Information-Quality based LV-Grid-Monitoring Framework and its Application to Power-Quality Control

Findrik, Mislav; Kristensen, Thomas le Fevre; Hinterhofer, Thomas; Olsen, Rasmus Løvenstein; Schwefel, Hans-Peter

Published in:
14th International Conference on Ad-Hoc Networks and Wireless

DOI (link to publication from Publisher):
10.1007/978-3-319-19662-6_22

Publication date:
2015

Document Version
Early version, also known as pre-print

Information-Quality based LV-Grid-Monitoring Framework and its Application to Power-Quality Control

Mislav Findrik[1], Thomas le Fevre Kristensen[2], Thomas Hinterhofer[1], Rasmus L. Olsen[2], Hans-Peter Schwefel[1,2]

[1] Telecommunications Research Center Vienna (FTW), Vienna, Austria
{findrik,hinterhofer,schwefel}@ftw.at
[2] Aalborg University, Aalborg, Denmark
{tfk,rlo}@es.aau.dk

Abstract. The integration of unpredictable renewable energy sources into the low voltage (LV) power grid results in new challenges when it comes to ensuring power quality in the power grid. Addressing this problem requires control of not only the secondary substation but also control of flexible assets inside the LV grid. In this paper we investigate how the flexibility information of such assets can be accessed by the controller using heterogeneous off-the-shelf communication networks. To achieve this we develop an adaptive monitoring framework, through which the controller can subscribe to the assets' flexibility information through an API. We define an information quality metric making the monitoring framework able to adapt information access strategies to ensure the information is made available to the controller with the highest possible information quality. To evaluate the monitoring framework, an event-driven voltage controller is simulated in an LV grid. This controller utilizes the flexibility of photovoltaic (PV) panels to get the voltages into acceptable ranges when the limit is exceeded. This is done by controlling the grid periodically during the time interval that starts when a voltage limit is exceeded and ends when an acceptable voltage level is reestablished. We show how the volatile behaviour of the PV panels causes overvoltages in a baseline scenario. We then show the controller's ability to keep the voltages within their limits. Lastly we show how control performance can be increased by optimizing information access strategies.

1 Introduction

The current electrical grid is facing increased penetration of renewable energy resources. In particular, in the low voltage (LV) grid photovoltaic (PV) panels are being widely installed on rooftops of the end customers, and electric vehicles are expected to be strongly present. Thus, the end customers are transforming from passive consumers to active "prosumers" that can locally generate and feed the power into the grid. The volatile nature of PVs may lead to over-voltage problems that can occur very rapidly in time [1]. Overcoming this challenge requires control of LV grid assets in order to maintain the voltage profiles [2, 3]. Reference [4] outlines the evolution steps of the low voltage grid controlling approaches, starting from "local control" and gradually moving towards "advanced control" that utilizes active management and control via communication infrastructures with limited bandwidth and availability.

We present a monitoring architecture that facilitates operations of control approaches on top of heterogeneous off-the-shelf communication infrastructure with varying network properties. The existing off-the-shelf communication infrastructures are an economically feasible solution for last-mile coverage of LV grids. However, such network infrastructures may be shared (e.g. cellular networks) and are not highly dependable in providing sufficient quality-of-service (QoS). In order to tackle these problems we present a monitoring framework acting as an adaptive communication middleware. The framework features the ability to adapt to network QoS conditions and configure information access to demands of the control approach.

The paper is organized as follows: in Section II the LV grid controller scenario is presented together with a description of the baseline data access scenario; in addition, assumptions on the communication network are presented. In Section III the adaptive monitoring framework is defined and optimization approaches for information access are presented. Section IV describes the simulation framework and shows the results of the monitoring framework evaluation. Finally, in Section V conclusions are made and future research directions are outlined.

2 Power Quality Control Scenario

The scenario of this paper assumes a centralized low voltage grid controller (LVGC) located in a secondary (medium voltage to LV) substation. When more distributed power sources penetrate into the LV grid, voltage control becomes more challenging. In particular, the voltage rise is the major issue in LV grids with high share of PVs due to active power injection and small X/R ratios [5]. The purpose of our reference controller is to keep the voltage levels within defined range by regulating grid-interfaced PV inverters upon over-voltage occurrence. As mentioned in [1], the advantage of having a centralized intelligent voltage control mechanism opposed to a local one at each PV is in ability to do fair and optimal decisions by orchestrating all resources in LV grid.

