# Monitoring Traffic Optimization in Smart Grid

Guillaume Habault, *Member, IEEE,* Maxime Lefrançois,
François Lemercier, *Member, IEEE,* Nicolas Montavont, *Member, IEEE,*
Periklis Chatzimisios, *Senior Member, IEEE,* and Georgios Z. Papadopoulos, *Member, IEEE*

***Abstract*—The emergence of micro-generation systems steadily increases and it raises concerns regarding their impact on the power grid. It is therefore crucial to efficiently integrate them into future Smart Grid Architectures, as there is not any standard way to monitor production units. Moreover, current data collection systems are simple and do not consider their impact on Local Area Networks. This article presents a set of proposed mechanisms that reduces the monitoring traffic, while offering management flexibility on large-scale systems. This study is illustrated with measurements performed on a small grid, and it shows that, for monitoring a PV production, both one minute and one second intervals provides the same production estimation, while significantly decreasing the associated traffic. It can be reduced even more by aggregating several measurements during a given period before sending them, and by using specific mechanisms to ensure reliability. This experiment also helps authors identify best practices for monitoring different equipment based on their behaviors.**

***Index Terms*—Monitoring, Semantic, Photovoltaic panel, Smart Grid, Nanogrid, Monitoring traffic, Constrained nodes.**

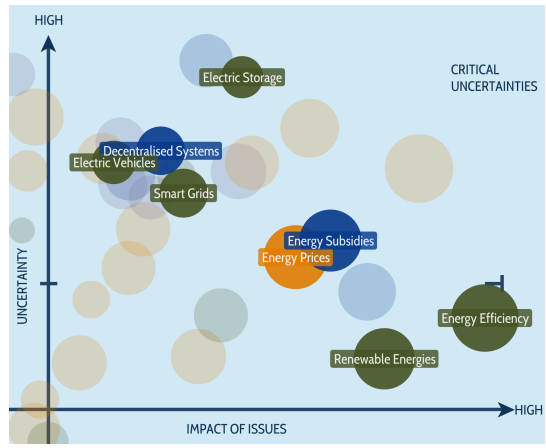



Fig. 1: Focus from the World Energy Issues Monitor in 2016.

## I. Introduction

Nowadays, there are number of notable challenges taking place in the energy domain. As stated in Eurelectric survey [1], the grid requires to adapt to the shift away from fossil energies and the emergence of less-reliable renewable micro-generation. Indeed, Figure 1 provides the scope on certain open energy issues and emphasizes their levels of criticality [2]. We noticed that some issues still requires some actions to especially solve Energy efficiency and Renewable energies integration, which according to this figure, have the most significant impact on the energy sector [1].

With the arrival of Electric Vehicles (EVs) and the growing number of electric appliances, the need for electricity and proper management [2] is steadily increasing. However, it poses certain challenges as current distribution networks will not be sufficient to transport the required amount of electricity, especially during peak hours. Their enhancement is not envisioned for a near future due to the associated high cost. Therefore, it is particularly recommended to consider the use of energy storage or employ local production in order to assist the distribution network and avoid any outage. As a result, in the following years the development of tools and systems that facilitate use of local production and that optimize consumption management will become essential.

Nowadays, utilities do not subsidize monitoring equipment of local production systems, as they do not directly benefit from them. However, the unpredictable nature of renewable sources makes it difficult for them to estimate consumption of sites equipped with such production units. Therefore, considering the popularization of these installations, utilities would gain from having access to production information for a better grid planning and management. This situation leads to the installation of non-standard production meters or monitoring equipment at users premises. Each industrial player having its own device, which often does not provide communication capabilities, while potentially limiting the information to an LCD screen. Properly monitoring these installations and connecting such devices to the grid, would be a significant advantage for utilities. Nevertheless, there is currently no standard to interconnect these devices with the Smart Grid and to inform on the production capabilities of these sites.

Meanwhile, smart appliances are being more embedded with communication and control capabilities offering users the opportunity to automate them. Such management and automation could even go further, and be related to the actual production, by linking them with production information.

G. Habault, N. Montavont, G. Z. Papadopoulos are with IRISA, IMT Atlantique Rennes Campus, France.

M. LeFrançois is with Ecole Nationale Supérieure des Mines, Laboratoire Hubert Curien, Saint-Étienne, France.

P. Chatzimisios is with the CCSN Research Lab and the Department of Informatics, Alexander TEI of Thessaloniki, Greece.

F. Lemercier is with IRISA, and Itron, France.

Corresponding author: Guillaume Habault (email: guillaume.habault@imt-atlantique.fr)

Manuscript received 06/02/2017; revised 12/06/2017.

[1]http://www.eurelectric.org/media/249736/power-statistics-and-trends-the-five-dimentions-of-the-energy-union-lr-2015-030-0641-01-e.pdf

[2]http://www.worldenergy.org/wp-content/uploads/2016/10/World-Energy-Resources_FullReport_2016.pdf

According to [3], future Smart Grid Architectures will probably rely on both Advanced Metering Infrastructure (AMI) (used by "consumption" smart meters) and Internet (used by smart appliances). Such future energy architecture will be composed of several Management Systems (MSs). These devices will manage given sites, areas or groups by collecting and analyzing information coming from managed devices as well as corresponding AMI's smart meters and external services available via Internet. For instance, MSs could utilize all these collected data to control consuming devices and decide how production will be used (stored or directly consumed) in order to reach an equilibrium between consumption and production. As a consequence and in order to enable an efficient management, MSs require data from all devices that consume, produce or store energy.

In this article, we investigate methods that lower the impact of continuously monitoring equipment. We propose a set of mechanisms that allow efficient equipment monitoring while reducing corresponding traffic.

The rest of the article is organized as follows. Section II presents an overview of the state of the art. Then, we present the context in which this study lies and the considered hypotheses. Section IV lingers over the five mechanisms proposed in this study to limit 1) the monitoring payload; 2) the monitoring traffic; 3) the effect of network losses, while maintaining high quality of data as well as providing flexibility in the process. The performances evaluation of the proposed solutions are illustrated with a photovoltaic (PV) panel monitoring testbed. Finally, Section VI concludes our article and provides some thoughts for future work.

