiguration at a wide area involving many other feeders. This is also a practical approach which prevents unnecessary switching which may affect switches with ageing.

Regarding the case study, the central agent of De-FDI-CR should find which of the three NOPs to close safely to guarantee a stable restoration of supply. Integration of distributed generation in smart grids implies that the network may face operational problems like reverse power flow, line congestion, over/under voltage situation, if it is not well managed. Therefore, an ex-ante optimal power flow analysis is needed prior to actuate the appropriate NOP for reconfiguration. Centralised Restoration (CR) logic gets the most updated data of the load and generation which is periodically uploaded to a data storage (e.g. every 15-minute data). During fault restoration process, it also accesses the network topology data to check all available Normally Open (NOP) switches which can be closed to supply the rest of affected feeder. To minimise losses and respecting all network constraints including voltage and current acceptable ranges, CR invokes an OPF tool in DMS for three possible network reconfiguration options based on the existing NOP status. This is performed through REST protocol.

Figure 5 shows the scheme of a realistic MAN backbone modelled to test and evaluate the performance of the De-FDI-CR service, also in terms of communication latency with the IEDs. Typically, modern MAN backbones leverage upon fibre optic links deployed across metropolitan areas and connecting different backbone routers in a ring configuration. These MANs, based on fibre optic technology, can guarantee connections with 100 Mbps. In our model, the links between two routers forming the ring (red lines) are full-duplex with a max length of about 10 km each and zero losses. We supposed that all IEDs deployed in the portion of distribution grid are connected to the Internet through the backbone router R1 in Figure 5. IEDs communicate with the De-FDI-CR service via MQTT through the message broker that is connected to routers R3 and R5, respectively. R1, R3 and R5 are connected with their respective sub-networks through 10Mbps full-duplex links (blue lines) with a max length of about 1 km and zero losses. Finally, routers R2, R4 and R6 serve three other different sub-networks that generate background traffic with different rates to realistically congest the MAN. To avoid bottlenecks between these last three routers and their traffic generators, a 200Mbps full-duplex links have been chosen (green lines) with max length of about 1 km and zero losses. Thus in our simulations, IEDs, De-FDI-CR service and message broker are in different locations in the same metropolitan area and communicate over the Internet. For these simulations, Mininet has been exploited.

Figure 6 reports our laboratory set-up where the entities and models of our co-simulation platform have been deployed for this scenario. In particular, an *OPAL-RT Digital Real Time Simulator* and four computers have been involved and used to run i) the *MQTT Message Broker*, ii) both *Data Storage* and the *Optimal Power Flow* algorithm, iii) the *Mininet Network Simulator* and the iv) *Restoration Algorithm of De-FDI-CR*, respectively.

These computers and the OPAL-RT are connected through a 100Gbps Ethernet switch that introduces a negligible communication latency of about 10ns. Gigabit Ethernet equipment provides backward compatibility to older 100 Mbps and 10 Mbps legacy Ethernet devices [46]. The computers and the OPAL-RT belong to the same dedicated local area network and communicate following the sequence diagram described in Section III-E. Thus, all the traffic generated by these actors is routed through the virtual router where the virtual MAN
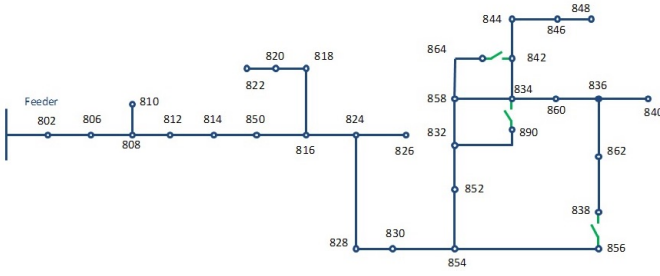
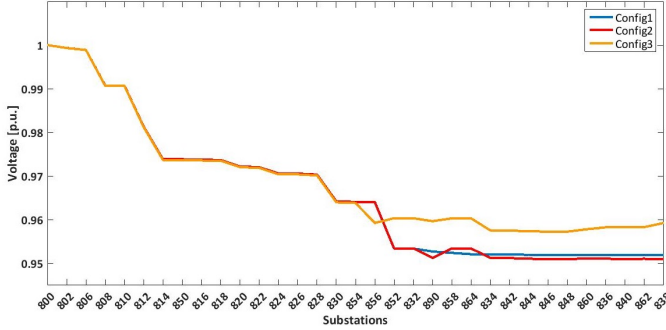Fig. 7: IEEE 34-node radial network with three normally open switches.

