Cloud-based IoT solution for State Estimation in Smart Grids: exploiting virtualization and edge-intelligence technologies

A. Meloni[a,*], P. A. Pegoraro[a], L. Atzori[a,*], A. Benigni[b], S. Sulis[a]

[a]*DIEE, University of Cagliari, Cagliari, Italy*
[b]*CEC, University of South Carolina, Columbia, SC, USA*

Abstract

Smart Grids (SGs) are expected to be equipped with a number of smart devices able to generate vast amounts of data about the network status, becoming the key components for an efficient State Estimation (SE) of complex grids. To exploit their potentials, the ICT infrastructure needs to be scalable to follow the increasing amount of data flows and flexible to give the possibility to assign and re-assign grid functions and data flow control policies at runtime, possibly in a context-aware manner. In this scenario, this paper proposes and validates a Cloud-IoT-based architectural solution for SE in SG that combines cloud-capabilities and edge-computing advantages and uses virtualization technologies to decouple the handling of measurement data from the underlying physical devices. Case studies in the field of distribution networks monitoring are also analyzed, demonstrating that the proposed architecture is capable to accomplish the assigned operational tasks, while satisfying the needed quality level from both the communication and the grid perspectives with a significant degree of flexibility and adaptability with respect to state of the art solutions.

Key words: Smart Grid, Internet of Things, Cloud, Edge, Virtualization,

*Corresponding author
Email addresses:* `alessio.meloni@diee.unica.it` (A. Meloni),
`paolo.pegoraro@diee.unica.it` (P. A. Pegoraro), `l.atzori@diee.unica.it` (L. Atzori),
`benignia@cec.sc.edu` (A. Benigni), `sara.sulis@diee.unica.it` (S. Sulis)

1. Introduction

The metamorphosis of power distribution systems from electric grids into Smart Grids (SGs) is currently underway and it is consistently changing, among the others, how operational tasks are performed. In particular, the introduction of Distributed Energy Resources (DERs) requires system operators to rethink the way grids are managed so as to face unexpected and quick dynamics. In order to tackle these needs, system operators are increasingly deploying new measurement devices such as Phasor Measurement Units (PMUs) and Smart Meters (SMs). These devices allow for gathering information to estimate the operating conditions of the grids, becoming central components in State Estimation (SE) functions in SG. In this regard, an effective monitoring system is required to exploit appropriately the features of the available devices [1].

The full achievement of a SG and implementation of SE also pass through an efficient and future-proof information and communication system able to keep up with the needs of SG operators [2][3][4]. PMUs are the most demanding sensing devices, since they can provide up to 60 measurements per second about the status of lines and nodes of the power grid, and in the near future this rate will increase significantly. Accordingly, the ICT infrastructure needs to be able to handle the resulting high-rate flows of measured data in a scalable way to address the expected wide deployment of smart grids. However, it is not only a matter of transmission capacity as the ICT infrastructure should be able to adaptively sense the status of the power grid, to understand where to send and store the data, and to make this data available for the different applications that are currently deployed and that can be deployed in the future on top of the smart grid relying on an effective SE. Accordingly, the use of proprietary infrastructures specifically and statically designed for each given use case is not a viable solution.

Such a choice would be poorly evolution-proof and would soon become inef-

ficient in a context where changes are happening at an unexpected rate and are not always predictable in terms of applications that will arise and data resources that will be used. In other words, planning an infrastructure statically designed for what is envisioned at a certain time, rather than deploying a future-proof solution could be cumbersome, since changes are far from being definite and definitive at the moment. More and more components are nowadays increasingly introduced, so that the costs for re-adapting a static design would rapidly become impractical, especially for Distribution System Operators dealing with portions of the grid as small as a city who are unlikely to deploy a brand new ICT infrastructure every time an evolution is needed [5].

The following characteristics are thus foreseen for the ICT infrastructure of the SG: scalability and elasticity, for accommodating big data flows and their related storage and computation operations; flexibility, for giving the possibility to assign and re-assign grid functions and data flow control policies also at runtime, possibly in a context-aware manner; possibility to virtualize (i.e. decouple) the information pool composed of measured data from the underlying physical devices, in order to make them reusable for more than one application and independent from the underlying communication and electrical specifics; future-proofness, so that new functions and devices can be added, removed or substituted in a modular manner without rethinking the entire ICT infrastructure from scratch.

For some of these characteristics, cloud computing can be seen as an enabling ICT technology. Both traditional cloud and edge computing models can be exploited in this scenario, where the later represents the configuration where network edge devices evolved into microcloud servers to be able to host not only advanced network functions but also application modules. This allows for better accommodating the desired level of quality of service and workload/traffic volumes, closer to users and/or client machines. Furthermore, the distributed architecture composed of large-scale geographically deployed edge nodes is inherently more scalable with respect to the rational cloud computing approach. Cloud-based solutions can address the non-trivial tasks related to storage, real-

time computation and optimization of the expected large amount of data. Using technologies adopted in the Internet of Things (IoT) domain [6] [7], which combine cloud and edge properties in a virtualized environment, fulfills the remaining requirements. Through resources virtualization [8], which is a common trait of recent IoT architectural solutions, it is possible to address appropriately the key data handling and communication needs of the SGs.

As an example, the advantage to decouple information resources from the physical devices producing them in the specific case of SE, allows to decouple the actual measurement rate of PMUs from the reporting rate to the state estimator in the cloud. This is of utmost important, since PMUs do not allow to change the rate and the measurement settings without stopping measurements.

Whereas recent works have proposed the adoption of the mentioned technologies, these have not completely exploited the virtualization and edge computing technologies to satisfy the SG and SE needs and they present limited experimental analyses, as highlighted in next section. On the basis of these considerations, after discussing the changes power grids are undergoing both from a structural and functional perspective, in this paper we provide the following contributions: a cloud-based SG architecture which improves the current state of the art in the domain of high-rate reporting SE, by adding flexibility to the data transmitted thanks to virtualization and adaptability thanks to edge intelligence; new adaptive algorithms for SE which exploit QoS-aware policies for context-aware data reporting, which fully exploit the key features of the proposed architecture; introduction of the possibility to adaptively change edge policies based on global information residing in the cloud; validation of the proposed architecture by emulation of a real distribution network (DN) over a real ICT infrastructure in case of a DN undergoing dynamic conditions in presence of distributed generation (DG).

The rest of the paper is organized as follows: Section 2 discusses the state of the art about architectures for SE thus giving the rationale behind this work; Section 3 presents the proposed architecture for SE detailing on the advantages of the architectural choices made; Section 4 details on the implementation of

4

the proposed architecture; after introducing the used topology and test cases, Section 5 evaluates the performance of the proposed architecture in terms of accuracy, bandwidth and latency; final conclusions are drawn in Section 6.