## II. Related Work

Local energy production and the usage of renewable sources for electricity production are recognized trends in the evolution of the grid, especially in smart cities [4]. However, it comes with a number of issues, such as how to integrate these new energy sources in the power grid or how to store the energy. The emergence of EVs is also introducing new problems for the grid, in terms of peak consumption, or mobile battery [5]. In order to mitigate these issues, a Demand Response management system is often proposed in the literature to balance the load between production and consumption [6].

All these applications require a dedicated network architecture. Such a network is dense by nature, and composed of a wide range of communication technologies [7]. Because this network requires to run different applications, it needs to provide different levels of quality of service, while controlling the energy consumption of sensitive parts of the network [4]. Deng et al. [6] insist on the bi-directional feature of the network to enable communication between consumption and production units and the infrastructure without a pre-defined hierarchy. On one side, the network should provide monitoring and reporting from the consumption and production units. On the other side, incentives or consumption policies help in controlling the load remotely.

Regarding the communication aspects, Ayaz et. al. [8] propose a non-orthogonal multiple access concept for Smart grid communication to improve spectral efficiency. It allows increasing the bandwidth or the number of supported users. In [9], [10] and [11] cognitive radio for smart grid systems are discussed. They show that every level of the smart grid communication could benefit from cognitive radio-based architectures, by employing mechanisms such as suboptimal distributed control algorithms to optimize medium access, physical layer or routing decisions.

Data aggregation is another feature needed for the Smart Grid, given the large amount of monitored data. Aggregation techniques such as LEACH [12] and its derivative improve the energy efficiency of large number of nodes by clustering the network. Those protocols are widespread in dense WSN, but offers less interest in our single equipment scenario.

In this article, we propose several mechanisms to provide an efficient monitoring system that relies on the Internet of Things (IoT) paradigm. Complementary to [13], [14] and [15], our challenge is to control the network usage, while maintaining high accuracy of collected data. A solution to reach this goal is to avoid continuous data retrieval by clustering and predicting collection points [16]. Following a similar concept, Gedik and al. [17] proposed a distributed approach that divides the sensing units into a collection part and a prediction part in order to still provide good quality of data. In the following, we will deeply study the trade-off between monitored data, its interpretation and the real-time features depending on the monitoring frequency.

## III. Background Description

As previously mentioned, in the near future, it will become harder for utilities to manage efficiently the grid. In addition, users could benefit from managing locally their production. Therefore, there is a need to monitor local production and interconnect it to Smart Grid architectures. Such innovative architecture designed around Management Systems (MSs) should provide the tools for management and control of local devices, assisted with collection of measurement values and external services.

In this article, we consider such scenario, where MSs manage a set of devices. These devices could be monitoring nodes (monitoring and/or controlling non-smart equipment) or smart appliances (devices already equipped with communication and control capabilities). MSs directly communicates with these devices, i.e. it receives measurements information at regular intervals and sends control commands. These communications will occur within the Local Area Network (LAN) through different types of access technologies such as IEEE 802.15.4, Wi-Fi or Ethernet.

However, such systems will employ several of these devices, which will increase the traffic in LANs. Moreover, some of these devices are constrained in terms of processing, memory or battery, which introduces additional challenges in the system. It is therefore critical to minimize their traffic in the network, to avoid overloading LANs with monitoring and controlling packets, while at the same time to reduce their consumption in order to preserve life time of battery-operated devices.

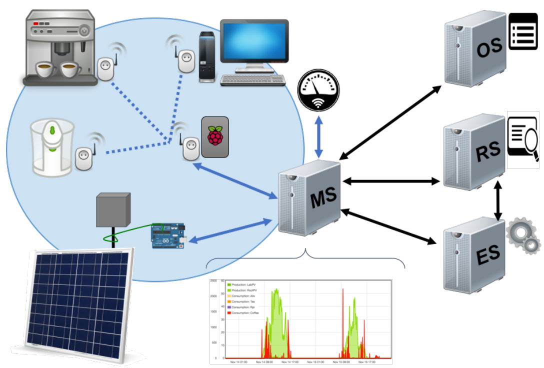



Fig. 2: Representation of the IMT Atlantique Nanogrid.

Figure 2 illustrates such a configuration with a testbed located in IMT Atlantique, Rennes Campus, France. This nanogrid is a smaller managed grid, that can produce, store and consume electricity. It is composed of:

1) An Arduino MEGA, monitoring the production of a fixed $50W$ photovoltaic (PV) panel;
2) Various controllable smart plugs, monitoring the consumption of non-controllable appliances – a coffee machine, an electric kettle and computers;
3) A smart meter, providing the full consumption of the site, and which is usually used for billing purposes;
4) An MS, collecting information from previous nodes, storing corresponding data and providing visualization tools. In the future, we planned to embed it with decision capabilities in order to automate decision and control of certain appliances.

This nanogrid is connected to a distributed architecture [18] that aims to interconnect different energy actors, and which is so far composed of: 1) A Registration Service (RS); 2) Various External Services (ES); and 3) An Ontology Service (OS). The former notably provides tools to search for and access to ES such as production or weather forecast, smart charging, etc. These services may help MSs efficiently manage the considered site as well as the deployed devices. For instance, a smart charging service provides on request the optimal profile for an EV to charge, based on given parameters (state-of-charge, parking duration, etc.). The later provides tools to generate and understand semantic messages. Thus, it helps MSs automatically interpret messages coming from these devices.

Most of the monitored appliances used in this nanogrid are currently not "smart". As a consequence, controllable monitoring nodes (Arduino and smart plugs) are used to actually monitor, as well as to control these appliances. These monitoring nodes measure instantaneous power consumption and/or production at regular intervals, referred as *Monitoring Intervals (MIs)* in the rest of the article. However, as previously mentioned, these monitoring nodes are constrained.

Smart plugs that monitor the appliances are using the Constrained Application Protocol (CoAP) over 6LowPAN and IEEE 802.15.4 [19]. As a result, the payload that can be sent with these nodes is limited to 127 Bytes without considering CoAP and 6LoWPAN headers. Moreover, the intelligence of these plugs is very limited.

The Arduino that monitors the PV panel is using CoAP v1 over Ethernet. Thus, it is limited to one CoAP payload to transmit its information (block is not implemented in the library used [3] and no fragmentation is therefore considered). As a consequence, the CoAP message sent by the Arduino is limited to fit in one UDP packet. Furthermore, we modified the CoAP library so that it operates as both a client and a server. Hence, the Arduino has less than 20% of available memory.