Fig. 8: Voltage profiles corresponding to different reconfiguration options - IEEE 34-Node Grid.

runs in Mininet.

VI. EXPERIMENTAL RESULTS

This Section presents the experimental results of applying our developed FDIR scheme on both the IEEE 34-node radial network [45] and a portion of an urban distribution grid, respectively (see in Section V).

A. IEEE 34-Node Case Study

Figure 7 shows a single line diagram of IEEE 34-node system. In order to examine different reconfiguration options, we added three normally open (NOP) switches between nodes 864 and 842, 890 and 834, and 856 and 838. The model of this case is implemented in MATLAB Simulink to be loaded to Opal-RT real-time simulator for running a real-time simulation. A single-phase to ground fault is triggered between nodes 858 and 834.

After detecting the fault and isolating the branch between nodes 858 and 842, FDIR central agent should decide which NOP to close to not only restore the rest of network, but also minimise the power loss and achieve the best possible voltage profile. Figure 8 plots different voltage profiles corresponding to different configurations. *Config1*, *Config2*, and *Config3* are configurations of the network after closing switches of 864-842, 890-834, and 856-838, respectively.

Figure 9 compares different power flows over lines considering different configuration options. Results plotted in Figure 8 and Figure 9 shows that *Config3* has the best system performance comparing to the other options.
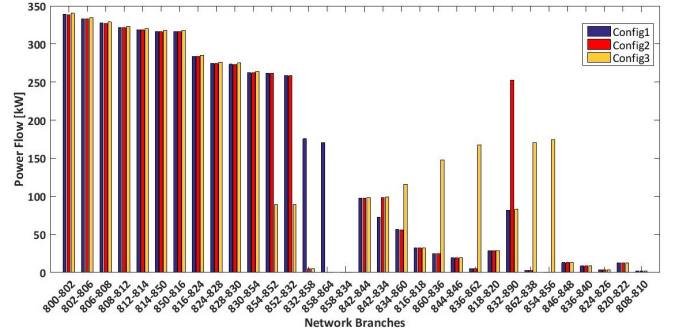

Fig. 9: Comparing power flow of the lines after three different reconfiguration cases - IEEE 34-Node Grid.

To quantify this difference, we calculated the voltage deviation index (VDI) defined as the sum of square of voltage deviation at each node (Equation 1). The voltage deviation index for *Config1*, *Config2*, and *Config3* are $0.0455pu$, $0.0463pu$, and $0.0373pu$, respectively. From power flow results, the total power loss of the system is the least with *Config3*. The total power loss for the three configurations as *Config1*, *Config2*, and *Config3*, is obtained $5011.6kW$, $4921.5kW$, and $4813.5kW$, respectively. According to the results, FDIR finally choose *Config3* to restore the system after fault.

$$VDI = \sum_{i=1}^{N} |V_n - V_i|^2 \qquad (1)$$

B. Three-feeder realistic case study in Turin

This section presents the experimental results of the proposed FDIR schema applied to the three-feeder network shown in Figure 4. This also demonstrates how our co-simulation platform would support an interoperability analysis of a use case following the SGAM.

Once the fault is triggered, an Inverse Definite Minimum Time over-current relay (IDMT) at MV3 detects violation and trigger a sectionalizer to disconnect supply of the rest of feeder from MV3. Figure 10 presents voltage and current signals captured by the relay IED next to the fault location (i.e. at MV3). In our case, all relay are tuned with the same settings of IDTM. Hence in case of neglecting delay of waveform propagation along the line, occurrence of a fault anywhere on the feeder would trigger all upstream relays. This is intentionally set in order to stress FDIR reaction for more peer-to-peer command exchange. The reaction is the chain of re-closing commands starting from MV3 to the beginning of the feeder. Re-closing all open switches of the upstream side is a part of supply restoration to re-energise all the loads of the feeder before the location of the fault.

Prior to restoration, the decentralised logic isolates the section between MV3 and MV4. This is done by sending a command from MV3 to MV4 right after detecting the fault.