The controller relies on real-time measurements of voltages at the grid connection points (i.e. electrical buses) and information about active power injected from PVs. This section provides a detailed overview of the voltage control algorithm, describes a baseline communication pattern in which measurements are sent by the assets, and it presents assumptions on the communication network used in this paper. Note that the development of a voltage controller is not the purpose of the paper; instead the paper will use the described controller as the example of a controller that utilizes several assets distributed across the LV grid for solving the over-voltage problem, and for which information access is provided by the adaptive monitoring framework.

2.1 Voltage control approach

The basic principle of the voltage controller is to be active only in the case the measured bus voltages are outside of a pre-defined upper voltage threshold. The limit in our studies shall be set to 5% of the nominal voltage (1.05pu). Upon sensing the voltage violation the controller will be triggered to start running in the periodic time steps T_s. Due to radial topologies of LV grids PVs located upstream the feeder are contributing to voltage rise at Bus i. Therefore, to handle the over-voltage event Bus i, the controller is designed not only to control injected power of PVs at Bus i, but also PVs located upstream the feeder of Bus i. The subset of upstream buses to which the control is spanning is denoted with X and cardinality of X with N. After each time step, the LVGC distributes set-points containing the maximal active power to PV inverters located at the bus where the voltage event has occurred and the upstream buses (see Figure 1).

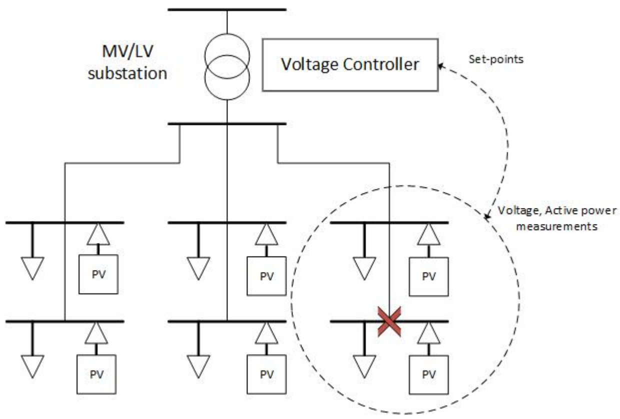

Fig. 1: An example of a LV grid with PVs and overvoltage violation at the last bus

LGVC calculates PV set-points based on the difference between the nominal and the measured voltage at Bus i (when there is over-voltage event at Bus i), as well as the measured active power injection of PV at Bus i and PVs of the buses in Set X. Subsequently upon the arrival, the PV inverters use these set-points for limiting the maximal power injection. The detailed voltage control algorithm is given by Algorithm 1.

read $V_{measured,i}$, $i \in BUS$;
if $V_{measured} > 1.05$ **then**
  $err = V_{measured} - 1.05$;
  $P_{max\ bus\ i} = P_{injected\_i} + K_p * err + K_i * \int_0^t err(\tau)d\tau$
  for $b \in X$ **do**
    $P_{max\ bus\ b} = P_{injected\_b} + K_p * err + K_i \int_0^t err(\tau)d\tau$
  end
end

Algorithm 1: PI control algorithm for stabilizing the voltage at the bus i utilizing PVs at the bus i and the upstream buses $b \in X$

The PV model used for evaluation purposes is adopted from the simulation framework in reference [6] (details are described in Section 4), where active power output in normal operation is given by:

$$p = \begin{cases} p_{rated}, \ \textbf{if} \ \mu A I_{solar} \geq p_{rated} \\ \mu A I_{solar}, \ \textbf{if} \ \mu A I_{solar} < p_{rated} \\ 0, \ otherwise \end{cases}$$

where μ is the efficiency of the solar cells, A area they cover and I_{solar} the solar irradiance.