As a result, our monitoring nodes transmit messages continuously to the MS and are limited in terms of intelligence.

With the expected large number of smart appliances in the future, it is therefore of crucial importance to study and propose mechanisms that optimize and reduce monitoring data to be sent. Such solutions will also limit the impact of monitoring traffic on LANs, while benefiting to battery-operated nodes as they will reduce energy consumption.

Listing 1: Turtle representation of PV panel message.

```
@prefix s:<http://purl.org/NET/seas#>.
@prefix e:<http://purl.org/NET/seas/eval#>.
@prefix r:<http://purl.org/NET/seas/quantity#>.
@prefix x:<http://www.w3.org/2001/XMLSchema#>.
@prefix q:<http://qudt.org/schema/qudt#>.
@prefix u:<http://qudt.org/vocab/unit#>.
@base <coap://gasp.ddns.net/>.
<pvpanel/1/power>a q:Quantity;
        q:quantityKind r:ElectricProduction.
<pvpanel/1>a s:Sensor.
[]a e:Observation,e:InstantaneousEvaluation;
        e:generatedBy<pvpanel/1>;
    e:quantity<pvpanel/1/power>;
    e:time"2015-11-06T15:36:33+01:00"^^x:dateTime;
    e:constantValue[q:unit u:Watt;
        q:numericValue"22.45"^^x:double].
```

## IV. Optimal monitoring parameters

In this Section, we particularly investigate the effect of monitoring and transmitting rates on the measurement accuracy, and we propose a light payload format. This study is performed on a PV panel monitoring use-case. However, note that the proposed mechanisms can be used to monitor other devices in an efficient manner and at low cost.

### A. Limiting the Payload to its Minimum

Monitoring a device may take different forms, and the measured data can be formatted in many different ways. In particular, the data unit is often not determined *a priori* (e.g. what is the unit of a production measurement? Joule, Watts, or Watts/hour). In traditional approaches, the monitored data is sent in JSON [20] and parsed by the MS. However, this method is not suited for larger-scale systems as a specific parser would be required for each incoming message. Semantic principles propose mechanisms to automatically interpret incoming messages. This interpretation is made possible by providing additional information that clearly describes the transmitted data. This solution offers the required adaptability to automate both monitoring and controlling of several devices with the least human intervention possible. It will also enable to dynamically change the structure of the data without having to modify the MS.

[3] https://github.com/1248/microcoap

For instance, the code provided in Listing 1 is the semantic representation used for our PV production in Turtle [21] format. In this semantic message, in addition to the measured data (i.e., timestamp and power), contextual information is given. It indicates that this is an electrical production, measured at a given time, by a sensor, from the PV panel number 1, in Watts. Additional information could be embedded as well, such as the PV panel temperature. However, the effective data only represent 5% of such messages, and the rest remains constant from one measurement to another. As monitoring data are expected to be sent several times during a day, sending semantic data descriptions in each message can introduce unnecessary overhead. Nevertheless, these descriptions could be stored once by MSs and used locally to automatically interpret future messages associated with them.

| STID | value1 | value2 | ... | valueN |
|---|---|---|---|---|

Fig. 3: Illustration of the binary templating payload format.

Therefore, we propose a *templating* mechanism that is used to both limit the payload to its minimum and offer flexibility. This mechanism consist of semantic templates and templating payloads. The latter provides a reference to the corresponding semantic template in addition to the measured data. Its format is depicted in Figure 3 and is composed of:

1) A Semantic Template ID (STID, 4 Bytes): the unique Identifier (ID) of the semantic template required to interpret the following information in the packet;
2) The necessary values to be sent (the size of each value is described in the template).

The semantic template is merely the semantic data description of the received message without any measured value (e.g. depicted in green in the PV example of Listing 1). Indeed, measured values may vary over time and therefore will be sent periodically by the node. Instead, the template provides the binary length of each measured value. Thus, the MS could retrieve the corresponding values from the templating payload and fill in the template accordingly. As a result, MS with semantic capabilities can automatically interpret any received information without being aware of the actual payload format.

| STID | TS | V |
|---|---|---|

Fig. 4: PV panel templating payload format.

In our nanogrid, instead of transmitting the full semantic message, the Arduino monitoring the PV panel will provide the three following values (as shown in Figure 4):

- STID: the ID of our semantic description (cf. Listing 1);
- TS: the timestamp of the measured value in epoch format;
- V: the measured instantaneous power in Watts.

However, this mechanism requires a method to retrieve templates corresponding to the received STIDs. MSs can request them from the available Ontology Service (OS). This service provides tools to generate templates as well as associate them with unique IDs. MSs might also request them directly from devices sending templating messages, if they can store their templates. An MS receiving a new templating payload will have to retrieve the corresponding semantic template and fill it with the received data. This mechanism results in a reduction of the monitoring payload by limiting its content while offering flexibility, as receiving devices can automatically interpret these messages with given semantic tools.

### B. Controlling Sleeping Periods

Sleeping techniques and duty cycling are key methods to reduce the footprint of monitoring tools. It can decrease the number of transmitted messages as well as the energy consumption of the monitoring system itself. Considering the intrinsic nature of the monitored equipment, sending periods can be defined. For instance, the consumption of a fridge is well-known and has a constant switching consumption profile. It is therefore not necessary to monitor it continuously to determine its consumption, it is sufficient to transmit the fridge consumption status update (i.e. timestamp and consumption).

In case of a PV panel, it is straightforward that the energy production will only occur during daytime. Therefore, both the monitoring device and the PV panel (for tracking system) should be in "sleeping" mode during the night. However, sunrise and sunset hours vary depending on both the location of the PV panel and the season. The monitoring system would then benefit to automatically adapt to these parameters.

In our testbed, the MS operates when the Arduino can send production measurements. Recall that due to the employed CoAP library, its processing capabilities are limited. Another advantage for such a configuration, is that a MS will request these data only once and then can share them with its managed devices. To retrieve forecast timestamps of astronomical sunrise and sunset, MS searches for a weather forecast service on the Registration Service (RS) based on the PV location. MS can therefore send CoAP commands to the Arduino in order to stop or start the PV production monitoring. This mechanism allows us to significantly reduce the network traffic and energy consumption while keeping high level of accuracy. In the following, we study the trade-off between network traffic and data accuracy.