After isolating the fault, MV3 initiates the chain of peer-to-peer command pass as discussed above. At the same time, MV3 also sends an alarm message to the centralised logic

of De-FDI-CR which contains the hardware ID of the IED installed at MV3. This launches the algorithm to create three possible network configurations based on the status of NOP1, NOP2 and NOP3; and select the best option to restore the rest of feeder. The three configurations formed by closing NOP1, NOP2 and NOP3 are named *Config1*, *Config2*, and *Config3*, respectively.

The affected feeder has a distributed generator with $29.05kW$ and $3.1kVAr$ production at the time of study. All the load and generation values are collected by meter IEDs (i.e. smart meters) installed at the substations. The data is reported out every 15 minutes through communication system. In our case, meter and relay IEDs are the same devices with the three features as well as event recording.

Considering the distributed generator as uncontrollable generation, an Optimal Power Flow (OPF) is called for the three configurations. We use the standard AC OPF that consists of 18 vectors of voltage angles and magnitudes and 2 vector of generator real and reactive power injections. Since there are only two controllable generators which belong to the same network operator, the objective would be loss minimisation rather than cost minimisation. The equality constraints are simply the full set of 36 nonlinear real and reactive power balance equations as there are 18 involved nodes. The inequality constraints consist of two sets of 17 branch flow limits as non-linear functions of the bus voltage angles and magnitudes. The variable limits include an equality constraint on any reference bus angle and upper and lower limits on all bus voltage magnitudes and real and reactive generator injections. The limits for the node voltage magnitudes are set between $0.95pu$ and $1.05pu$.

Central logic algorithm is written in Python and it calls MATLAB MATPOWER to execute OPF. Changing the configurations and running OPF for the three cases, resulted in three different sets of power flow of the lines (Figure 11). We obtained $8.20kW$, $9.16kW$, and $7.48kW$ power loss for *Config1*, *Config2*, and *Config3*, respectively.

Figure 12 is depicted to represent the voltage profile of the three cases; as it is shown in this figure, *Config3* has a much better voltage profile comparing to *Config1* and *Config2* in most of the substations.

As for the IEEE 34-node radial network, we calculated the VDI following the Equation 1. The voltage deviation obtained given by the difference between calculated voltage and the rated voltage equal to $1pu$ or $22volts$. Voltage deviation index for *Config1*, *Config2*, and *Config3* are $0.0012pu$, $0.0021pu$, and $0.0008pu$, respectively.

Since, no violations or reverse power flow are observed in the three cases, all are acceptable choices, however ranking them based on either power losses or voltage deviation index makes CR send actuation command to NOP3 to reconfigure the network as *Config3* shown in Figure 14.

Figure 13 reports the results on communication latency when a fault is located and restored at different congestion rates, neglecting the computation time of De-FDI-CR service. The computation time of the algorithm did not exceed a few milliseconds ($10ms$ to $20ms$) which is negligible in our case. This is the time needed to send an alarm from smart meters
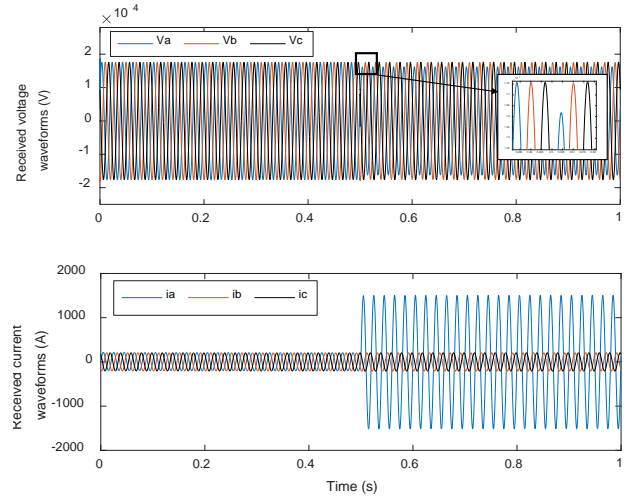