2.2 Baseline Data Access

The described controller is located in the secondary substation and needs to receive updates from the assets in the grid. A baseline data access is defined for each grid asset: the assets send their update messages to the LGVC according to a Poisson process where the parameter MTBU describes the mean time between updates. The Poisson updates are chosen since their arrival before the controller execution are random, thus yielding the average result as if fix updates would be randomly scheduled in time. An update contains values of the local voltage and active power injection for each asset. In the evaluation section later, all assets will use the same MTBU parameters. The baseline data access is illustrated in Figure 2. The figure also shows two phases. In the inactive phase, the controller receives updates from the assets but does not communicate new set-points. Only when triggered by an out-of band value of a voltage sensor (received at time t_1 at the controller in Figure 2), the controller transits to the active phase in which it communicates set-points to the assets.

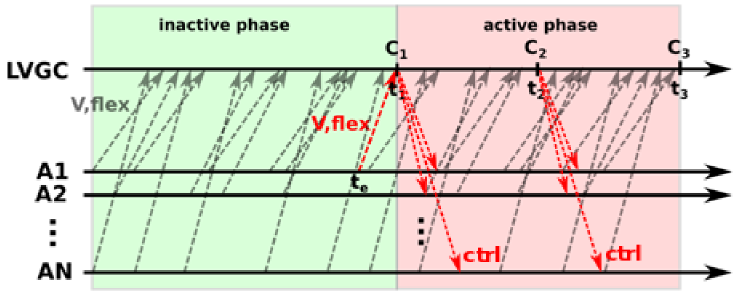

Fig. 2: Baseline data access with Poisson interval updates

2.3 Network modelling

We assume a reliable communication network, e.g. utilizing retransmissions to compensate for losses. Therefore, the network is abstracted via a stochastic process characterizing the message delays. For the evaluations in this paper, we use independent Poisson processes for the communication between the assets and the controller. Poisson process is chosen, as a light tailed distribution (fast decaying tail) of transmission delays resulting from assumption that number of retransmissions follows a geometric distribution. The latter is true when the errors are independent of the size of the transmitting packet (see reference [7] for the details). The upstream delay U effecting the messages from the assets to LVGC is characterized by a rate λ_U, while downstream delay from LVGC to the assets is characterized by a rate λ_D. For evaluation purposes later on we assume that upstream and downstream delays have the same rate, e.g they are symmetric. Furthermore, in the evaluation message re-ordering effects are neglected, since delay values are chosen such that $MTBU > 1/\lambda_U$, thus yielding a low probability of a message $i-1$ arriving before the message i.

3 Adaptive Monitoring Framework

The main contribution of this paper is to show the benefits of using dynamic data access compared to the static baseline scenario. Consequently, this chapter contains a detailed specification of the adaptive monitoring framework.

3.1 High-Level Architecture

The architecture of the monitoring framework consists of four functional layers:

The **Application Programming Interface (API)** layer is a software library used by grid applications for registration and authentication, to request monitoring services (i.e. variable subscription), to receive monitoring data and to issue actuation commands. The main purpose of the API layer is to abstract from the monitoring implementation and to do connectivity management (e.g. timeouts, heartbeats, etc.) to reliably notify the grid application about issues in the communication between the API layer and the publishing layer. The **Publishing Layer** is the main source of monitoring data for a grid application. Applications using the API layer library always need to connect to a local or remote publisher component in order to get data, instead of connecting directly to the sensor measuring the variable. Like the processing and the monitoring layer, the publishing layer is implemented by an independent software application. It receives data from processing or monitoring layers running locally on the same platform or on remote machines. The **Processing Layer** can be used optionally to process measurements on their way from the measurement point (sensor) to the publishing layer. Examples for processing are averaging, outlier filtering or data aggregation. The **Monitoring Layer** implements the measurement and collection of monitoring variables.