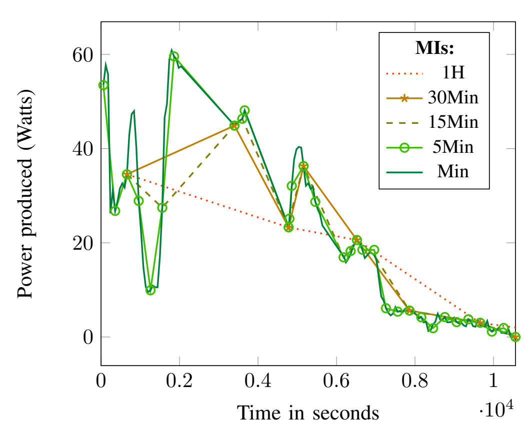
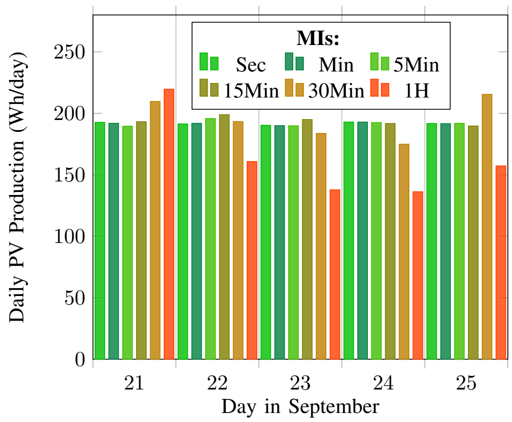
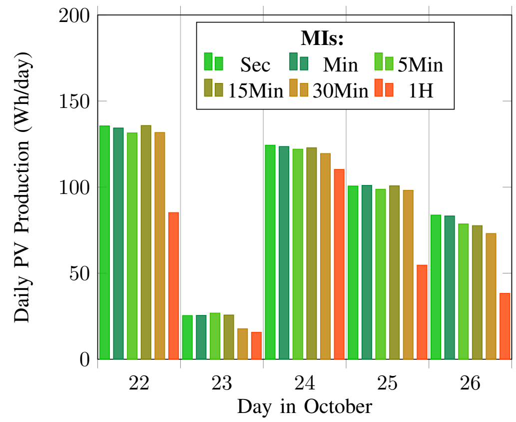



Fig. 5: PV production measurement differences with various MIs.




(a) Week A.




(b) Week B.

Fig. 6: Comparison of daily PV energy production estimation using different monitoring intervals.

### C. Determining the Optimal Monitoring Interval

A PV panel energy production varies over time depending on various environmental parameters, such as the presence of clouds or the position of the sun. While monitoring every second gives an accurate estimation of the PV panel production, it may generate large traffic. Nevertheless, the same level of accuracy might be reached with a higher monitoring interval. However, the more the interval increases, the more likely it will miss some production fluctuations. In this Section, we quantify the error introduced when different monitoring intervals are set. We consider as a baseline the production monitoring data that was collected from the Arduino at every second. Based on these measurements, we evaluate the error introduced by employing different monitoring intervals with five "virtual" PV panels, i.e. monitored every minute, five minutes, fifteen minutes, thirty minutes and every hour. For the MS everything is transparent.

Figure 5 illustrates the measurement differences that occurs between these five virtual PV panels. Each curve represents the production of a PV panel during three hours on November, the 2$^{nd}$ 2016. As it can be observed, PV panels that are monitored every minute and every five minutes have similar production flows. On the contrary, with $MI$=15$mins$, we can see that the monitoring system missed two peaks during this period, while for even higher intervals, several fluctuations are missed. Missing these fluctuations may lead to over-, or under-estimate the resulting production. However, by cumulating these estimations, some errors may compensate over time.

Figure 6a and 6b compare the daily production with different MIs, during one week in September 2016 (week A) and an other one in October 2016 (week B). The daily production for an MI of one second and one minute are almost identical. For an MI of five [resp. fifteen] minutes, the daily production is still very close to the baseline (on average the error is 1% [resp. 3%]). However, for MIs of thirty minutes and one hour, the errors are more significant (i.e. from 6% to 30%). A 3% error on our monitored PV panel represents a difference of $6Wh$ (on September, 21$^{st}$). Considering a $7.5kW$ PV panel installation, an error of 3% would then represent an error of approximately $870Wh$, which is not negligible.

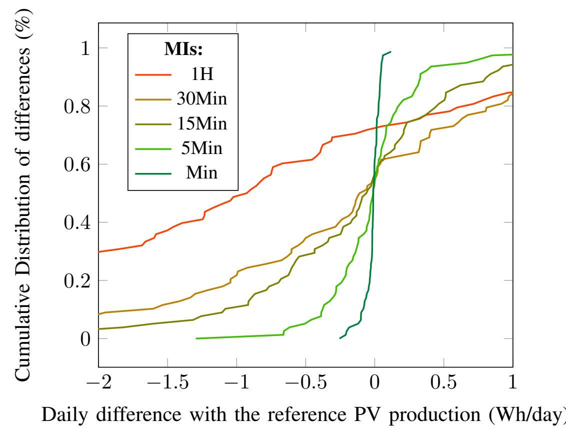



Fig. 7: Comparison of PV production estimation differences on a four months period.

Figure 7 represents the difference between the baseline daily production against the ones using different monitoring intervals during our four months study. These results confirm that a PV panel monitored every minute has an estimated daily production very similar to the baseline. During this period, for 60% of the days, the differences between these two intervals were nearly null. It also illustrates that when the MI increases, the daily production difference becomes higher.

Giving the results that we observed, the monitoring interval trade-off is between one and fifteen minutes. Using $MI$=1$min$ gives a very precise estimation, while using $MI$=15$min$ allows us to reduce by 99.9% the number of transmitted messages (the Arduino is sending only one message against 900 with $MI$=1$sec$). Depending on the size of the PV installation and the usage of the production estimation, a given error can be tolerated. However, on even middle scale nanogrids, a 3% error in the production estimation can be important.

Considering scenarios where the grid transmits solicitations to nanogrids (e.g. "*use only renewable energy for a given period*"), such errors might result in planning mistakes. Hereafter, we develop an additional feature that allows nanogrids to maintain a relevant data accuracy, while limiting the packet rate.