Fig. 10: Retrieved wave-forms for the single-line to ground fault.
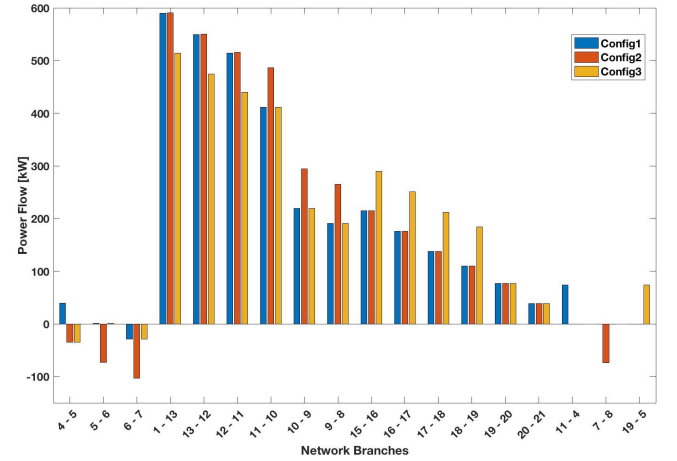

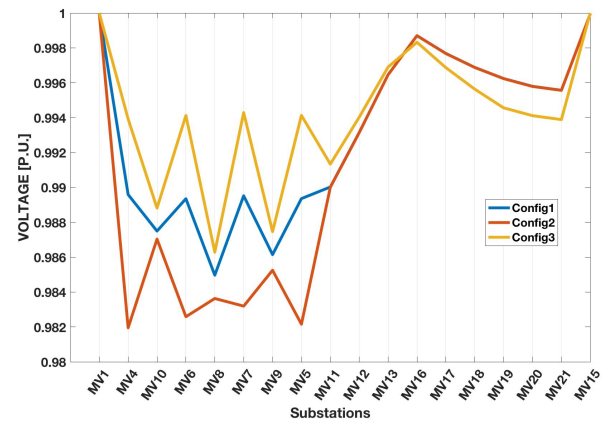Fig. 11: Comparing power-flow of the lines in the three configurations.

Fig. 12: Comparing voltage profiles in the three configurations.

to De-FDI-CR and to send back an actuation command from De-FDI-CR to NOPs via MQTT across the MAN as described in Section V. Both alarms and commands are sent as messages
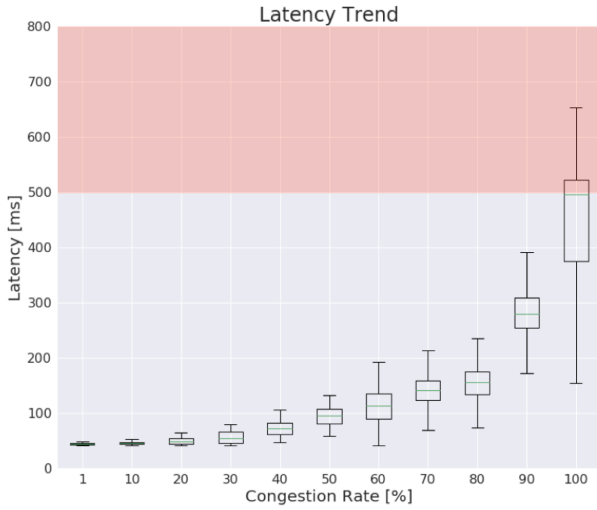
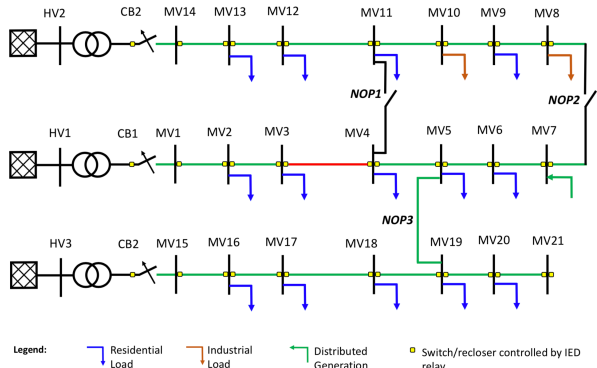Fig. 13: Communication latency at different congestion rates.

Fig. 14: Final reconfiguration of the network for restoration.

Performance Requirements	Performance Classes	Values	Example of services
Transfer Time	TT0	>1000 ms	Files, events, log contents, SCADA
	TT1	1000 ms	Events, alarms
	TT2	500 ms	Operator commands
	TT3	100 ms	Slow automation interactions
	TT4	20 ms	Fast automation interactions
	TT5	10 ms	Releases, status changes
	TT6	3 ms	Trips, blockings

TABLE II: Communication requirements and performance classes for power systems defined by IEC61850 [9]

a realistic simulation scenario. Indeed, referring to the performance classes in Table II, defined by the standard, our experimental results on time delay satisfy the requirements of classes *TT0*, *TT1* and *TT2*. This is also confirmed when the MAN congestion rate is 100%, which is a very critical situation in communication network. However in this very worst scenario, only few packets exceeded the $500ms$ reaching a max time delay of about $650ms$ (see red portion in Figure 13). It is worth noting that Internet providers and network managers try to avoid this critical situation that, in long periods, could lead to the collapse of MAN itself.