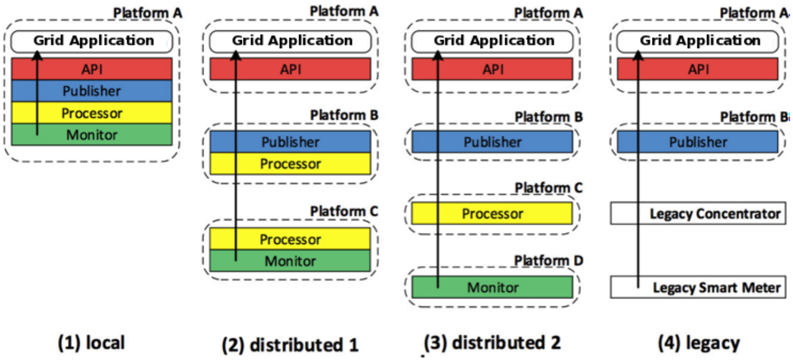

Fig. 3: Application examples of distributed monitoring architecture

Figure 3 shows different integration scenarios of the monitoring system: In (1), all software is deployed on a single computing platform. In (2), a variable is measured (monitor) and processed on platform C, transmitted to platform B for further processing, storage and access management. The grid application is running on platform A and uses the API library to access monitoring data provided by the publisher running on platform B. Scenario (3) is very similar to (2), but all monitoring layers are deployed on dedicated hardware platforms. Finally, scenario (4) shows the usage of the SmartC2Net monitoring system in case of legacy systems like smart meters or concentrators. In our paper scenario, a monitor component is running on every PV asset and data is transmitted to a publisher component in the LVGC placed in the secondary substation. The voltage control algorithm described in Section 2.1 represents the grid application using the API software library to access monitoring data from the publisher.

3.2 Adaptivity Concept

After introducing the main ideas and the high-level architecture of the monitoring framework, we will now describe the developed approach for adaptive data access in more detail. As already mentioned in the introduction chapter, we are using a grid control approach only active in case voltage levels exceed boundaries. This approach of only being active if voltages are out-of-band assures minimal communication requirements (i.e. bandwidth), as no communication and control is needed under normal stable grid conditions. Furthermore, it allows the asset to produce/inject independently of the current power grid status in case voltage levels are stable. In case of a violation of a soft voltage threshold, the considered approach switches to periodic control until the voltages returned to acceptable levels. As a consequence, we will investigate two different phases of the monitoring and control:

- inactive phase
- active phase

Figure 4 depicts our dynamic data access approach, where no monitoring data transfers are happening prior to the active phase (i.e. prior to the reception of the voltage violation event at time t_e). As a result, the input data for the first control step of the active phase needs to be acquired as a reaction to the voltage violation event. In this phase, we call it the *starting phase*, the controller can request its input data dynamically (e.g. subscription to the neighbouring controllable assets of the voltage event location) from the adaptive monitoring framework. After a configurable timeout (waiting time t_{wait}), the voltage controller will do its first computation step C_1 regardless of the complete reception of all asset flexibilities. In case some asset information is not available after t_{wait}, the controller cannot use it for control and the asset is considered to be a normal non-controllable load or producer. As a result, no set-point will be sent to this asset. For simplicity, the control messages are not depicted in Figure 4. Arrows from the LVGC to the assets represent the request of data.

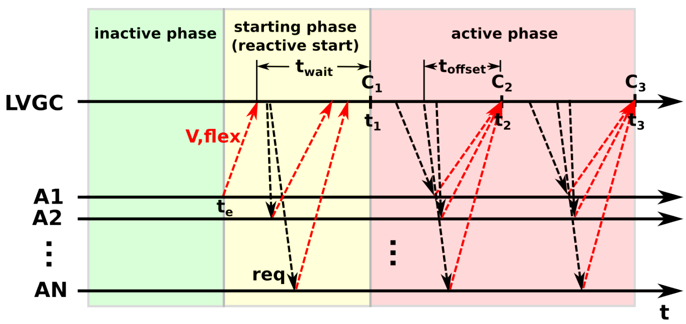

Fig. 4: Reactive start approach

During the active phase, we investigate an approach that uses scheduling of updates as introduced in [8] in order to be synchronised with the voltage controller steps C_i. As a consequence, the resulting data quality is high as the delay between the measurements and the control step is minimised.