2. Background and Rationale

Major Smart Grid's Strategic Research Agendas such as [9] in the European Union and [10] in the United States are converging towards the same key needs for the ICT infrastructure supporting the SG: the use of the cloud as the virtual solution for the collection, parsing and distribution (at any time and at any place) of data from all nodes of the SG; the importance to leverage results from projects on topics such as the Future Internet and IoT into the energy domain; a successful abstraction and interfacing of grid components and actors so as to ensure interoperability and integration of the most diverse applications.

In the recent past, based on these key principles, a number of relevant works on the use of the cloud for the SG appeared [11, 12, 13, 14, 15, 16].

As to the domain of SE, Table 1 summarizes the most relevant works and classifies them based on architectural features and actual performance evaluation. [17] has been the first work proposing a cloud-based architecture using PMUs for SE. Sensed data are sent to virtual instances in the cloud redundantly and then retrieved for SE. A latency evaluation is also present at the end of the paper although no reference topology is given. In [18], an information centric platform for the use of PMUs in SE is presented. The aim is to decouple the data plane from the control plane to create an infrastructure able to promptly adapt to bandwidth needs. However, the bandwidth policy considered is exclusively communication network-driven and does not take into account the state of the electric network and the importance of a measured data. As a matter of fact, latency and bandwidth but not accuracy results are given. In [19], a multi-tier hierarchical network for PMU data aggregation is proposed in order to eliminate the redundant data received from the lower level PDCs but the approach does not use either the cloud or any virtualization mean so that flexi-

5

bility of the system cannot be guaranteed. Although it does not consider either the use of the cloud or the use of PMUs, in [20] local intelligence is used in order to optimize data flows and avoid overloading of data or needed computation at the control center in a low voltage network monitoring system. The latency and bandwidth of the system are also evaluated over a real power grid topology with real devices. The last work examined is [21] which despite considering SMs rather than PMUs is important because SE accuracy is evaluated.

Architectures for State Estimation (SE)	Architectural features					Performance evaluation			
	Cloud-based	Virtualization	Edge intelligence	Bandwidth adaptive policies	Use of PMUs	Bandwidth evaluation	SE accuracy evaluation	Latency evaluation	Real devices and topology
Maheshwari et al. [17]	✓	✓	X	X	✓	X	X	✓	X
Chai et al. [18]	X	X	X	✓	✓	✓	X	✓	✓
Gharavi et al. [19]	X	X	X	X	✓	✓	X	✓	X
Lu et al. [20]	X	✓	✓	X	X	✓	X	X	✓
Pau et al. [21]	✓	✓	X	X	X	X	✓	✓	✓
Proposed	✓	✓	✓	✓	✓	✓	✓	✓	✓

Table 1: Architectural features and performance evaluation in the state of the art's architectures for SE

In this paper, based on the preliminary results of [22] and [23], an architecture for SE is presented with all the characteristics listed in Table 1: a cloud-based architecture which relies on virtualization and edge intelligence of some virtualized entities in order to parse the state of the power grid by using PMUs while guaranteeing given bandwidth constraints in an adaptive manner.

In addition, the proposed architecture has been tested in terms of bandwidth, latency and accuracy of the SE on the IEEE 34-bus test network, using real PMUs prototypes and a real ICT network with virtualized entities hosted on either off-the shelf development boards or cloud instances provided by Google services.

3. Proposed architectural choices for State Estimation

SEs techniques have been designed to estimate the state of the grid, in terms of node voltages and/or branch currents, starting from the measurement

devices available on the field. These devices can be significantly heterogeneous, measuring different electrical quantities with diverse accuracies and reporting rates. The aim of the SE is to use efficiently these measures so that an accurate picture of the network status can be obtained. This information is fundamental for downstream decisions. The accuracy of the estimation results is decisive in order to have a grid safely operated.

Fig. 1 shows the layers of the virtualization framework that is implemented in the proposed solution and that exploits the key virtualization features in the IoT domain [6] [7]. This encompasses the entire communication chain from physical devices to applications. Moreover, the two options considered for the location of virtualized entities are shown. For the sake of completeness, physical devices have been split into ICT and non-ICT compliant. As a matter of fact, while some components of the SG are natively equipped with some sort of communication capabilities (e.g. PMUs), other grid components such as legacy electromechanical switches or wind turbines need an ICT interface to communicate with the rest of the network.

Physical devices are associated to a virtual counterpart which is called Virtual Object (VO). A VO is an entity that virtualizes and enriches the characteristics of one or more physical devices. Furthermore, it gives authorized users the possibility to access and request resources and functionalities in a reusable and interoperable manner, without knowing about the means and protocols that are needed to reach and retrieve information from physical objects. As an example, PMUs can manage communications up to the transport level and usually establish a connection to a single host. Using a VO allows to enrich the communication's stack capabilities and made data available for multiple hosts. VOs can be physically placed at the edge of the communication network (e.g. in the same subnetwork as the physical device) or in the cloud. Whether to choose the former or the latter solution depends on the specific use.

In the proposed system, VOs have been placed at the edge of the communication network. As a matter of fact, PMUs produce a high load of data if used at their maximum rate. This maximum rate is crucial to detect possible dy-

7

namics of a given node but reporting all the data remotely can be cumbersome rather than useful, if the network is in near-steady state. With our architectural choice, data is received at the maximum rate by the VO in the edge to accomplish tasks with strict latency constraints needing fine-grained information over time. The VO, also implements context-aware local policies in order to decide whether to send data remotely or not based on the actual state of the electric network. Another important aspect is that proper security is ensured by the fact that VOs reside in a local network. As a matter of fact, by means of the proposed solution, security critical information can be elaborated and actions can be taken locally, thus ensuring better reliability to cyber attacks by delivering remotely only necessary information and with no more than the required degree of detail. In addition, local security from external attacks can be delegated to a firewall protecting a local area, so that data exchanged between the VO and the physical device do not need heavy encryption releasing local resources from a time- and computation-consuming task. This is in line with the recommendations of IEC [24] regarding actions with stringent latency. When data has to be transmitted remotely, the use REST APIs (Representational State Transfer Application Programming Interfaces) guarantees the possibility to use de facto cyber security solutions such as SSL/TLS (Secure Sockets Layer / Transport Layer Security) as further discussed in Section 4.3.