### D. Aggregating Samples to Increase Accuracy

A well-known solution to limit the number of transmitted packets is to aggregate several measurements and send them altogether, refered as *Sampling Rate (SR)*. As a result, instead of sending directly each measurement, the node waits for a given interval, i.e. *Sending Interval (SI)*, before sending all stored measurements since the last packet transmission. The SR is therefore the relation between the SI and the MI. Such aggregation allows us to maintain a short MI, and thus reduce the estimation error, while keeping the network usage very low. However, this comes at the cost of additional delays. The MS will receive measurements after they were actually measured. However, depending on the equipment monitored and the data usage, it might not be an issue to be less real-time while providing accurate measurements. It is therefore possible to employ both monitoring and sending intervals to reach an optimum. Based on the obtained results, a fair trade-off between the accuracy and the network usage could be $MI$=$1min$ and $SI$=$15mins$.

The templating mechanism described in Section IV-A is particularly essential in this case, as it further reduces the payload in each packet. However, the associated template would have to be modified to consider the SR, i.e. sending several measurements at once. Note that, as previously mentioned, this semantic mechanism prevents from modifying the code at the MS, as it will "learn" from the new template how to decapsulate such a new payload.

| STID | TS | MI | V1 | V2 | ... | V15 |
|---|---|---|---|---|---|---|

Fig. 8: PV panel templating payload format with aggregation.

As it is illustrated in Figure 8, from now on, the monitoring node will send the MI value, followed by all the measured values in addition to the usual STID and the timestamp of the first measurement from this set. Thus, the monitoring node only requires to set one timestamp per packet, instead of having one timestamp per measurement. The semantic template associated to this payload will give all the necessary information to interpret all these values, type and units as well as the length in the payload. This sampling aggregation solution increases the binary templating payload, and thus may not be compatible with certain constrained nodes. Therefore, each scenario will have to determine its own trade-off between data accuracy, reception delay and payload size. In fact, the choice of both MI and SI will depend on i) the monitored equipment and its fluctuation rate; ii) the usage of collected measurements; and iii) the capabilities of devices used.

In addition, the templating mechanism, associated with the semantic concepts, offers the opportunity for the MS to control these intervals on-the-fly. An MS would be able to request a monitoring node to change both its MI and SI based on certain information. For instance, we may consider that based on temperature forecasts, an MS could anticipate a household behavior (i.e. modifying the heating consumption) and adapt intervals accordingly.

This interval control would provide certain flexibility for local node management.

## V. Study of Network Losses Impact

Sampling aggregation maintains a low number of transmitted messages and relevant data accuracy. However, it is prone to packet losses, especially under high sending intervals. In fact, loosing a packet that is transmitted every hour, which may include several measurements, could affect negatively the estimation production. In our testbed, communication happens over Ethernet (on a private network), and thus we achieve close to $100\%$ network reliability. However, such monitoring system could be performed over Low power and Lossy Network (LLN) technologies, which are prone to packet losses [22].

### A. Increasing Network Reliability

As TCP cannot be used at the transport layer over LLNs, and as a reliable mechanism at the application layer would be too costly to implement in a real infrastructure, we investigate how we can make a LLN more reliable at the MAC layer.

In the following, we demonstrate how a Power Line Communication (PLC) line and a wireless link can be degraded due to external noises [23], [24], and demonstrate that by using multiple interfaces we can enhance the Packet Delivery Ratio (PDR) performance.

TABLE I: Reliability Experimentation Setup

| Parameter | Value |
|---|---|
| Topology | one-hop |
| Number of nodes | 2 (including the root) |
| Number of sources | 1 source |
| Noise type | White Noise |
| Noise Frequency range | 0.03$Hz$ to 700$kHz$ |
| Noise Amplitude | -3 to 10 $dBm$ |
| Number of packets | 200 |
| Routing | RPL |
| traffic pattern | 1 pkt /3 $sec$ |
| Number of packets per run | 500 |
| Standard | P1901.2 |
| RF Standard | 802.15.4 (6TiSCH) |
| Reliability metric | Packet Delivery Ratio |

To this aim, we deploy an experiment consisting of two Itron smart meters, i.e. see Table I for setup details. The first meter acts as the source of the data packet (i.e. the monitoring node), while the other as the receiver (i.e. the MS). Two communication technologies are used on both nodes: PLC and IEEE 802.15.4. We varied the link quality of the two interfaces over time, by introducing white noise on the PLC link and reducing the transmission power for the other. We performed three experimental campaigns: 802.15.4 only, PLC only, and an hybrid configuration, where the two nodes can use both technologies. In the latter, we extend the algorithm from [25] that selects the best interface, in order to let the sender use the other one in case of transmission failure.

Figure 9 shows a comparison of PDR performances between PLC only and hybrid scenarios. In the PLC only case, when the noise exceeds $-1dBm$, we notice that the link quality is decreasing, and thus the PDR performance drops. On the contrary, when using both technologies, the PDR is always close to $100\%$ for a radio link not really degraded ($94.5\%$ of radio PDR). However, even if the radio link is degraded ($18.7\%$ of radio PDR), the PDR decreases but remains above PLC only scenario.

TABLE II: Comparison of PV production estimation error (in %) during Week B with and without Redundancy Rate (RR)

| | Monitoring Interval | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **Second** | | **Minute** | | **5 Mins** | | **15 Mins** | | **30 Mins** | | **1 h** | |
| NL=0 | 0.00 | | 0.39 | | 2.53 | | 1.39 | | 6.30 | | 33.30 | |
| | With 10% Network Losses (NL = 0.1) | | | | | | | | | | | |
| *Sampling Rate* | RR=0 | RR=1 | RR=0 | RR=1 | RR=0 | RR=1 | RR=0 | RR=1 | RR=0 | RR=1 | RR=0 | RR=1 |
| 1 | 0.02 | 0.01 | 0.50 | 0.25 | 2.53 | 1.27 | 3.67 | 1.84 | 18.12 | 9.06 | 38.65 | 19.33 |
| 5 | | | 0.19 | 0.10 | 5.06 | 2.53 | 17.25 | 8.62 | 26.68 | 13.34 | 57.62 | 28.81 |
| 15 | | | 0.12 | 0.06 | 5.22 | 2.61 | 25.92 | 12.96 | 58.22 | 29.11 | NA | |
| 30 | | | 2.92 | 1.46 | 12.75 | 6.37 | 48.06 | 24.03 | NA | | | |
| 60 | 0.06 | 0.03 | 7.36 | 3.68 | 36.31 | 18.16 | NA | | | | | |
| 300 | 0.15 | 0.08 | 25.28 | 12.64 | NA | | | | | | | |
| 900 | −1.39 | −0.7 | NA | | | | | | | | | |

