Automation in Construction 145 (2023) 104622

Contents lists available at ScienceDirect

Automation in Construction

journal homepage: www.elsevier.com/locate/autcon

Open source platform application for smart building and smart grid controls

Sebastian Blechmann [a,*], Igor Sowa [b], Markus Hans Schraven [a], Rita Streblow [a], Dirk Müller [a], Antonello Monti [b]

[a] RWTH Aachen University, E.ON Energy Research Center, Institute for Energy Efficient Buildings and Indoor Climate, Mathieustr. 10, Aachen, Germany
[b] RWTH Aachen University, E.ON Energy Research Center, Institute for Automation of Complex Power Systems, Mathieustr. 10, Aachen, Germany

ARTICLE INFO

Keywords:
Smart building
Smart grid
Cloud control
FIWARE
Internet of things (IoT)

ABSTRACT

In this work, an open source platform, based on the FIWARE software framework and other open source components, is used to perform experimental cloud control on two use cases from the smart building and smart grid domains. All communication between the platform and the field layer is realized via the public internet and therefore encryption, authentication, and authorization measures are installed. In the first use case, the supply temperature of a conventional heating circuit is controlled as it is a common task in building energy systems. In the second use case, the power balance of a simulated microgrid is monitored by real phasor measurement units and a controller is used to maintain grid stabilty. The suitability of the platform is validated using requirements derived from literature. The platform is applicable to both use cases. Though, limitations and prospective areas for improvement are identified.

1. Introduction

Our energy systems are transforming and we need to use less energy in general while using more energy from renewable sources. Nevertheless, the building sector is still responsible for 36% of the global final energy use and 37% of the energy-related carbon dioxide emissions [58]. The recent increase in the utilization of monitoring has revealed that many existing buildings exhibit a gap between the predicted (as-planned) and the measured performance (as-is), also known as the energy performance gap [48,63]. However, modern data-driven control strategies that could close this gap require huge amounts of data [6].

In order to increase the amounts of renewable energies, the electrical power system is transforming towards a distributed, hybrid system [44]. In this transformation, there is a focus on distribution networks and low-inertia systems dominated by power electronics, such as microgrids [45]. Effective and resilient operation of such networks requires fast control, which is conventionally achieved by local measurements only similarly to legacy control in power system [22]. Through the single use of local measurements, control in microgrids has its limitations, such as unwanted coupling between control objectives and angle instabilities [40]. Yet, microgrid control requires low-bandwidth communication in order to restore nominal conditions. Therefor, improved communication between remote devices and controllers can contribute to significant advance in microgrid controls, enabling additional functionalities while simplifying control structures at the same time [22,40,44].

Both, the acquisition of necessary field data and realization of fast communication can be achieved by the utilization of modern information technology. For some years already, there has been an undeniable trend towards an increasing interconnection of people, devices and services through the internet [51]. This shift from the traditional use of the internet to an interconnected world is described by the concept Internet of Things (IoT). All those things, e. g. devices and services, create large amounts of data. The resulting constant and dense availability of data results in new use cases, e.g. smart applications. These smart applications offer a significant potential tackling challenges in smart energy [34,36,56,64], more specifically the operation of smart buildings [1,35,38] and smart grids [21,55]. One central element in order to make use of these applications and services are IoT platforms as they connect “sensors or actuators, systems and people” [68]. However, the different nature of data from different domains and the lack of standardization make it a challenge to integrate and utilize data from various sources and use the full potential of the IoT [2,4,8,14,31]. Device manufacturers have built ”IoT islands” [37] when connecting their devices to their proprietary clouds and services. This leads to “little or no

---

* Corresponding author.
E-mail address: sebastian.blechmann@eonerc.rwth-aachen.de (S. Blechmann).

https://doi.org/10.1016/j.autcon.2022.104622
Received 13 November 2021; Received in revised form 24 September 2022; Accepted 10 October 2022

interoperability" between different manufacturer clouds, to an "intranet of things" [37]. Nonetheless, there is a way to overcome these interoperability issues, make them open source. Using open source technologies that work with open standards drives interoperability [68], which prevents limitations using proprietary solutions or single vendor dependencies. One open source IoT platform solution commonly used in research is FIWARE [5]. FIWARE is a project funded by the European Union that has the mission "to build an open sustainable ecosystem around public, royalty-free and implementation-driven software platform standards that will ease the development of new Smart Applications in multiple sectors" [19]. FIWARE is free of charge, has an active developer community and its modularity offers high individuality. Therefore, FIWARE is used in a broad range of use cases. In this paper, we outline energy-related use cases where FIWARE is applied in 2. Yet, to our knowledge, there is no work dealing with cloud controls of real energy systems over the public internet.