Resources and functionalities offered by VOs are exploited by Micro Engines (MEs) which can be defined as cognitive mash-ups of VOs created in order to accomplish a certain high level task and to give a uniform interface to the application level, which is independent from the underlying resources actually available. The advantages of having a ME on the edge of the communication network are usually connected to: the smaller traffic load generated in the communication network; the diminished latency which provides a more responsive system for delay critical applications; the increased security ensured by the fact that ME functions take place behind a firewall. The advantages of a remote ME are: integration in a cloud infrastructure, which allows greatest computational and storage capabilities which elastically adapt to changes in the
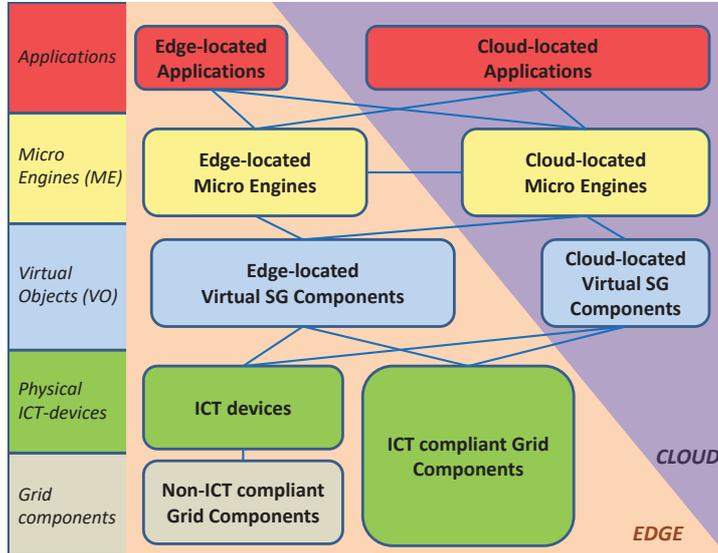
8

Figure 1: Virtualization layers used in IoT and their possible location according to the discussed architecture.

application (e.g. when more computational power is needed for parsing measurement data); global visibility (differently from local MEs), which facilitates the composition of services exploiting VOs in different remote locations.

In our case, the primary goal of the ME is SE of a given part of the electric network. In addition, the ME monitors the bandwidth from the VOs linked to it in order to decide whether to take actions to avoid an overload in terms of bandwidth or storage used. The ME, can also interface with other MEs (e.g. for multi-area SE) and with application-level parties. This allows to relieve interested parties from grid details such as the number of PMUs installed, their location or the topology of the network. To accomplish these goals, the best location for MEs is in a cloud instance, which is able to satisfy the computation needs elastically while giving the possibility of geographical replication for reachability reasons [25].

The highest layer of the proposed architecture regards applications, which accomplish high level functions leveraging on one or more underlying MEs. In this case, the location depends on the actual application considered. As an

9

example, a visualization application can easily be implemented as a web service hosted in the cloud.

4. Implementation of the proposed architectural choices for State Estimation

In this section, the details on the implementation of the architectural choices for SE proposed in the previous section are given.

4.1. Physical devices for State Estimation

PMUs are some of the most important yet most ICT resource-consuming physical devices in SG and provide synchronized phasors at a given sampling rate, typically 50-60 fps (frame per second). The $IEEE\ C.37.118.1 - 2011$ [26] and $IEEE\ C.37.118.1a - 2014$ [27] are the latest synchrophasor standards, defining synchrophasors, frequency, and rate of change of frequency (ROCOF) measurement under several operating conditions. The standard $IEEE\ C.37.118.2$ [28] defines a protocol for real-time exchange of synchronized phasor measurement data between power system equipment.

Consider a DN in which N_P PMUs are deployed. These PMUs can measure a number N_Q of electrical quantities such as voltage and current phasors and frequency at a given sampling rate. Due to how PMUs are built, the sampling rate of the physical PMU cannot be changed at runtime but it is fixed and must be set prior to data transmission start, which corresponds to the reception of command "turn on transmission of data frames" by the PMU. If a change on the rate is required, then the PMU must be stopped. Gathered measurements are sent with a GPS-synchronized timestamp to the set sink. In the proposed architecture, data is received by the corresponding VOs, which runs at the edge of the communication network as a sink for the PMU. Each PMU, creates a TCP socket with the corresponding VO and send measured data according to [28] at its maximum reporting rate.

To test in a realistic way the PMU-based architecture, real PMU prototypes implemented using National Instruments CompactRIO modular technology [29]

10

for automatic measurement system have been used in this paper. The synchronization for each PMU is achieved by means of a GPS receiver (that gives a pulse per second signal, PPS, with an accuracy of 100 ns) and each prototype can work either as a fully equipped PMU that acquires and elaborate signals or as a PMU hardware emulator. In the latter case, the PMU prototype computes synchronized measurements starting from pre-stored signals or simulates an expected measurement output. The PMU emulator is particularly suited to test dynamic operating conditions in a controlled environment so as to compare different algorithms and configurations using exactly the same signals from the network. A two-cycle P-class compliant [26] algorithm has been used throughout the tests, since its specification are independent from the reporting rate. More information on these PMU prototypes and on the chosen algorithm can be found in [30].

4.2. Virtual Objects

VOs enrich the capabilities of physical devices. For the particular case at hand, once a VO receives data from the PMU, it performs relevant processing in order to decide whether the received measurement of $q \in [1, ..., N_Q]$ must be sent to the ME or not for further processing according to the given metric. In our case, we consider the quantity q to be voltage and data is sent according to the following condition expressed for PMU $p \in [1, ..., N_P]$

$$\frac{|m_{pq}(t) - M_{pq}(t)|}{M_{pq}(t)} > T_{pq} \tag{1}$$

where $m_{pq}(t)$ is the measurement of q made by PMU p at time t, $M_{pq}(t)$ is a value representing the memory the VO has at time t with reference to the previous measurements of quantity q, and T_{pq} is the threshold enabling the transmission of the measurement q. The memory value may be computed taking into account either all of the measurements received from the PMU or only those sent by the VO. Without loss of generality we assume that $M_{pq}(t)$ is the last measurement sent by the considered VO prior to time t

$$M_{pq}(t) = m_{pq}(t - T_{lms}) \tag{2}$$

11

where T_{lms} represents the time interval between the transmission of consecutive measurements from the VO to the ME. In the following, T_{pq} is set to 1% because 1% is accuracy limit of the total vector error (TVE), and thus of the phasor amplitude error, prescribed by the standards [26], [27] for a compliant PMU under steady-state conditions. Therefore, the measurement received at time t will be sent if and only if its value differs more than 1% from the last sent measurement, thus allowing to follow both fast and slow variations. This value takes into account the high accuracy of the PMU measurements actually available in steady-state conditions [30, 31, 32, 33].

Equation 2 can also be generalized to N_M past measurements sent by the PMU, defining $M_{pq}(t)$ as

$$M_{pq}(t) = \sum_{i=1}^{N_M} w_{pq}(i) \cdot m_{pq}(t - i \cdot T_s) \tag{3}$$

where N_M is the number of considered past measurements, T_s is the sampling rate of the PMU and $w_{pq}(i)$ is the weight associated to measurement $m_{pq}(t - i \cdot T_s)$.

The differences between (2) and (3) are the following: the former focuses on how much the last sent measurement differs from the actual one, i.e., how much the measurements used for the state estimation are varying; the latter focuses on the entire flow of data generated by the PMU and averages over subsequent measurements to avoid a false positive due to the noise. In the following we make use of the formula defined in (2).