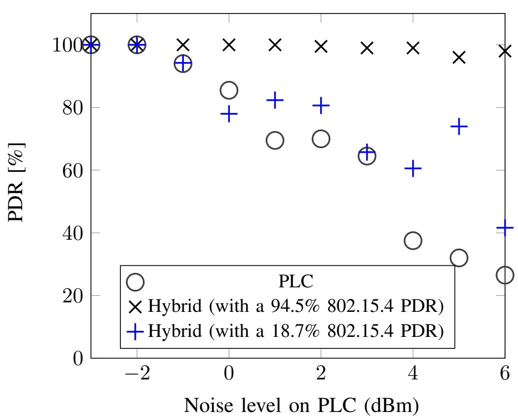



Fig. 9: Packet Delivery Ratio (PDR) depending on noise level over PLC line.

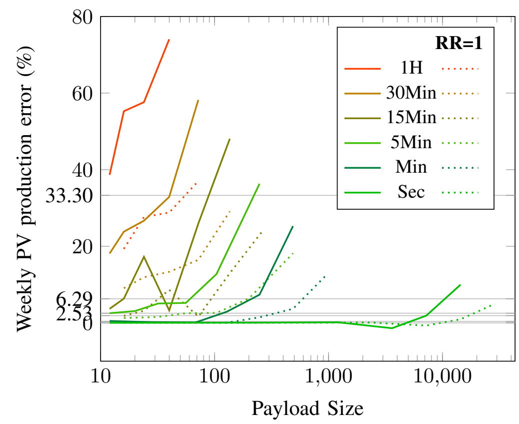



Fig. 11: Redundancy effect on weekly production estimation.

Through this second experiment, we can make the following observations. First, we see that the link quality degrades essentially with the noise, leading to have a low PDR. Second, we show that by employing a hybrid network, we may maintain high level of PDR. However, when both links are bad, we observe a low reliability performance, which explains that mechanisms are required to limit the resulting losses.

| | | | | | |
|---|---|---|---|---|---|
| STID | $TS_n$ | MI | V1 | V2 | V3 |
| | $TS_{n-1}$ | MI | V1 | V2 | V3 |
| | $TS_{n-2}$ | MI | V1 | V2 | V3 |

Fig. 10: PV panel templating payload format with redundancy.

## *B. Introducing a Redundancy Mechanism*

In order to mitigate this packet loss, we introduce in our system the possibility to set a redundancy scheme. It allows for a monitoring node to add in its current messages, some of the previous messages. Thus, a given packet will then not only contain the measurements taken during the last SI, but also the $n^{th}$ previous ones (i.e. n is the Redundancy Rate (RR)). For instance, let us consider that the Arduino uses the following parameters: $MI$=5$mins$, $SI$=15$mins$, and $RR$=2. It will therefore send the current set of measurements as well as the last two sets of sent measurements. In such configuration, the payload of the message sent to the MS will have nine production measurements as illustrated in Figure 10. For example, at 9:30 the Arduino will send the value measured from 9:15 ($TS_n$) to 9:30 as well as the stored data measured at 9:00 ($TS_{n-1}$) and at 8:45 ($TS_{n-2}$).

Table II illustrates the impact of a 10% Network Losses (NL) on scenarios using different redundancy rates as well as monitoring and sending intervals. The results confirm that transmission losses have a significant impact on high SI. With $NL$=10% and $MI$=1$min$, modifying the SI from one minute to one hour increases the production estimation error by 7%. However, the redundancy mechanism allows the system to recover from packet losses, and thus lowers the resulting error.

As previously mentioned, the payload size depends on both sampling and redundancy rates. In order to study the trade-off between accuracy, delay and payload size, Figure 11 presents the relation between the size of the payload (when considering that each binary value within this payload has a 4 Bytes length) and the weekly production error. Solid lines represent the error evolution, with different monitoring and sending intervals and so, payload sizes. While the dotted lines represent the same error evolution, but with $RR$=1.

As it can be observed, the effect of network losses is absorbed by using both low sending and monitoring intervals (every second or minute). However, as expected, these losses have a more significant impact when higher sending intervals are employed. Nevertheless, for $RR$=1, the weekly production error is divided by two, which is already greatly enhancing the results. Whereas, for $RR$=2, the production error reaches the threshold set by monitoring intervals.

This study helps us determine that for a PV production monitoring, the daily production error is lowered to $0.06\%$ with the following configuration: $MI$=1$min$, $SI$=15$mins$ and $RR$=1. In our testbed, with this configuration, the payload sent by the Arduino is of 140 Bytes, which is fairly low and fits into one CoAP payload.

Under these parameters, we have approximately increased the payload by 10, compared to a configuration where both monitoring and sending intervals equals one minute.

However, we have reduced traffic by almost 900 sent packets, compared to a naive approach where the Arduino was sending measurements every second.

## VI. Conclusion and Future Work

In this article, we proposed a set of schemes to limit the impact of monitoring nodes on LANs. In fact, with the increasing energy demand and the popularization of local renewable production, systems that manage both consumption and production phases will be required in the future. Several mechanisms have been proposed in this article to optimize the monitoring traffic. For these mechanisms, we define several parameters such as monitoring and sending intervals as well as a redundancy rate. The optimal value for these parameters severely depends on the type of the monitored equipment, the data usage, the constraints of monitoring nodes and might also be subjective to users. In this article, we identify some best practices depending on device behavior or usage.

For instance, it is not necessary to continuously send measurements for "switching" equipment, such as lights, which have an almost fixed consumption. It is sufficient to provide a status notification message, which includes a timestamp and the new consumption value.

For equipment that has variable and possibly unpredictable consumption or production such as devices with heating elements or used for renewable energy production, both monitoring and sending intervals should be set in order to capture all fluctuations. For instance, our study concludes that for a PV monitoring system, a monitoring interval of one minute and a sending interval of fifteen minutes provide good results.