In this work, we investigate the suitability of a FIWARE-based platform for two real-world applications from the smart building and smart grid domain. We assess the suitability using requirements we develop for each use case: 1) A monitoring and control use case for a conventional heating circuit: The supply temperature of the admixing circuit in a test stand is controlled via a proportional-integral-derivative (PID) controller using FIWARE's REST APIs. 2) A control use case of an electrical network with phasor measurement units (PMU): A real-time simulation of a microgrid is monitored by PMUs and controlled via a proportional-integral (PI) controller running in the cloud considering latency requirements.

The remainder is structured as follows: In section 2, current projects using FIWARE components and how they differ from this work are described. Subsequently, the contributions of this work are summarized. The used components and the structure of our IoT platform are described in section 3. In section 4, the use cases and their requirements, and test **scenarios** are described while the results of the tests are presented and discussed in section 5. Last, conclusions are drawn in section 6.

2. Related work and paper contributions

In this section, the concept behind FIWARE is described followed by current research where FIWARE-based solutions are used. Concluding, based on identified gaps, the contribution of this paper is pointed out.

2.1. FIWARE

FIWARE is an open source approach funded by the European Union to "accelerate the development of smart solutions" for different domains [19]. FIWARE uses a modular software structure, which leads to high flexibility so it can be customized for various use cases. FIWARE uses the data exchange protocols NGSI v2 [12] and NGSI-LD [17] and standardized interfaces like REST APIs to exchange data with smart devices and software components. The offered software components of FIWARE are called "Generic Enablers" (GE) and can be categorized in different layers, like "core context management", "context processing, analysis and visualization", "interfaces to IoT, robotics and third party systems" and last "data/api management publication monetization". One can choose one or several components from the offered GEs and connect them with third-party or self-written components. An overview of the available GEs and the source code are available on GitHub [18].

2.2. FIWARE-based research and applications

Over the past years, FIWARE-based platforms and applications have proven to be a suitable framework for different domains, like agricultural monitoring applications [32], ship navigation [60], renewable energy system monitoring [61], car monitoring [13], and smart building monitoring [20,33,57,59].

Pozo et al. [43] work on the topics of access control and its influence on the performance of a FIWARE-based architecture deployed via the containerization technology Docker[1]. The authors execute performance evaluations of FIWARE's security and data protection enablers. Furthermore, the authors analyze the bandwidth, CPU and memory usage, and latency for those components. They conclude that the security layer using the OAuth 2.0 protocol does not compromise the performance of the system.

Araujo et al. [7] investigate the performance of FIWARE's Orion Context Broker (Orion) and two different IoT Agents (IoTAs) in a variety of vertically and horizontally scaled systems. The authors observe a bottleneck in throughput to the database for load tests using Lightweight Machine to Machine (LWM2M) and Message Queuing Telemetry Transport (MQTT) protocols. However, these issues have been addressed by the release of MQTTv5 [9] and its implementation in FIWARE's IoTAs JSON[2] and UltraLight 2.0[3]. MQTTv5 comes with the feature of shared subscriptions so that load can be distributed to different clients. Internal performance tests support this thought: We conducted data burst tests to the MQTT broker with a similar architecture in comparison to Fig. 1 in an encapsulated network, thus leaving out all security measures. It is out of scope for this paper to present details but it is planned to publish a more sophisticated analysis.