3.3 Task Management Approach

The *task manager* is a sub-component of the publishing layer and is responsible for configuring, controlling and coordinating the adaptivity concept mentioned in the last section. The basic approach for the management of the full set of controller monitoring tasks is depicted in Figure 5. It is based on a tight interlinking with the *Quality Estimator* and the *Network QoS Manager*, which are also sub-components of the publisher: (1) The task manager can request a data quality estimation for a specific variable var_{id} and a set of access configurations cfg_{id1},cfg_{id2},.... A configuration cfg defines a specific access configuration (push, pull, event-based, scheduled, etc.). For this paper, we are focusing on the (scheduled) pull (i.e. request-reply) access technique (c.f. Section 3.4. for more details). (2) After receiving a new estimation request, the quality estimator requests the communication network QoS options from the according monitoring source defined by var_{id}. (3) This request is answered with a set of QoS options QoS_1,QoS_2,... from the given source address. The set can be empty if there is no connectivity at all or it can contain multiple QoS objects if the connection can handle different priorities or because of the existence of multiple routes/technologies. A QoS object consists out of a delay distribution and a loss probability. For simplicity reasons, we consider only one QoS option per asset for the studies in this paper. (4) In case there is connectivity, the data quality estimation is acknowledged by returning a unique estimation id est_{id}, otherwise a NACK is returned. (5) Based on the available QoS options, the quality estimator can now compute the data qualities to be expected for the set of requested access configurations. The results of the data quality estimation is finally reported to the task manager by using the updEstimation() function. (6) The task manager will then decide for one access configuration, and the decision is sent to the quality estimator.

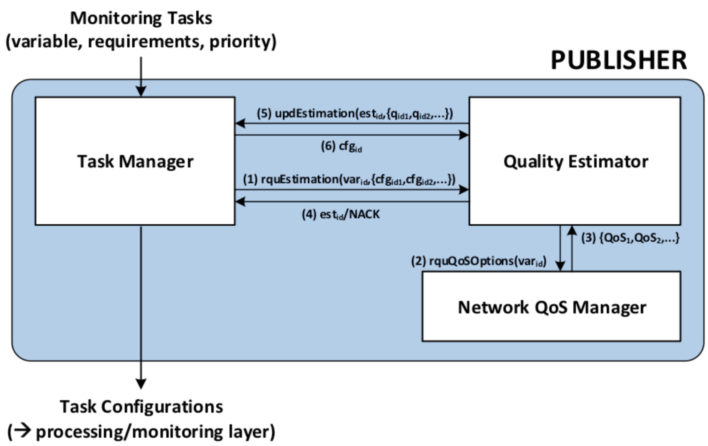

Fig. 5: Monitoring task management approach

3.4 Information Quality Estimation

In this section we present the algorithms for quality estimation running inside of **Quality Estimation** component of the monitoring framework. The algorithms are used for optimizing the information access to the grid measurements (e.g. voltages and PV power outputs) for the controller described in Section 2. The information access is optimized separately for two phases of the grid state; namely, for the first one in which there is no overvoltage and controller is inactive and the second one in which overvoltage has occurred and controller is running in equidistant periodic time steps T_s.

In this paper we measure information quality using the mismatch probability (mmPr) metric. Let S(t) is the state of an asset and S'(t) be the state of the asset known by the controller. S'(t) is updated through the communication network making it a step function that only changes when updates arrive, whereas S(t) can be continuous. We then define the mmPr for the time instance at which the controller runs(t_c) as:

$$Pr(S(t_c) \leq S'(t_c)) \tag{1}$$

This quality metric is used because if the asset has less active power available than the controller thinks, the controller may determine set-points for the asset which is not implementable.

The mmPr depends on two factors: (1) the time from the information is read locally at the asset until the controller set-points have been distributed, which can caused by network delays or caches. (2) the dynamics of the information, i.e. how quickly the information changes to a mismatching value. As stated previously network delays are modelled using Poisson processes. The dynamics of the available power of the PV inverters, considered in the control scenario, is modelled in continuous time through control theory. However, for the purpose of estimating information quality, this model is mapped into a Markov process for its mathematical benefits. This mapping is done by sampling the PV model, discretizing the samples, and fitting the transitions to a Markov process defined by its generator matrix Q.

Information access Optimization in Inactive Phase In the starting phase, flexibility information will be sent using a reactive information access strategy. In this case, the mmPr for a single asset can be calculated as[9]:

$$Pr(mm) = \int_0^{\infty} Pr(mm|t) f_{mmPr}(t) dt \tag{2}$$

where $Pr(mm|t)$ is the probability of the information changing to a mismatching value during the time t, and $f_{mmPr}(t)$ is the density function of the total experienced delay. The total experienced network delay is here defined as the time from the information was read at the sensor until the controller has distributed set-points based on the information, and is, thereby, dependent on the information access strategy.