Apart from the detection of dynamics in the monitored quantity, VOs also send periodic updates in static conditions. Defining t_{pq}^{LS} as the time when the last q measurement has been sent, a new measurement data will be sent at

$$t = t_{pq}^{LS} + T_s \cdot R_{pq} \tag{4}$$

with R_{pq} representing the subsampling factor applied by the VO. For example, if in static conditions only 1 of the 50 measurements per second received from the PMU is sent by the VO, then $R_{pq} = 50$. $T_s \cdot R_{pq}$ is thus the actual sampling interval received by the state estimator.

12

In our testbed, each VO is hosted in a CapeDwarf installation in a local server, which is an open source implementation of the PaaS (Platform as a Service) Google App Engine (GAE). Indeed, the VOs are processes that run locally on the edge of the network but could be moved into the cloud if the latency allows for and more computational power is needed. Accordingly, a VO process could be moved from CapeDwarf to GAE in the cloud.

4.3. Micro Engines

Using the measurement data received from VOs along with forecast based on historical information of the loads and generators (the so-called pseudomeasurements, required for the observability of the system), an estimation of the network state is computed by the ME, in terms of voltage phasor at each node and branch-current phasor for each branch.

In this paper, the case of a Distribution System State Estimation (DSSE) is considered. Compared to transmission systems, monitoring systems for DNs are more challenging for a number of reasons. Distribution grids are characterized, among others, by a very large number of nodes, with varied load, and different voltage levels. They have not only three-phase but also two/single-phase configurations and several types of non-symmetrical loads, leading to different degrees of unbalance, while the presence of DER and DG can be very significant (see [34] and [35] for a deeper discussion on the DSSE challenges). Several methods for the DSSE have been presented in the literature, mostly based on a weighted least squares (WLSs) formulation. Recently, a new branch current DSSE (BC-DSSE) was proposed [36], proving the same accuracy, but faster execution, as node voltage DSSE. This is the method applied in this paper for the estimation of the operating conditions using PMU measurements also including classic χ-squares and normalized residuals tests for bad data detection and identification [37].

When a new measurement is received by the ME, a new estimation is triggered for the timestamp indicated in the received packet. If no measurements with the same timestamp are found for the other monitored nodes, the most

13

recent measurements are used for the estimation. It is also possible that new

₃₂₀ measurements with the same timestamp are received once DSSE has already been performed. In this case, the ME will compute again the state of the network from the measurements available at that moment in order to obtain a more accurate estimation to be associated with the corresponding timestamp. The DSSE is designed to be fast and the refinement of the estimation is auto-

₃₂₅ matic. DSSE output does not depend on previous estimates to avoid the risk of poor performance under dynamic conditions. While this can appear inefficient from a computational point of view, it also offers an important advantage to applications, which can start to benefit from the last system SE even before all measurements have been received. Also, notice that the implemented solution

₃₃₀ allows to parse the state of the network even when some data packets are still missing or lost due to packet loss. A detailed discussion on the impact of the number and type of PMU measurements on DSSE can be found in [38].

In this paper the DSSE routine is implemented in a ME in the cloud, so that the necessary computational and storage elasticity required by the proposed

₃₃₅ system is provided. As a matter of fact MEs are hosted in Google Cloud App Engines of class $B8$ which run in a 4.8 GHz processor, but can clone themselves in case the instantiated resources are not sufficient. If multiple instances are created to face the need for more computational resources, this process is not perceived from the outside and consistency is managed internally. In particular,

₃₄₀ when a new measurement is received, data are first of all written to the shared memcache and only then, data of interest to accomplish the state estimation are read from the same memcache which is shared among instances. This guarantees consistency in the case multiple instances are operating in parallel on a shared set of data. Currently, Google has deployed more than 15 data centers across the

₃₄₅ world and more are expected for the future. However, while Google cloud has been used in the paper, the proposed architecture can easily use cloud services from other cloud providers (e.g. Amazon).

VOs act as interfaces between the local subnetwork and the cloud. Communication between the VO in the edge and ME in the cloud is accomplished using

14

REST APIs. This ensures interoperability and a protocol agnostic interfacing between the VOs and MEs. Moreover, using HTTP gives the possibility to implement security protocols on top of HTTP such as TLS/SSL, which cannot be exploited in direct PMU-to-state estimator communication since PMUs communication capabilities stop at the transport level. While this will add latency to the data transmission, we expect it to be limited for purposes with no strict time constraints which, in the proposed solution, can be accomplished locally as already described in Section 3.

In the proposed system, communication takes place in $PUSH$ mode, which means that data is sent automatically from the VO to the ME through HTTP POSTs based on the forwarding policy given in the previous subsection. Data at the ME is parsed, processed, stored and, in addition, sent to an application for the visualization to operators of the estimated state of the network.

In our specific case, we decided to give also an additional task to the ME receiving PMU data, so as to show context-awareness [39] at the ME level which is based on global rather than local information as in the VO case. Specifically, the ME has to dynamically update the R_{pq} value at the VO in order to guarantee that the average bandwidth utilization across the network tends to an objective value D_{obj}, which is selected by the SG operator based on the internal quality targets and network configuration. This update is communicated to the VO piggybacking on the answer to the HTTP POST used by the VO to send the measurements. Here, D_{obj} is expressed as the amount of data received and stored in the ME normalized over time (i.e. [B/s]).

We then compute the average rate as the cumulative data generated from system's start until time t by any PMUs p and for any quantity q, normalized over time:

$$D_{avg}(t) = \frac{\sum_{p,q,t_x} \left[D_{pq}(t_x)\right]}{t} \tag{5}$$

where for notation convenience, with t_x we refer to each instant of time at which a VO has sent measurement data. $D_{avg}(t)$ can be implemented with two counters for: the duration time since system start; the amount of overall data

15

received. Each time the ME receives a new measurement, the value R_{pq} at time t is updated as

$$R_{pq}(t) = \left\lceil (1 + \Delta(t)) \cdot R_{pq}(t_{pq}^{LS}) \right\rceil \tag{6}$$

where

$$\Delta(t) = \alpha \cdot \frac{D_{avg}(t) - D_{obj}}{D_{avg}(t) + D_{obj}} \tag{7}$$

and $0 < \alpha \leq 1$ is a parameter which modules the sensitivity. Here $\alpha = 0.1$ is chosen. Therefore, the updated value $R_{pq}(t)$ is the value $R_{pq}(t_{pq}^{LS})$ increased or decreased by a factor which depends on the objective bandwidth and the overall history of data received up to time t by the ME (as described in Equations 5 and 7).