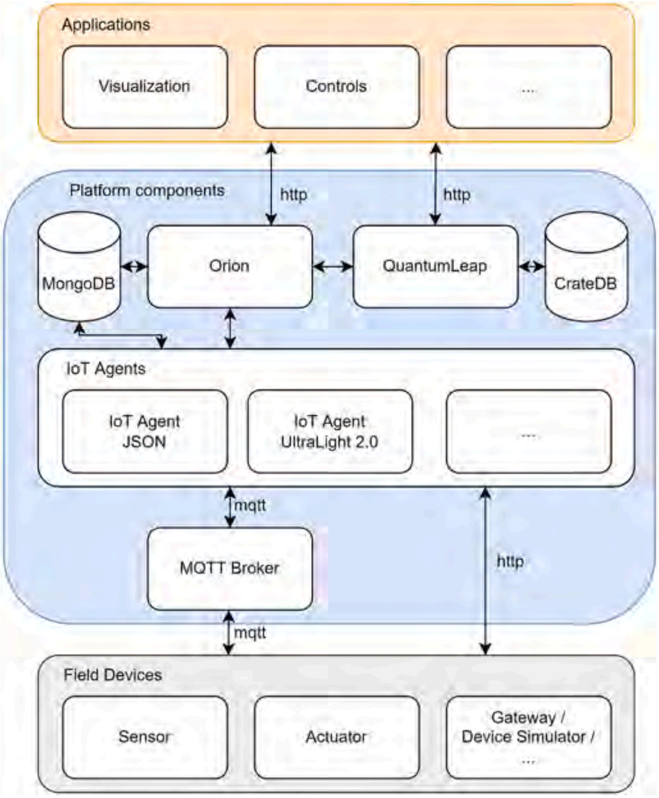

Fig. 1. Service architecture of the platform.

While [20,33,57] only addressed building or energy monitoring, [29,53] conducted control experiments using FIWARE. More specifically, in [53] a virtual room simulation exchanges data with both a virtual heater and a virtual temperature sensor over a FIWARE-based platform while a PID controller is used to control the zone temperature. Kümpel et al. [29] worked on a hierarchical multi-agent control for

---

[1] https://www.docker.com.

[2] https://github.com/telefonicaid/iotagent-json/releases/tag/1.13.0.

[3] https://github.com/telefonicaid/iotagent-ul/releases/tag/1.12.0.

a simulated air-handling unit (AHU). In both, [29,53] the FIWARE-based cloud control achieved sufficient results. Still, both studies mention the possibility that latencies could affect the control quality. Nonetheless, these studies only focused on the control of simulated devices and systems. Another gap is that most systems are built upon encapsulated networks with no specific security measures counteracting in creating a transferable use-case for real-world applications. As one of the most common control tasks in building energy systems, we decide to investigate the operation of a heating temperature control via the secured FIWARE platform.

In the context of power system applications, research in [23,24,41,42] focus on the integration of PMU measurements into the FIWARE platform. In [41], the authors integrate PMU measurements in a monitoring use case for a real network with a measurement reporting rate of up to 10 phasors per second, while in [23,42] the reporting rate increases to 50 and 100 phasors per second. The authors highlight the importance of further improvement due to performance issues with high reporting rates. Yet, to our knowledge, there is no communication-based control of a microgrid using a FIWARE-based setup, especially not applying security measures. To fill the gap, we conduct a communication-based control loop with a measurement reporting rate of 100 phasors per second via the secured FIWARE platform.

2.3. *Contribution*

The main contributions of this paper are summarized as follows: 1) A single FIWARE-based platform is used to conduct control experiments with real-world applications on both a smart building, and a smart grid use case. The suitability of this platform is validated based on requirements of real-world applications from the smart building and smart grid domain. 2) The viability of cloud control via a FIWARE-based platform in a production-ready system considering security measures is demonstrated. Platform, devices, and controllers don’t communicate via an encapsulated IT network but via the public internet over encrypted connections with authentication and authorization measures applied. 3) The feasibility of contributions 1) and 2) are demonstrated on a platform setup that relies entirely on open source software components.

3. Setup architecture

The method used in this work is an experimental evaluation of the suitability of the FIWARE-based platform considering real-world requirements in smart building and smart grid control applications. This section provides an overview of the used platform architecture followed by a description of the specific platform-related experimental setup for the field study. The remaining components of the experimental setups including the use cases and their requirements are described in section 4.

3.1. *Platform architecture*

All platform components are instanced via Docker. We use the Docker swarm mode for orchestration of our components. Docker Swarm allows replicating software components and therefore allows load balancing. A software component instanced in Docker is called a service. The used software image of each component is summarized in Table 2 in the appendix and offered for download from Docker Hub[4]. An overview of the platform architecture is shown in Fig. 1. Information how to set up such platform for testing purposes and further tutorials are available on our Github presence[5].

The core of the platform is FIWARE’s Orion that is used as the central context management interface. Orion is stateless and therefore saves all information in a database, the MongoDB. Orion only stores the last values in MongoDB. The capability to save time series data is added by the use of the GE QuantumLeap. A subscription needs to be created at Orion for it to send notifications about new values or values matching certain conditions to a notification endpoint, in this case QuantumLeap. Latter processes the incoming notification and saves the data to a high performance time series database, CrateDB. CrateDB recently shut down its enterprise version and switched to a full open source license even for distributed clusters. Both, Orion and QuantumLeap offer REST APIs to communicate with them and exchange data.