Finally, the proposed templating mechanism enables the MS to remotely control these parameters, for instance based on external information. It offers the possibility to adapt the data granularity based on requirements. All these mechanisms provide higher flexibility to the system and could significantly enhance node configuration, and thus scalability.

Our ongoing research work consists of further develop the intelligence integration of the MS and allows it to take decisions related to monitoring and control of equipment. In order to reach an optimal level of management, the MS would have to retrieve requirements and information from the nodes, the users (to avoid any undesired equipment unavailability) and the grid. Based on these data, it will have to determine optimal rules to efficiently control each node within the group that it manages. The MS will then take decisions such as 1) shifting the consumption in time; 2) using local production to compensate any new consumption without overloading the grid; or 3) using stored energy; and perhaps 4) switching off some consuming devices.

## Acknowledgment

This work has been partially funded by the ITEA2 12004 SEAS (Smart Energy Aware System) project. Authors would like to thank Tadanori Matsui (Keio University) for its work and valuable contributions in this study.

## References

[1] S. J. Lewis, "Analysis and management of the impacts of a high penetration of photovoltaic systems in an electricity distribution network," in *2011 IEEE PES Innovative Smart Grid Technologies*, 2011.

[2] A. Ouya, B. Martinez De Aragon, C. Bouette, G. Habault, N. Montavont, and G. Z. Papadopoulos, "An Efficient Electric Vehicle Charging Architecture based on LoRa Communication," in *Proc. of the IEEE SmartGridComm*, 2017.

[3] F. Lemercier, G. Habault, G. Z. Papadopoulous, P. Maille, P. Chatzimisios, and N. Montavont, "Communication architectures and technologies for advanced Smart Grid Services," in *Transportation and Power Grid in Smart Cities: Communication Networks and Services*. UK: John Wiley, 2017, ch. 8.

[4] M. Erol-Kantarci and H. T. Mouftah, "Energy-Efficient Information and Communication Infrastructures in the Smart Grid: A Survey on Interactions and Open Issues," *IEEE Communications Surveys and Tutorials*, vol. 17, no. 1, pp. 179–197, 2015.

[5] R. Yu, W. Zhong, S. Xie, C. Yuen, S. Gjessing, and Y. Zhang, "Balancing Power Demand Through EV Mobility in Vehicle-to-Grid Mobile Energy Networks," *IEEE Trans. Industrial Informatics*, vol. 12, no. 1.

[6] R. Deng, Z. Yang, M. Chow, and J. Chen, "A Survey on Demand Response in Smart Grids: Mathematical Models and Approaches," *IEEE Trans. Industrial Informatics*, vol. 11, no. 3, pp. 570–582, 2015.

[7] M. H. Rehmani, M. Erol-Kantarci, A. Rachedi, M. Radenkovic, and M. Reisslein, "IEEE Access Special Section Editorial Smart Grids: a Hub of Interdisciplinary Research," *IEEE Access*, vol. 3.

[8] A. Ahmad, M. H. Rehmani, H. Tembine, O. A. Mohammed, and A. Jamalipour, "IEEE Access Special Section Editorial: Optimization for Emerging Wireless Networks: IoT, 5G, and Smart Grid Communication Networks," *IEEE Access*, vol. 5, pp. 2096–2100, 2017.

[9] G. A. Shah, V. C. Gungor, and Ö. B. Akan, "A Cross-Layer QoS-Aware Communication Framework in Cognitive Radio Sensor Networks for Smart Grid Applications," *IEEE Trans. Industrial Informatics*, vol. 9, no. 3, pp. 1477–1485, 2013.

[10] A. A. Khan, M. H. Rehmani, and M. Reisslein, "Cognitive Radio for Smart Grids: Survey of Architectures, Spectrum Sensing Mechanisms, and Networking Protocols," *IEEE Communications Surveys and Tutorials*, vol. 18, no. 1, pp. 860–898, 2016.

[11] ——, "Requirements, Design Challenges, and Review of Routing and MAC Protocols for CR-Based Smart Grid Systems," *IEEE Communications Magazine*, vol. 55, no. 5, pp. 206–215, 2017.

[12] J. R. I. Pedrasa and G. L. Ortiz, "LEACH-based communication network with a modified sleep protocol," *IEEE Innovative Smart Grid Technologies - Asia*, Nov. 2015.

[13] M. Dhimish, V. Holmes, and B. Mehrdadi, "Grid-connected PV monitoring system (GCPV-MS)," in *2016 4th International Symposium on Environmental Friendly Energies and Applications (EFEA)*, 2016.

[14] I. M. Moreno-Garcia, V. Pallares-Lopez, M. Gonzalez-Redondo, J. Lopez-Lopez, M. Varo-Martinez, and I. Santiago, "Implementation of a real time monitoring system for a grid-connected PV park," in *2015 IEEE International Conference on Industrial Technology (ICIT)*, Mar. 2015, pp. 2915–2920.

[15] G. Bayrak and M. Cebeci, "Monitoring a grid connected PV power generation system with labview," in *2013 International Conference on Renewable Energy Research and Applications (ICRERA)*, 2013.

[16] S. Jeong, H. Kim, D. K. Noh, and I. Yoon, "Energy-aware data aggregation scheme for energy-harvesting wireless sensor networks," in *2016 First IEEE International Conference on Computer Communication and the Internet (ICCCI)*, Oct. 2016, pp. 140–143.

[17] B. Gedik, L. Liu, and P. S. Yu, "ASAP: An Adaptive Sampling Approach to Data Collection in Sensor Networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 18, no. 12, pp. 1766–1783, Dec. 2007.

[18] G. Habault, J. Hursti, and J. Bonnin, "Defining a Distributed Architecture for Smart Energy Aware Systems," in *Complex Systems Design & Management*, 2016, ch. 8.