4.4. Applications

In this paper, for the sake of simplicity, the considered final application is a visualization system, which shows the state estimation results over time in the various monitored and non-monitored nodes of the distribution network.
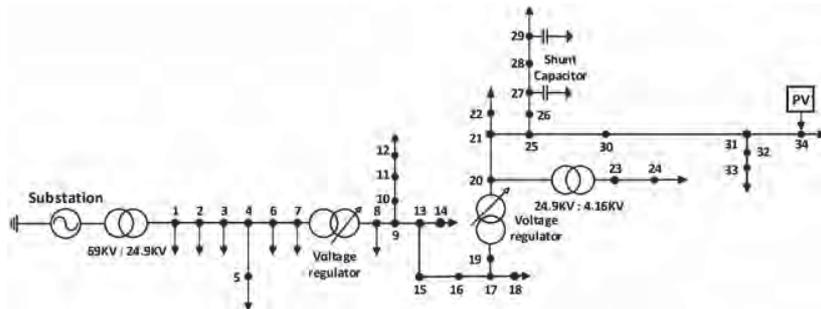
5. Performance evaluation

Figure 2: Test system: IEEE 34 Node Test feeder

16

The proposed architecture has been validated in a number of test scenarios. In particular, the results discussed in the following have been obtained on a three-phase test system based on the IEEE 34-bus test feeder, derived from an actual system in Arizona and with the following characteristics: long lines and a quite loaded system with two in-line on-load tap changers and two capacitor banks. This system is of particular interest since it has proven to be very sensitive to the impact of DG [40]. Fig. 2 reports the network diagram. Details about line parameters (line lengths and impedances), nominal loads and generators can be found in [41].

A photovoltaic plant of 2MW capacity is installed at node 34. The PV plant is modeled as a PQ injection (where the active, P, and reactive, Q, powers are specified) and the bandwidth of the PQ injection is limited by a second order transfer function tuned to mimic the behavior of the interconnecting filter of the plant converter. Several irradiation variation conditions have been considered to create the test cases used in this work.

The system is simulated using Opal-RT. Opal-RT is a digital real time simulator for electric and electro-mechanical systems. Through the use of analog and digital IO it is possible to interface the simulator with external hardware (Hardware In the Loop) so that the measurement devices operate as they were connected to a real electric system. The PMU prototypes presented in Section 4.1 can be connected to the Opal-RT output signals thus allowing to emulate realistically the acquisition stage and the whole measurement system.

The presented results have been obtained using three PMUs that measure the voltage phasors at bus number 3, 9 and 31 of the network (Fig. 2). Four test operating conditions have been considered, each running $3 \cdot 10^2$ times:

- Case 1: PV plant switches on at $t = 25.3$ s with a real power $P = 2$ MW.

- Case 2: PV plant switches on at $t = 24.3$ s with a real power $P = 2$ MW and switches off at after 2 s.

17

- Case 3: PV plant varies fast the injected power in the range [1 MW −
  1.5 MW] during ten seconds from $t = 20$ s.

- Case 4: PV plant varies slowly the injected power from zero to 2 MW as
  a ramp starting at $t = 0.3$ s and ending at $t = 10.3$ s.

The test cases can be considered as representative of possible PV behavior
and are sufficient to show the outcomes of the proposed approach. In general,
many factors impact on the operative conditions of the network (weather con-
ditions, season, time of the day, etc.) but the considerations drawn from the
above cases and discussed in the following have general significance.

5.2. Accuracy evaluation

Two accuracy indices have been used to compare the estimates given by a
fixed-step estimator with those computed by the described variable rate system.

The absolute average of node voltage magnitude estimation error (referred
to with MAVE in the following)

$$MAVE_\phi = \frac{1}{NT} \sum_{i=0}^{T-1} \sum_{n=1}^{N} |V_{\phi,VR}(n,i) - V_{\phi,FR}(n,i)| \qquad (8)$$

where $V_{\phi,FR}(n,i)$ is the per-unit estimate of the voltage magnitude of node n
of phase ϕ at the instant iT_s obtained with a fixed maximum reporting rate
50 fps per PMU and $T_s = 20$ ms, and $V_{\phi,VR}(n,i)$ is the estimate performed
by means of the variable DSSE obtained by the proposed estimation system.
N and T are, respectively, the number of nodes and the number of considered
time instants. Since the ME computes the DSSE only for the measurement
timestamps received from VOs (at T_{VR} instants), for the aim of comparison
the estimation of the proposed system is used for comparison with subsequent
timestamps until a new estimation is available. This choice gives a worst case
for the comparison since more elaborated interpolation methods could be used.
Because of the variable rate of the proposed architecture, another comparison
index has also been used. The dynamic time warping distance (DTW) measures
the similarity between two temporal sequences which may vary in time or speed

18

and is often used for pattern comparison [42]. Each point of the sequence is represented by the N-dimensional point of the estimated amplitude voltages and the based considered distance is the euclidean one. The DTW is thus a cumulative index for nodes and time instants and has been normalized with respect to the duration to give a better intuition.

Table 2 reports the accuracy results for the first phase (phase a) of the three-phase network. From the results it is clear how the difference between the implemented policy and the classic use at the maximum rate is limited in terms of accuracy. Fig. 3 shows an example of the estimation dynamics during four seconds of the test for Case 3[1]. The graph confirms intuitively that, even in an unmonitored node the proposed architecture allows to follow the evolution, due to the triggering effect of the VO of monitored nodes.
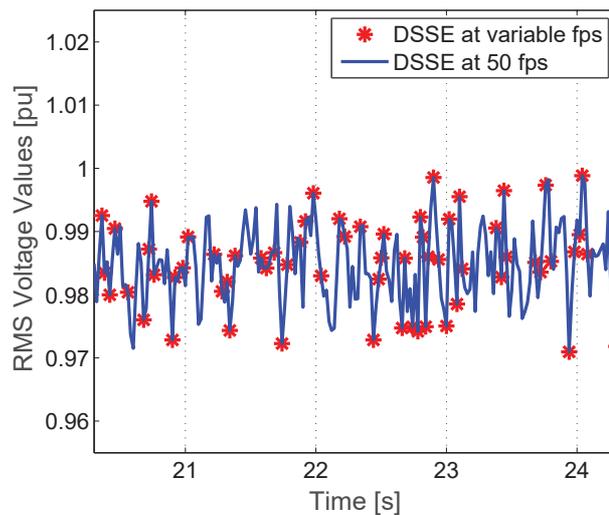

Figure 3: Case 3: voltage magnitude estimation at node 25 with fixed and variable rate DSSE.

[1]the graph for a constant reporting rate of 50 fps is plotted by a continuous line for the sake of simplicity

5.3. Bandwidth evaluation

In order to compute the performance in terms of bandwidth reduction, the Bandwidth Saving Ratio (BSR) is defined as the ratio between the amount of data sent to the ME by all considered VOs over the overall data sent in the full-rate case of a constant 50 fps reporting rate.

In our implemented scenario, the size of an HTTP POST sent to the ME is constant and equal to $D_{pq} = 424$ Bytes (including payload and headers of the communication stack's layers). As to the D_{obj}, it has been set to 4.24 kB/s, which brings to $D_{obj}/D_{pq} = 10$ fps. In the case in which all the produced data are transmitted (our proposed adaptive transmission algorithm is not used), $D_{obj}/D_{pq} = 150$ fps, since three PMUs are used. Therefore, asymptotically $BSR = 0.0667$ for the considered scenarios, which is a considerable reduction in light of the excellent accuracy results and considering that in the full-rate case the data rate would be $D_{obj} = 63.6$ kB/s.