In order to connect devices working with different protocols, FIWARE offers a variety of IoTAs. These translate certain protocols into the platform’s internal NGSI protocol. In this work, the IoTAs for the protocols JSON and UltraLight 2.0 are used. Devices are registered at the IoTAs so that the data stream is forwarded to Orion. The IoTAs save their device data in the same MongoDB instance like Orion does, just in a different database. The IoTAs offer Rest APIs for device registration and data exchange over http. Low performance devices usually do not come with an http stack. The IoTAs JSON and UltraLight 2.0 come with client libraries for AMQP and MQTT. In this work, the popular lightweight MQTT protocol for asynchronous messaging [3,50] is used for communication between the platform and the field devices. As an MQTT broker, we use an adapted version of the mosquitto broker. This allows communication with an external identity and access management (IDAM), as described below. The used devices are described in section 4.

Software like visualization tools, control, or data processing algorithms can communicate with the platform’s APIs. Software can run on the same hardware as the platform, on different hardware in the same network, or outside the platform. This depends on whether the service is trustworthy. The applied control services are further described in section 4.

For the service configuration for Orion, MongoDB, IoTAs and CrateDB we refer to the recommendations in the according manuals.

3.2. *Security measures*

Despite the software described, further components are used to realize confidentiality, authenticity and integrity. The architecture of these components is presented in Fig. 2. As IDAM the open source software Keycloak is used. It supports the use of OpenID Connect and OAuth 2.0 for authentication and authorization, respectively. If no local configurations, like access control list and username-password-list, exist, the MQTT broker checks with the IDAM upon incoming connection: First, the broker checks whether the client is allowed to connect. Subsequently, the broker checks whether the client is allowed to publish and subscribe to the desired MQTT topics. Http requests reaching the

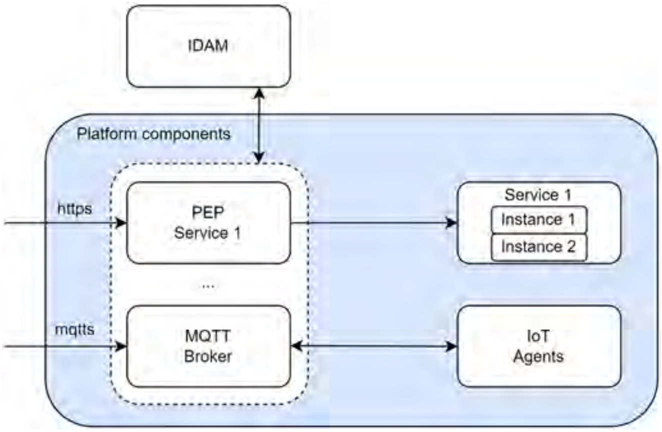

Fig. 2. Schematic of authentication and authorization flows between PEP proxy / MQTT broker and IDAM.

---

[4] https://hub.docker.com/.

[5] https://github.com/N5GEH

platform via the public internet are caught by a policy enforcement point (PEP) proxy, in this work gatekeeper. The PEP proxy checks with the IDAM whether the requesting client is authenticated and authorized to access the requested resource. The MQTT broker and PEP proxies only accept TLS encrypted connections.

The choice of these components is just a recommendation based on our previous experience with these components. Nevertheless, all components need to be configured carefully to ensure security. We recommend the security to be tailored to the specific use case needs. Suggestions on how to secure a gateway in IoT applications are written down in [11].

3.3. Hardware specifications and orchestration

All platform components are running on virtual machines (VMs) hosted on a VSphere cluster with Ubuntu 20.04.3 LTS. In total, the cluster consists of 7 VMs as shown in Fig. 3. The according hardware ressources are listed in Table 1. On each VM the containerization technology Docker (Docker enginge CE 20.10.8). Using Docker swarm mode, all 7 VMs are placed in an encapsulated network, a swarm network, and form a so called swarm. Docker swarm allows scaling and distribution of services to multiple VMs which in conclusion allows load balancing. The swarm is formed by 3 manager and 4 worker nodes. Two of the worker nodes host the services that need to be reachable through the internet. The remaining 2 worker nodes host the databases, they are attached to a network storage and tagged as storage nodes. By duplicating each node type at least once we prevent the platform from crashing in case one node goes down due to maintenance or errors. We recommend, that duplicated nodes are not hosted on the same hardware and the use of replicated databases for business cases. Furthermore, access between different containers should be restricted to the absolute necessary.