[19] "IEEE Standard for Local and metropolitan area networks–Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs) Amendment 3: Physical Layer (PHY) Specifications for Low-Data-Rate, Wireless, Smart Metering Utility Networks," *IEEE Std 802.15.4g-2012 (Amendment to IEEE Std 802.15.4-2011)*, pp. 1–252, April 2012.
[20] I. Davis, T. Steiner, and A. Le Hors, "RDF 1.1 JSON Alternate Serialization (RDF/JSON)," W3C Recommendation, Nov. 2013.
[21] E. Prudhommeaux and G. Carothers, "RDF 1.1 Turtle - Terse RDF Triple Language," W3C Recommendation, Feb. 2014.
[22] H. Balakrishnan, V. N. Padmanabhan, S. Seshan, and R. H. Katz, "A comparison of mechanisms for improving TCP performance over wireless links," *IEEE/ACM Transactions on Networking*, vol. 5, no. 6, pp. 756–769, Dec. 1997.
[23] V. Kotsiou, G. Z. Papadopoulos, P. Chatzimisios, and F. Theoleyre, "Is Local Blacklisting Relevant in Slow Channel Hopping Low-Power Wireless Networks?" in *Proc. of the IEEE ICC*, 2017.
[24] G. Z. Papadopoulos, A. Gallais, G. Schreiner, E. Jou, and T. Noel, "Thorough IoT testbed Characterization: from Proof-of-concept to Repeatable Experimentations," *Computer Networks*, vol. 119, pp. 86–101, 2017.
[25] F. Lemercier, N. Montavont, L. Toutain, K. Vijayasankar, R. Vedantham, and P. Chiummiento, "Support for hybrid network in RPL," in *Proc. of the IEEE SmartGridComm*, 2016, pp. 527–532.

**Guillaume Habault** received both the M.Sc. degree in networks and software systems in 2009 and the Ph.D. degree in computer science and telecommunications in 2015 from IMT Atlantique (formerly Télécom Bretagne), Rennes, France.

He worked as a Research Engineer with Orange Labs in 2009 and Cisco in 2010, where he participated on improving last-mile protocols and access to networks. Currently, he works as a Post-doctorate for Institut Mines Télécom. He particularly participated in designing, testing and implementing an Internet of Things architecture for the European project *Smart Energy Aware Systems* (SEAS). His current research interests include, but are not limited to, management and coordination of distributed micro-generation sources in Smart Grid. Such objectives require innovative communication solutions to have real-time as well as dynamic control and monitoring of energy loads.

**Maxime Lefrançois** is a former student of École Normale Supérieure de Cachan, and prepared and passed the Agrégation exam in Mechanics in 2008. He then studied Signal Processing and Computer Science in Grenoble, and obtained his Ph.D. in 2014 in Computer Science at the University of Nice - Sophia Antipolis. He then joined the Connected-Intelligence team at École des Mines de Saint-Etienne to design and develop the Knowledge Model for the SEAS (Smart Energy Aware Systems) research project.

His research interests include Knowledge Representation and Reasoning, Semantic Web and Linked Data, and Semantic Interoperability for the Internet of Things. He is a co-editor of the W3C/OGC Semantic Sensor Network ontology specification. The ETSI SmartM2M working group recently voted two work items with the goal to enhance the SAREF standard with some of the design, development, and publication choices made in the SEAS project.

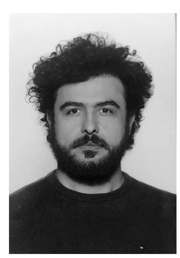
**François Lemercier** received a B.Sc. degree in Electrical Engineering, a M.Sc. degree in Electronics and a M.Sc degree in Electronics and telecommunications from the University of Rennes, France, in 2004, 2006 and 2014 respectively. Since 2015, he is pursuing his Ph.D. at IMT Atlantique (formerly Telecom Bretagne) in the Department of "Network Systems, Cybersecurity and Digital law" and at Itron France. His research interest include routing, interface management and hybrid technologies over smart grid networks.

**Nicolas Montavont** is a full professor at IMT Atlantique, head of the IRISA team called OCIF (Communicating objects and Internet of the Future). He obtained his M.Sc. and PhD degree in Computer science from the University of Strasbourg, France, in 2001 and 2004 respectively. He did a post-doc at National institute of Standard and Technologies (NIST) in Gaithersburg, USA. His research topics during these first years of research were mobility and multihoming management in IPv6 networks. He chaired the Mobile Node with Multiple Interfaces (Monami6) working group at the IETF from 2005 a 2007 and contributed to several standards, with a strong experience in horizontal and vertical handovers.

Pr. Montavont participated to several national (Cyberté, Remora) projects, as well as European projects (Anemone, Sail). From 2007, he chaired an internal working group at IMT Atlantique to work on energy efficiency in communication networks. His research topics evolve toward Internet of Things, smart grids and green communication infrastructure and protocols. His is currently involved in developing Information systems for industrial network and energy monitoring.

**Periklis Chatzimisios** (M'02-SM'12) obtained his Ph.D. in wireless communications from Bournemouth University (UK) in 2005 and his B.Sc. in informatics from Alexander TEI of Thessaloniki (ATEITHE) (Greece) in 2000. He serves as an Associate Professor, the Director of the Computing Systems, Security and Networks (CSSN) Research Lab and a Division Head in the Department of Informatics at ATEITHE, Greece. He is the editor/author of 8 books and more than 110 peer-reviewed papers and book chapters. His research interests include performance evaluation and standardization activities of mobile/wireless communications, Internet of Things and Big Data. His published research work has received more than 2300 citations by other researchers.

Dr. Chatzimisios is involved in several standardization and IEEE activities serving as a Member of the Standards Development Board for the IEEE Communication Society (ComSoc), IEEE ComSoc Standards Program Development Board and IEEE ComSoC Education & Training Board as well as Vice Chair of the Technical Committee on Big Data and Secretary of the IEEE ComSoc Technical Committee in Information Infrastructure and Networking.



**Georgios Z. Papadopoulos** (M'10) serves as an Associate Professor at the IMT Atlantique in Rennes, France. Previously, he was a Postdoctoral Researcher at the University of Bristol. He received his Ph.D. from University of Strasbourg, in 2015 with honors, his M.Sc. in Telematics Engineering from University Carlos III of Madrid in 2012 and his B.Sc. in Informatics from Alexander T.E.I. of Thessaloniki in 2011.

Dr. Papadopoulos has participated in various international and national (FP7 RERUM, FIT Equipex) research projects. Moreover, he has received the prestigious French national ANR JCJC grant for young researchers. He has been involved in the organization of many international events (AdHoc-Now18, IEEE ISCC17). His research interests include Industrial IoT, LPWAN and Smart Grid.

Dr. Papadopoulos has received the Best Ph.D. Thesis Award granted by the University of Strasbourg and he was a recipient of two Best Paper Awards (IFIP Med-Hoc-Net 2014 and IEEE SENSORS 2014).