Table 2 shows the BSR for the 4 test cases. It can be seen that the bigger the dynamic of the system, the bigger the BSR value. The difference found among the 4 tested cases is due to the finite length of the tests. In the asymptotic case, the value of BSR in all the 4 cases would be the same if the same D_{obj} is chosen. Therefore, the fact that the BSR values are close to the asymptotic value tells us that the bandwidth policy used is properly working despite the limited observation window used. This can be seen also in Fig. 4, showing $D_{avg}(t)$ normalized by D_{pq} in the 4 test cases.

All the 4 cases present an initial transient and then converge to the D_{obj}/D_{pq} value unless a dynamic is present. Let us call DS_{ratio} the number of measurements sent in dynamic state over those sent in steady state according to R_{pq}. In the test case 1, $DS_{ratio} = 0.0089$. Therefore, no relevant variations triggering dynamic measurements transmissions are present except when the PV plant switches on. The slow convergence is due to the small value of α. Test case 2 and 4 report a DS_{ratio} of 0.0175 and 0.0233 respectively. It can be seen that in test case 2 there is a slight but sudden increase of $D_{avg}(t)/D_{pq}$ when the PV plant is switched on and off. The dynamic of test case 4 is slow and therefore
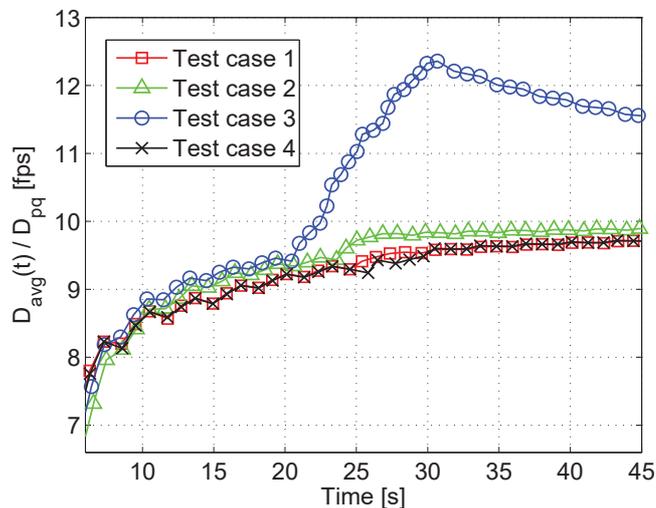
20

Figure 4: Average data rate normalized to the HTTP packet size, reported as a function of time for the four test cases analyzed.

the interval between two measurement transmissions triggered by the dynamic threshold condition is probably close to the interval $T_s \cdot R_{pq}$ (remember that each time the VO sends a measurement to the ME, the value t_{pq}^{LS} is updated). This results in the absence of sudden changes of the derivative of the curve.

485   The last and more significant result is that obtained for test case 3. In this case, $DS_{ratio} = 0.2327$. As it can be seen from the plot, $D_{avg}(t)/D_{pq}$ starts increasing around 20 s as it would be expected, but as soon as the dynamic part of the electric signal ends, $D_{avg}(t)/D_{pq}$ starts converging back to the objective value thanks to the implemented policy.

490   The above discussion shows how the proposed policy allows meeting the bandwidth target while keeping the accuracy high for the DSSE under dynamic conditions. It is interesting that the same bandwidth could be obtained by exploiting the proposed architecture to subsample the PMU input streams at VO level. The adopted D_{obj} is reached by retaining a packet every 15, but with respect to Table 2 results, both MASE and DWT largely increase: for instance, MASE increases by a factor of 2.2 to 97.1, depending on the test case.

21

Test case	MAVE [p.u.]	DTW [p.u./s]	BSR [B/B]
Case 1	5.5e-06	0.0016	0.0644
Case 2	8.0e-06	0.0021	0.0658
Case 3	4.2e-04	0.1121	0.0759
Case 4	2.6e-04	0.0448	0.0628

Table 2: Accuracy and bandwidth saving comparison between the variable rate proposed estimation and the fixed rate counterpart

5.4. Latency evaluation

In this subsection, the results obtained for the 4 test cases in terms of latency are analyzed. Fig. 5 shows the communication paths considered for the communication latency values given in Table 3. Notice that both the PMUs and the VOs are GPS synchronized. Therefore gathered latency measurements have an accuracy well below the millisecond, even though we restrict to the millisecond precision according to the PMU timestamp format. All the values are obtained over a number of packets in the order of $5 \cdot 10^2$.

PMU-to-VO represents the interval between the timestamp of a measurement and the moment when that measurement arrives to the VO in the same subnetwork. As it can be seen from Table 3 and according to the discussion in the previous sections, placing the VO locally results in a small latency, as it can be noticed comparing the minimum, maximum and especially the average values. Moreover, notice that the values for PMU-to-VO also contain the intrinsic delay introduced by the PMU measurement (PMU measurement latency, as defined in [26]), which, in the specific tests, accounts for 30 ms. Examples of measured PMU reporting latencies are reported in [43].

VO-RTT represents the time elapsed from the moment a certain measurement is received by the VO to the moment the same VO receives an HTTP answer to its HTTP POST containing the new $R_{pq}(t)$ value. VO-RTT also includes the ME processing interval, which is the time elapsed from the moment

the ME in the cloud starts processing the HTTP POST received, to the moment before the HTTP answer is sent. The ME processing interval includes: updating the necessary counters and measurement information received from the PMUs; gathering all the latest information about the distribution network; parsing the network state estimation. The reason for computing both VO-RTT and ME processing is that the ME is not GPS-synchronized and has its own clock synchronized via NTP protocol and thus with a time accuracy well-above the one of GPS-synchronized devices. Moreover, knowing ME processing allows to subtract it to the VO-RTT value in order to know the sole influence of the network, the processing delay and subsequently the time elapsed from the PMU timestamp to the moment the DSSE can be used by applications.

As it can be seen both the VO-RTT and the ME processing value are considerably variable. In the former case, this is associated to the use of the public Internet which have an unpredictable behaviour and for which there is no quality guarantee. In the latter case, the high variability is due to: the cloud resources used, which are free of charge but without any Service Level Agreement (SLA); the time needed to store and retrieve the measurement data, which is variable and also SLA-free; the time needed for the iterative algorithm estimating the state of the distribution network, for which the number of iterations depends on the values measured. Nevertheless, as it can be seen from the overall results, the worst overall latency among the 4 cases is 211.9 ms. We can thus assume that applications will be able to exploit the values of the state estimation well-below 500 ms, which is the $TT2$ performance class threshold given by IEC for operator commands [44]. The obtained results are further supported by the fact that this is a worst case scenario for the cloud entities' location, since in a real deployment the proximity between local VOs and MEs in the cloud is expected to be finer than the one obtained using the cloud from Google.