VII. CONCLUSIONS

Deployment of new algorithms and control strategies in smart grids requires interoperability analysis and systematic transitions. This implies using reference models like SGAM to map the new use cases. Mapping use case in SGAM enables the possibility of performing an interoperability analysis that is overlooked in classic laboratory tests.

This paper demonstrated how our multi-model co-simulation platform could be used to map and simulate new strategies and algorithms with a SGAM-based approach. The integration of digital real-time simulators enable the possibility of developing HIL setups for testing physical devices while integrating new algorithm and strategies. Moreover, the integration of communication network simulators in the platform enables the possibility to test and develop use cases by taking into account the performance of the communication networks and protocols.

The experimental results of simulating a self-healing scheme use case by means of our platform demonstrated both the capability of our platform and the effectiveness of the proposed service. A new FDIR approach, named De-FDI-CR, is proposed with capability of both centralised and decentralised FDIR logic; while solving the limitations of the two logic. It respects the communication latency constrain of the IEC 61850 standard using the MQTT protocol over a MAN for exchanging messages. This was a limitation in conventional centralised FDIR. Possible violation less than 25% of messages violate the maximum latency constrain only in the condition of fully congested MAN. While, we are also exploiting a centralised intelligence, the fast restoration capability of decentralised logic is respected. Our co-simulation platform realised the feasibility of deployment of this scheme in the real-world system.

of about 85 bytes each, compliant with the JSON data-format.

As shown in Figure 13, our analysis started from considering first the MAN with a congestion rate of 1% where routers are almost completely unloaded. Thus, routers process incoming packets almost immediately. In this scenario, the median value of the communication time needed to report and to restore a fault is about $40ms$.

Increasing the overall MAN congestion up to 50% by increasing the background traffic, the median value for the communication latency raised almost constantly without exceeding the $100ms$. This is due to the impact of the generated background traffic on routers. Indeed, the time spent by a packet in a router before being processed increases and this creates queues on the routers, as expected.

Growing the congestion from 60% to 80%, the median value for the communication latency slightly increased from almost $110ms$ to $160ms$. Finally, with MAN congestion equal to 90% and 100% the median time delay is about $280ms$ and slightly lower than $500ms$, respectively.

The IEC 61850 standard [9] defines the communication requirements to be addressed in the power distribution networks. Thanks to our distributed multi-model co-simulation platform, we are able to evaluate these communication requirements for a specific service (i.e. De-FDI-CR) in

REFERENCES

[1] J. Bruinenberg, L. Colton, E. Darmois, J. Dorn, J. Doyle, O. Elloumi, H. Englert, R. Forbes, J. Heiles, P. Hermans *et al.*, "Cen-cenelec-etsi smart grid co-ordination group smart grid reference architecture," *CEN, CENELEC, ETSI, Tech. Rep*, pp. 98–107, 2012.

[2] H. Farhangi, "The path of the smart grid," *IEEE power and energy magazine*, vol. 8, no. 1, 2010.

[3] C. Dufour and J. Bélanger, "On the use of real-time simulation technology in smart grid research and development," *IEEE Transactions on Industry Applications*, vol. 50, no. 6, pp. 3963–3970, 2014.

[4] F. Guo, L. Herrera, R. Murawski, E. Inoa, C.-L. Wang, P. Beauchamp, E. Ekici, and J. Wang, "Comprehensive real-time simulation of the smart grid," *IEEE Transactions on Industry Applications*, vol. 49, no. 2, pp. 899–908, 2013.

[5] W. Li, M. Ferdowsi, M. Stevic, A. Monti, and F. Ponci, "Cosimulation for smart grid communications," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 4, pp. 2374–2384, Nov 2014.