$Pr(mm|t)$ is computed from the Markov process describing the flexibility information. Since we consider the flexibility of assets, too much flexibility will have no negative consequences for the system, and the information is only considered to be mismatching if the flexibility of the asset is in a lower state when it is utilized than it was when the information was read. For a general Markov process defined by the generator matrix $\mathbf{Q}$ this means[9]:

$$Pr(mm|t) = \sum_{i=1}^{M} \left(\pi_i \sum_{j=1}^{i-1} p_{ij}(t) \right) \tag{3}$$

where π is the stationary probabilities of the Markov Chain, $p_{ij}(t)$ is the probability of being in state j at time t given the state i at time 0, and can be calculated from standard transient Markov chain calculations[9].

$f_{mmPr}(t)$ is for the reactive information access strategy calculated in [9] as:

$$f_{\text{mmPr,rea}} = (f_u * f_d)(t) \tag{4}$$

where f_u and f_d is the density function of the network delay in the upstream and downstream respectively. In this paper we consider several assets providing flexibility information. Therefore, the controller must either wait for all answers to arrive, or determine a proper time to wait before it runs using the information available. We optimize this time by minimizing the mmPr averaged over all relevant assets. However, since we now consider the controller running at the latest after a predetermined waiting time, the stochastic upstream network delay is replaced by the deterministic waiting time. Using this and the reciprocal of the network downstream delay λ_d we get:

$$f_{\text{mmPr,rea}} = f_d(t - t_{wait}) \qquad , t > t_{wait} \tag{5}$$

Using this, the mmPr, $mmPr_{rec}$, can be calculated for a single asset assuming the response was received before the waiting period was over. If the response was not received before the end of the waiting period, there is no old information to be used resulting in a certain information mismatch. Using this and the distribution of the upstream delay $F_u(t)$ we can calculate the mmPr for a single asset as:

$$mmPr_{1Asset} = mmPr_{rec}F_u(t_{wait}) + 1 \cdot (1 - F_u(t_{wait})) \tag{6}$$

This mmPr notion can be extended to N assets by conditioning on the number of responses that is received and averaging over all assets.

$$mmPr_{tot} = \sum_{n=0}^{N} \frac{n \cdot mmPr + (N-n)1}{N} \cdot Pr(\text{n responses arrive}) \tag{7}$$

Information access Optimization in Active Phase Instead of getting pushed updates from the assets in Poisson intervals. The task manager utilizes pull access (e.g. request-response communication patter) to retrieve the information from each controlled asset in the active phase. The goal of the task manager is to find T_{offset} value such that $mmPr$ metric is minimized and asset information arrives before the controller executions C_i (see Figure 4). T_{offset} is a function of upstream and downstream delays (λ_U and λ_D) as well as information dynamics ($\mathbf{Q}$ matrix). In recent paper [8] we derived analytical formulas for deriving $mmPr$ curve for different data access strategies. Here, we recall the formula for case of pull data access:

$$mmPr_{pull} = \int_0^{T_o} \int_0^{T_o-t} \sum_{i=1}^{S} \pi_i Q_{i,i}[exp(Qs)]_{i,i}\mathrm{d}s \int_0^{T_o-t} \lambda_U exp(-\lambda_U x)\mathrm{d}x$$
$$\lambda_D exp(-\lambda_D t)\mathrm{d}t + (1 - \int_0^{T_o}(\frac{\lambda_D\lambda_U}{\lambda_U-\lambda_D}exp(-\lambda_D x) + \frac{\lambda_D\lambda_U}{\lambda_D-\lambda_U}exp(-\lambda_U x)dx)) *$$
$$\int_0^{T_o}\int_0^{T_o+T_s-t} \sum_{i=1}^{S} \pi_i Q_{i,i}[exp(Qs)]_{i,i}\mathrm{d}s\lambda_D exp(\lambda_D t)\mathrm{d}t$$

From the equation above, T_{offset} is derived at the point where $mmPr_{pull}$ is minimal. Due to its length, the closed-form solution is omitted from the paper.