	Latency	MIN	MAX	MEAN
TEST CASE 1	PMU-to-VO	38.0 ms	49.0 ms	39.3 ms
	VO-RTT	132.0 ms	303.0 ms	163.3 ms
	ME processing	44.3 ms	209.9 ms	71.5 ms
	Overall Latency	171.0 ms	343.0 ms	202.6 ms
TEST CASE 2	PMU-to-VO	37.0 ms	47.0 ms	37.4 ms
	VO-RTT	134.0 ms	349.0 ms	171.7 ms
	ME processing	42.7 ms	195.4 ms	76.0 ms
	Overall Latency	171.0 ms	386.0 ms	209.1 ms
TEST CASE 3	PMU-to-VO	38.0 ms	48.0 ms	39.14 ms
	VO-RTT	133.0 ms	318.0 ms	167.5 ms
	ME processing	46.6 ms	221.4 ms	75.1 ms
	Overall Latency	172.0 ms	357.0 ms	206.7 ms
TEST CASE 4	PMU-to-VO	41.0 ms	48.0 ms	41.6 ms
	VO-RTT	131.0 ms	348.0 ms	170.3 ms
	ME processing	46.3 ms	160.5 ms	75.6 ms
	Overall Latency	174.0 ms	389.0 ms	211.9 ms

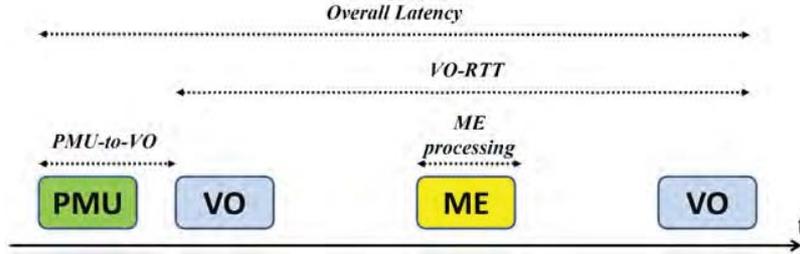Table 3: Minimum, maximum and average latency due to various parts of the system, according to Fig. 5

Figure 5: Legend for the latencies shown in Table 3

6. Conclusions

In the past, the evaluation of the state of the electric network was normally performed at a slow rate, appropriate for steady state operating conditions. The increasing presence of distributed energy resources jeopardizes the safe management of the power grid. This has required the introduction of bandwidth-consuming sensing devices such as PMUs, so that dynamics can be promptly detected by the monitoring system in order to properly react. To this goal, the paper discusses an IoT architecture for the effective monitoring of the power grid and presents a practical implementation for state estimation in active distribution systems which is bandwidth- and accuracy-efficient. As a matter of fact, the virtualization capability of IoT, the advantages of edge intelligence and the computation power and flexibility of the cloud have been exploited to obtain a flexible monitoring system built on a PMU-based wide area measurement system. The analysis of the performance of the proposed monitoring architecture have been conducted on an active network derived from an actual system in Arizona modeled as an IEEE 34-bus test feeder by using Opal-RT simulator and real PMU-prototypes. The results show that the QoS application-specific requirements in terms of latency and accuracy of the estimation can be ensured with a significant reduction in the bandwidth required with respect to the case of PMUs with a full-operating estimation frequency.

25

References

[1] W. Wang, Y. Xu, M. Khanna, A survey on the communication architectures in smart grid, Computer Networks 55 (15) (2011) 3604 – 3629. `doi:http://dx.doi.org/10.1016/j.comnet.2011.07.010`.

[2] V. C. Gungor, D. Sahin, T. Kocak, S. Ergut, C. Buccella, C. Cecati, G. P. Hancke, A survey on smart grid potential applications and communication requirements, IEEE Transactions on Industrial Informatics 9 (1) (2013) 28–42. `doi:10.1109/TII.2012.2218253`.

[3] M. Kuzlu, M. Pipattanasomporn, S. Rahman, Communication network requirements for major smart grid applications in HAN, NAN and WAN, Computer Networks 67 (2014) 74 – 88. `doi:http://dx.doi.org/10.1016/j.comnet.2014.03.029`.

[4] D. Mocanu, E. Mocanu, P. Nguyen, M. Gibescu, A. Liotta, Big iot data mining for real-time energy disaggregation in buildings, in: IEEE International Conference on Systems, Man, and Cybernetics, 2016, pp. 9–12.

[5] F. Bouhafs, M. Mackay, M. Merabti, Links to the future: Communication requirements and challenges in the smart grid, IEEE Power and Energy Magazine 10 (1) (2012) 24–32. `doi:10.1109/MPE.2011.943134`.

[6] L. Atzori, A. Iera, G. Morabito, The internet of things: A survey, Computer Networks 54 (15) (2010) 2787 – 2805. `doi:http://dx.doi.org/10.1016/j.comnet.2010.05.010`.

[7] I. Farris, R. Girau, L. Militano, M. Nitti, L. Atzori, A. Iera, G. Morabito, Social virtual objects in the edge cloud, IEEE Cloud Computing 2 (6) (2015) 20–28. `doi:10.1109/MCC.2015.116`.

[8] M. Nitti, V. Pilloni, G. Colistra, L. Atzori, The virtual object as a major element of the internet of things: A survey, IEEE Communications Surveys Tutorials 18 (2) (2016) 1228–1240. `doi:10.1109/COMST.2015.2498304`.

[9] Strategic research agenda for europe's electricity networks of the future. URL `http://www.smartgrids.eu/documents/sra2035.pdf`

[10] NIST, Framework and roadmap for smart grid interoperability standards, release 3.0. URL `www.nist.gov/smartgrid/upload/NIST-SP-1108r3.pdf`

[11] S. Rusitschka, K. Eger, C. Gerdes, Smart grid data cloud: A model for utilizing cloud computing in the smart grid domain, in: First IEEE International Conference on Smart Grid Communications (SmartGridComm), 2010, pp. 483–488. `doi:10.1109/SMARTGRID.2010.5622089`.

[12] X. Fang, S. Misra, G. Xue, D. Yang, Managing smart grid information in the cloud: opportunities, model, and applications, IEEE Network 26 (4) (2012) 32–38. `doi:10.1109/MNET.2012.6246750`.

[13] M. Yigit, V. C. Gungor, S. Baktir, Cloud computing for smart grid applications, Computer Networks 70 (2014) 312 – 329. `doi:10.1016/j.comnet.2014.06.007`.

[14] S. Bera, S. Misra, J. J. P. C. Rodrigues, Cloud computing applications for smart grid: A survey, IEEE Transactions on Parallel and Distributed Systems 26 (5) (2015) 1477–1494. `doi:10.1109/TPDS.2014.2321378`.

[15] H. Kim, Y. J. Kim, K. Yang, M. Thottan, Cloud-based demand response for smart grid: Architecture and distributed algorithms, in: IEEE International Conference on Smart Grid Communications (SmartGridComm), 2011, pp. 398–403. `doi:10.1109/SmartGridComm.2011.6102355`.