[6] A. Estebsari, E. Patti, and L. Barbierato, "Fault detection, isolation and restoration test platform based on smart grid architecture model using intenet-of-things approaches," in *2018 IEEE International Conference on Environment and Electrical Engineering and 2018 IEEE Industrial and Commercial Power Systems Europe (EEEIC/I&CPS Europe)*. IEEE, 2018, pp. 1–5.

[7] L. Bottaccioli, A. Estebsari, E. Pons, E. Bompard, E. Macii, E. Patti, and A. Acquaviva, "A flexible distributed infrastructure for real-time co-simulations in smart grids," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 6, pp. 3265–3274, 2017.

[8] E. Patti and A. Acquaviva, "Iot platform for smart cities: Requirements and implementation case studies," in *Research and Technologies for Society and Industry Leveraging a better tomorrow (RTSI), 2016 IEEE 2nd International Forum on*. IEEE, 2016, pp. 1–6.

[9] IEC61850-5 Edition, "Communication networks and systems in substations–part 5: Communication requirements for functions and device models."

[10] K. Hopkinson, X. Wang, R. Giovanini, J. Thorp, K. Birman, and D. Coury, "Epochs: a platform for agent-based electric power and communication simulation built from commercial off-the-shelf components," *IEEE Transactions on Power Systems*, vol. 21, no. 2, pp. 548–558, 2006.

[11] H. Georg, S. C. Mller, C. Rehtanz, and C. Wietfeld, "Analyzing cyber-physical energy systems:the inspire cosimulation of power and ict systems using hla," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 4, pp. 2364–2373, Nov 2014.

[12] C. Shum, W. Lau, T. Mao, H. S. Chung, K. Tsang, N. C. Tse, and L. L. Lai, "Co-simulation of distributed smart grid software using direct-execution simulation," *IEEE Access*, vol. 6, pp. 20 531–20 544, 2018.

[13] H. Lin, S. S. Veda, S. S. Shukla, L. Mili, and J. Thorp, "Geco: Global event-driven co-simulation framework for interconnected power system and communication network," *IEEE Transactions on Smart Grid*, vol. 3, no. 3, pp. 1444–1456, 2012.

[14] C.-h. Yang, G. Zhabelova, C.-W. Yang, and V. Vyatkin, "Cosimulation environment for event-driven distributed controls of smart grid," *IEEE Trans. on Industrial Informatics*, vol. 9, no. 3, pp. 1423–1435, 2013.

[15] V. H. Nguyen, Y. Besanger, Q. T. Tran, C. Boudinnet, T. L. Nguyen, R. Brandl, and T. I. Strasser, "Using power-hardware-in-the-loop experiments together with co-simulation for the holistic validation of cyber-physical energy systems," in *2017 IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT-Europe)*, Sept 2017.

[16] D. Bian, M. Kuzlu, M. Pipattanasomporn, S. Rahman, and Y. Wu, "Real-time co-simulation platform using opal-rt and opnet for analyzing smart grid performance," in *2015 IEEE Power Energy Society General Meeting*, July 2015, pp. 1–5.

[17] V. Venkataramanan, A. Srivastava, and A. Hahn, "Real-time co-simulation testbed for microgrid cyber-physical analysis," in *2016 Workshop on Modeling and Simulation of Cyber-Physical Energy Systems (MSCPES)*, April 2016, pp. 1–6.

[18] H. Kim, K. Kim, S. Park, H. Kim, and H. Kim, "Cosimulating communication networks and electrical system for performance evaluation in smart grid," *Applied Sciences*, vol. 8, no. 1, p. 85, 2018.

[19] M. Mirz, L. Razik, J. Dinkelbach, H. A. Tokel, G. Alirezaei, R. Mathar, and A. Monti, "A cosimulation architecture for power system, communication, and market in the smart grid," *Complexity*, vol. 2018, 2018.

[20] "IEEE 1516-2010 - IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA)– Framework and Rules." [Online]. Available: https://standards.ieee.org/standard/1516-2010.html

[21] "The Network Simulator - ns-2." [Online]. Available: https://www.isi.edu/nsnam/ns/

[22] "OPNET Network Simulator." [Online]. Available: https://www.riverbed.com/gb/products/steelcentral/opnet.html

[23] S. Schütte, S. Scherfke, and M. Tröschel, "Mosaik: A framework for modular simulation of active components in smart grids," in *Proc. of SGMS11*. IEEE, 2011, pp. 55–60.