4 Simulation Framework and Evaluation Results

In order to evaluate the monitoring framework we implemented a prototype of the publisher layer inside of DiSC simulation framework [6]. DiSC is MATLAB based open source simulation framework that contains implementation of PV panels, as well as the data of real solar irradiation and household consumption collected in Denmark. Moreover, it allows verification of different control approaches. Utilizing DiSC framework we constructed a LV grid with 7 buses, each having 15 household loads connected together with PVs with total rated power of 7kW. DiSC tool was configured to generate data for all assets and calculate voltages each second, which is the highest precision offered. The voltage profile during a summer day was generated for all buses and shown in Figure 6.

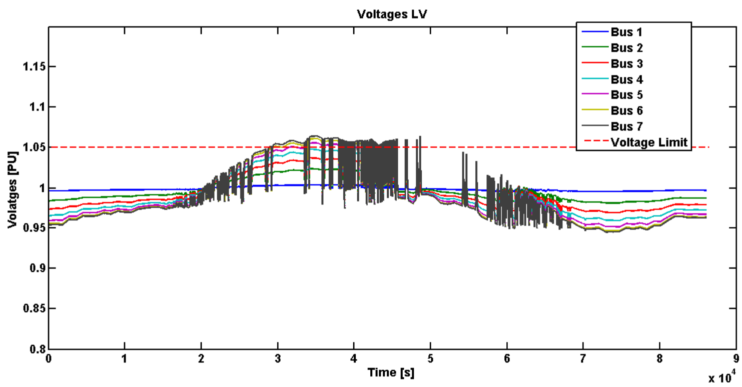

Fig. 6: Voltage profile for all buses during 86400 seconds (24 hours)

In normal operating conditions with no PVs in LV grid, the feeder voltage decreases as the distance from the substation increases. However, with PV installations at all buses the most far bus becomes the most sensitive one where

overvoltage occurs first. Since simulating the whole monitoring stack takes significant amount of time our analysis is further focused on observing the controller performance from $2.6 * 10^4 s$ until $4.5 * 10^3 s$, since in that time interval, overvoltage events are occurring at Bus 7.
Furthermore, to compare the monitoring framework solution against the baseline data access a suitable performance index has to be defined. For this purpose we define Overvoltage Surface (OS) index. The index is calculated by multiplying time in which the bus suffered overvoltage with the voltage value subtracted by critical threshold 1.05pu (i.e integral of voltage curve going above 1.05pu). Note, that during zero delay conditions and infinite rate updates the controller would have OS index value 0 since overvoltage would never happen, whereby worse QoS network conditions or different data access strategies would yield higher index value. The simulation configuration parameters for the controller, the baseline data access and the monitoring framework are given in Table 1.

Table 1: Simulation configuration parameters

Fixed configuration parameters	**Value**
The controller period T_S (active phase)	10sec
K_p (Proportional gain of the controller)	3000
K_p (Integral gain of the controller)	1000
N (# of buses upstream used for over-voltage control)	1
MTBU for baseline access (inactive and active phase)	10sec
Variable configuration parameters	**Values**
T_w	[4 6 7]sec
T_{offset}	[3.6 4.4 6.1]sec
Symmetrical mean delays $\frac{1}{\lambda_U} = \frac{1}{\lambda_D}$	[1 2 3]sec

The baseline data access is compared against the monitoring framework adaptive over three different mean delay values, namely 1, 2 and 3 seconds in both directions. Adaptive parameters T_w and T_{offset} are calculated for corresponding delays using equations given in previous section, which are configured before the simulation runs. The comparison of voltage profiles when using the monitoring framework opposed to the baseline data access is shown in Figure 2. For further analysis OS indexes are calculated form voltage profiles obtained for different delays and presented in Table 2.
The results show that control, even with the poor data quality resulting from the baseline access scenario, is effective in reducing the OS index (hence reducing the overvoltage effect) by more than half for average network delays of 1s or 2s. The baseline scenario is thereby affected by these network delays, the increase from 1s to 3 seconds increases the OS index for almost 100%. Using the optimized information access via the monitoring framework, the OS index is reduced further by almost another factor of 4 and its sensitivity to network delays is significantly reduced.