27

[16] A. Liakopoulos, A. Zafeiropoulos, P. Gouvas, Smart grid evolution through the exploitation of future internet networking concepts, in: IEEE International Conference on Industrial Technology (ICIT), 2012, pp. 415–420. `doi:10.1109/ICIT.2012.6209973`.

[17] K. Maheshwari, M. Lim, L. Wang, K. Birman, R. van Renesse, Toward a reliable, secure and fault tolerant smart grid state estimation in the cloud, in: IEEE PES Innovative Smart Grid Technologies (ISGT), 2013, pp. 1–6. `doi:10.1109/ISGT.2013.6497831`.

[18] W. K. Chai, N. Wang, K. V. Katsaros, G. Kamel, G. Pavlou, S. Melis, M. Hoefling, B. Vieira, P. Romano, S. Sarri, T. T. Tesfay, B. Yang, F. Heimgaertner, M. Pignati, M. Paolone, M. Menth, E. Poll, M. Mampaey, H. H. I. Bontius, C. Develder, An information-centric communication infrastructure for real-time state estimation of active distribution networks, IEEE Transactions on Smart Grid 6 (4) (2015) 2134–2146. `doi:10.1109/TSG.2015.2398840`.

[19] H. Gharavi, B. Hu, Scalable synchrophasors communication network design and implementation for real-time distributed generation grid, IEEE Transactions on Smart Grid 6 (5) (2015) 2539–2550. `doi:10.1109/TSG.2015.2424196`.

[20] S. Lu, S. Repo, D. D. Giustina, F. A. C. Figuerola, A. Lf, M. Pikkarainen, Real-time low voltage network monitoring - ict architecture and field test experience, IEEE Transactions on Smart Grid 6 (4) (2015) 2002–2012. `doi:10.1109/TSG.2014.2371853`.

[21] M. Pau, E. Patti, L. Barbierato, A. Estebsari, E. Pons, F. Ponci, A. Monti, Low voltage system state estimation based on smart metering infrastructure, in: IEEE international workshop on applied measurements for power systems (AMPS 2016), 2016.

[22] A. Meloni, P. A. Pegoraro, A. Atzori, S. Sulis, An iot architecture for wide

28

area measurement systems: a virtualized pmu based approach, in: IEEE International Energy Conference (ENERGYCON), 2016, pp. 1–6.

[23] A. Meloni, P. A. Pegoraro, A. Atzori, P. Castello, S. Sulis, Iot cloud-based distribution system state estimation: Virtual objects and context-awareness, in: IEEE International Conference on Communications (ICC), 2016, pp. 1–6.

[24] IEC TC57, Power systems management and associated information exchange data and communications security. Part 6: Security for IEC 61850 (2007).

[25] L. Pamies-Juarez, P. Garca-Lpez, M. Snchez-Artigas, B. Herrera, Towards the design of optimal data redundancy schemes for heterogeneous cloud storage infrastructures, Computer Networks 55 (5) (2011) 1100 – 1113. doi:http://dx.doi.org/10.1016/j.comnet.2010.11.004.

[26] IEEE standard for synchrophasor measurements for power systems (Dec. 2011). doi:10.1109/IEEESTD.2011.6111219.

[27] IEEE standard for synchrophasor measurements for power systems – amendment 1: Modification of selected performance requirements (Apr. 2014). doi:10.1109/IEEESTD.2014.6804630.

[28] IEEE standard for synchrophasor data transfer for power systems (Dec. 2011). doi:10.1109/IEEESTD.2011.6111222.

[29] N. Instruments, Compactrio platform.
URL http://www.ni.com/compactrio/

[30] P. Castello, J. Liu, C. Muscas, P. A. Pegoraro, F. Ponci, A. Monti, A fast and accurate pmu algorithm for P+M class measurement of synchrophasor and frequency, IEEE Transactions on Instrumentation and Measurement 63 (12) (2014) 2837–2845. doi:10.1109/TIM.2014.2323137.

[31] D. Belega, D. Macii, D. Petri, Fast synchrophasor estimation by means of frequency-domain and time-domain algorithms, IEEE Transactions on Instrumentation and Measurement 63 (2) (2014) 388–401. `doi:10.1109/TIM.2013.2279000`.

[32] A. J. Roscoe, I. F. Abdulhadi, G. M. Burt, P and M class phasor measurement unit algorithms using adaptive cascaded filters, IEEE Transactions on Power Delivery 28 (3) (2013) 1447–1459. `doi:10.1109/TPWRD.2013.2238256`.

[33] A. Monti, C. Muscas, F. Ponci, Phasor Measurement Units and Wide Area Monitoring Systems, 1st Edition, Academic Press, Elsevier, 2016.

[34] D. D. Giustina, M. Pau, P. A. Pegoraro, F. Ponci, S. Sulis, Electrical distribution system state estimation: measurement issues and challenges, IEEE Instrumentation Measurement Magazine 17 (6) (2014) 36–42. `doi:10.1109/MIM.2014.6968929`.

[35] A. Primadianto, C. N. Lu, A review on distribution system state estimation, IEEE Transactions on Power Systems PP (99) (2017) 1–1. `doi:10.1109/TPWRS.2016.2632156`.

[36] M. Pau, P. A. Pegoraro, S. Sulis, Efficient branch-current-based distribution system state estimation including synchronized measurements, IEEE Transactions on Instrumentation and Measurement 62 (9) (2013) 2419–2429. `doi:10.1109/TIM.2013.2272397`.

[37] A. Abur, A. G. Expòsito, Power System State Estimation. Theory and Implementation., Marcel Dekker, New York, 2004.

[38] C. Muscas, M. Pau, P. A. Pegoraro, S. Sulis, Uncertainty of voltage profile in pmu-based distribution system state estimation, IEEE Transactions on Instrumentation and Measurement 65 (5) (2016) 988–998. `doi:10.1109/TIM.2015.2494619`.

[39] M. Donohoe, B. Jennings, S. Balasubramaniam, Context-awareness and the smart grid: Requirements and challenges, Computer Networks 79 (2015) 263 – 282. doi:http://dx.doi.org/10.1016/j.comnet.2015.01.007.

[40] J. A. Silva, H. B. Funmilayo, K. L. Bulter-Purry, Impact of distributed generation on the ieee 34 node radial test feeder with overcurrent protection, in: 39th North American Power Symposium (NAPS), 2007, pp. 49–57. doi:10.1109/NAPS.2007.4402285.

[41] IEEE PES Distribution Test feeders.
URL http://ewh.ieee.org/soc/pes/dsacom/testfeeders/

[42] H. Sakoe, S. Chiba, Dynamic programming algorithm optimization for spoken word recognition, IEEE Transactions on Acoustics, Speech, and Signal Processing 26 (1) (1978) 43–49. doi:10.1109/TASSP.1978.1163055.

[43] P. Castello, C. Muscas, P. A. Pegoraro, S. Sulis, Automated test system to assess reporting latency in pmus, in: IEEE International Instrumentation and Measurement Technology Conference (I2MTC), 2016, pp. 127–132.

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