[24] M. Fowler and J. Lewis, "Microservices," 2014. [Online]. Available: http://martinfowler.com/articles/microservices.html

[25] S. Newman, *Building Microservices*. O'Reilly Media, Inc., 2015.

[26] P. T. Eugster, P. A. Felber, R. Guerraoui, and A.-M. Kermarrec, "The many faces of publish/subscribe," *ACM CSUR*, June 2003.

[27] "Message Queue Telemetry Transport (MQTT)." [Online]. Available: http://mqtt.org

[28] R. T. Fielding and R. N. Taylor, "Principled design of the modern web architecture," *ACM Transactions on Internet Technology*, vol. 2, no. 2, pp. 115–150, 2002.

[29] E. Patti, A. L. A. Syrri, M. Jahn, P. Mancarella, A. Acquaviva, and E. Macii, "Distributed software infrastructure for general purpose services in smart grid," *IEEE Transactions on Smart Grid*, vol. 7, no. 2, pp. 1156–1163, 2016.

[30] E. Patti, A. Acquaviva, and E. Macii, "Enable sensor networks interoperability in smart public spaces through a service oriented approach," in *Advances in Sensors and Interfaces (IWASI), 2013 5th IEEE International Workshop on*. IEEE, 2013, pp. 2–7.

[31] E. Ancillotti, R. Bruno, and M. Conti, "The role of communication systems in smart grids: Architectures, technical solutions and research challenges," *Computer Communications*, vol. 36, no. 17-18, pp. 1665–1697, 2013.

[32] C. P. Mayer and T. Gamer, "Integrating real world applications into omnet++," *Institute of Telematics, University of Karlsruhe, Karlsruhe, Germany, Tech. Rep. TM-2008-2*, 2008.

[33] A. Varga, "Discrete event simulation system," in *Proc. of the European Simulation Multiconference (ESM'2001)*, 2001.

[34] "Mininet." [Online]. Available: http://mininet.org/

[35] "OMNeT++ Discrete Event Simulator." [Online]. Available: https://www.omnetpp.org/

[36] "The Network Simulator - ns-3." [Online]. Available: https://www.nsnam.org/

[37] J. Zhang, B.-C. Seet, and T. T. Lie, "An event-based resource management framework for distributed decision-making in decentralized virtual power plants," *Energies*, vol. 9, no. 8, p. 595, 2016.

[38] "ISO 17800:2017 - Facility smart grid information model." [Online]. Available: https://www.iso.org/standard/71547.html

[39] A. Estebsari, E. Pons, E. Bompard, A. Bahmanyar, and S. Jamali, "An improved fault location method for distribution networks exploiting emerging lv smart meters," in *2016 IEEE Workshop on Environmental, Energy, and Structural Monitoring Systems (EESMS)*, June 2016, pp. 1–6.

[40] A. Estebsari, E. Pons, E. Patti, M. Mengistu, E. Bompard, A. Bahmanyar, and S. Jamali, "An iot realization in an interdepartmental real time simulation lab for distribution system control and management studies," in *2016 IEEE 16th International Conference on Environment and Electrical Engineering (EEEIC)*, June 2016, pp. 1–6.

[41] A. Estebsari, E. Pons, T. Huang, and E. Bompard, "Techno-economic impacts of automatic undervoltage load shedding under emergency," *Electric Power Systems Research*, vol. 131, pp. 168–177, 2016.

[42] H. F. Habib, T. Youssef, M. H. Cintuglu, and O. A. Mohammed, "Multi-agent-based technique for fault location, isolation, and service restoration," *IEEE Transactions on Industry Applications*, vol. 53, no. 3, pp. 1841–1851, May 2017.

[43] D. L. Ransom and C. Chelmecki, "Using goose messages in a maintiemain scheme," *IEEE Transactions on Industry Applications*, vol. 50, no. 1, pp. 17–24, Jan 2014.

[44] A. B. Nassif, "An analytical assessment of feeder overcurrent protection with large penetration of distributed energy resources," *IEEE Transactions on Industry Applications*, vol. 54, no. 5, Sep. 2018.

[45] "IEEE 34-node radial network." [Online]. Available: https://ewh.ieee.org/soc/pes/dsacom/testfeeders.html

[46] "IEEE 802.3-2018 - IEEE Standard for Ethernet." [Online]. Available: https://standards.ieee.org/standard/8023-2018.html