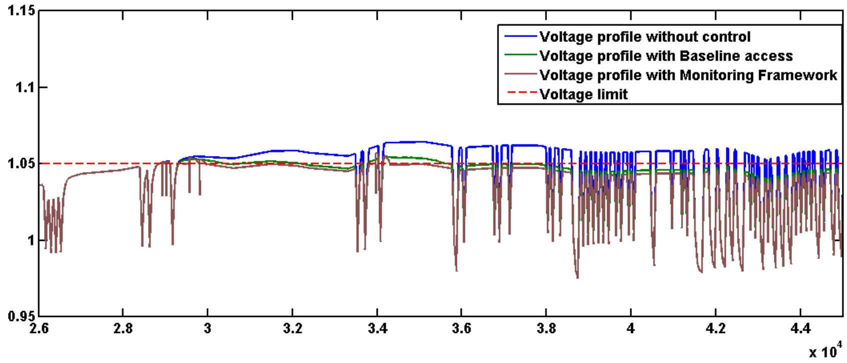

Fig. 7: Voltage profile at Bus 7 for the reference control scenario with 1sec symmetric delays

Table 2: Comparison of data access schemes based on OS index

Simulation Scenario		OS index
Without control (worst case)		11 494
Symmetric delays 1s	Baseline Data Access	2700,8
	The monitoring framework	1308,5
Symmetric delays 2s	Baseline Data Access	3644,0
	The monitoring framework	1352,1
Symmetric delays 3s	Baseline Data Access	5219,1
	The monitoring framework	1382,4

5 Conclusion and Future Work

In this paper a novel monitoring framework for Smart Grid controllers with adaptive information access is presented. We provided a detailed description of the monitoring framework components and described the algorithms within components as well as their interaction. Furthermore, a sample controller for regulating voltage profiles in LV grid was specified and the benefits from the adaptivity provided by the monitoring framework are evaluated via the simulation with respect to the baseline data access. It is shown that control performance with the monitoring solution is significantly improved. The future work shall focus on extensive test-bed evaluation of the framework with different type of Smart Grid controllers running over heterogeneous access networks (power-line communication, cellular networks, xDSL and WiFi).

Acknowledgments

This work is part of SmartC2Net research project, supported by the FP7 framework programme under grant No318023. FTW is also supported by the Austrian Government and City of Vienna ithin the competence center COMET.

References

[1] Ran, B., Negeri, E., Baken, N., Campfens, F.: Last-mile communication time requirements of the smart grid. In: Sustainable Internet and ICT for Sustainability (SustainIT), 2013. (Oct 2013) 1–6

[2] Groenbaek, J., Bessler, S., Schneider, C.: Controlling ev charging and pv generation in a low voltage grid. In: Electricity Distribution (CIRED 2013), 22nd Int. Conference and Exhibition on. (June 2013) 1–4

[3] Kupzog, F., Schwalbe, R., Prggler, W., Bletterie, B., Kadam, S., Abart, A., Radauer, M.: Maximising low voltage grid hosting capacity for pv and electric mobility by distributed voltage control. Elektrotechnik und Informationstechnik **131**(6) (2014) 188–192

[4] Einfalt, A., Kupzog, F., Brunner, H., Lugmaier, A.: Control strategies for smart low voltage grids. In: Integration of Renewables into the Distribution Grid, CIRED 2012 Workshop. (May 2012) 1–4

[5] Juamperez, M., Yang, G., Kjar, S.: Voltage regulation in lv grids by coordinated volt-var control strategies. Journal of Modern Power Systems and Clean Energy **2**(4) (2014) 319–328

[6] Pedersen, R., Sloth, C., Andresen, G.B., Wisniewski, R.: Disc: A simulation framework for distribution system voltage control. Submitted for European Control Conference (2015)

[7] Bertsekas, D., Gallager, R.: Data Networks (2Nd Ed.). Prentice-Hall, Inc., Upper Saddle River, NJ, USA (1992)

[8] Findrik, M., Groenbaek, J., Olsen, R.: Scheduling data access in smart grid networks utilizing context information. In: Smart Grid Communications (SmartGridComm), 2014 IEEE Int. Conference on. (Nov 2014) 302–307

[9] Kristensen, T., Olsen, R., Rasmussen, J.: Analysis of information quality in event triggered smart grid control. In: Proceedings of the 2015 IEEE 81st Vehicular Technology Conference. (2015)