4. Use cases

In this section, the two use cases, their according requirements, and their interaction with the cloud platform are described.

4.1. Building energy system use case

In this work, as a first step, we investigate the suitability of the FIWARE platform in a use case of a standard heating circuit temperature control via PID controller. Heating temperature control is a common task in the operation of building energy systems (BES) as it is used in heat distribution and heating, ventilation and air conditioning (HVAC) system applications to condition the supply air. First, requirements for the control of BES are summarized. Second, the used setup and the communication flows are described.

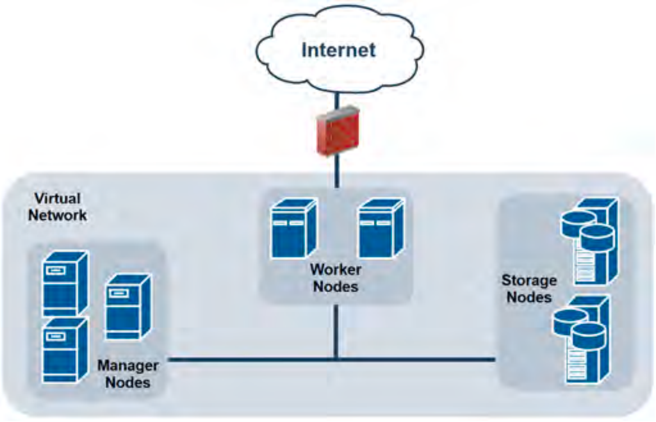

Fig. 3. Server architecture.

Table 1
Hardware specifications for each node type, CPU clock speed 2.3 GHz.

Node Type	Hardware Specifications
Worker Node	4 CPU, 4 GB RAM, 50 GB HDD
Manager Node	8 CPU, 8 GB RAM, 50 GB HDD
Storage Node	4 CPU, 8 GB RAM, 250 GB HDD

4.1.1. Building energy system use case requirements

Concrete requirements in terms of measurable key performance indicators on the control of BESs are hard to come by and heavily dependent on the application. For example, key performance indicators (KPI) like overshoot, rise time and settling time will have considerably stricter requirements when conditioning an operating room or laboratory than an office climate control. On the one hand, in applications, researchers would often compare to a reference control rather than to an absolute reference. On the other hand, many sophisticated control approaches already exist, however their control performance assessment is limited as there is no specific reference system and control to be compared to [28].

To verify if the control via the cloud platform is suitable or not, we derive the requirements for the application from literature in the following while the specific setup used in this work is described in section 4.1.2:

- In [65], Zhan and Chong investigate the requirements for model predictive control (MPC) in buildings. In accordance to literature, they find a model mismatch error to be acceptable below +/− 1 K when comparing modeling methods and their validation.
- When referring to dynamic response analysis, according to O'Neill et al. [39], tolerance bands of +/− 1, 2 or 5% are usually applied with respect to the reference. Additionally, they define an acceptable error band of +/− 0.56 °C for both hot water supply and room temperature control using the variance band KPI.
- According to the ASHRAE comfort standard, the authors in [62] state, that their control approach satisfies comfort requirements if the indoor supply air temperature violation of +/− 0.5 K holds for less than 15 consecutive minutes.
- In [67], Zhang et al. compare a fuzzy control with a model reference prediction fuzzy adaptive control simulating a valve-controlled heating system. They list a couple of KPIs like the rise time and overshoot. However, they do not provide any reference for a sufficient performance.
- Another metric is introduced by Haissig [25], who defines control quality as the amount of time, the temperature stays within a tolerance of the setpoint devided by the overall time. Haissig compares the control quality of a PI controller to an adaptive fuzzy control for a room temperature tolerance of +/− 0.5 °C and +/− 0.25 °C.

Most of these studies provide a reference for the indoor air temperature rather than the underlying heating circuit control temperature, whereas the latter usually has a lower inertia and higher dynamics. Therefore, it reacts much more sensible and should be allowed to tolerate higher fluctuations. Moreover, the majority of the studies listed above uses absolute temperature deviations. In this paper, we set a tolerance of 1% for the temperature to deviate from its reference as acceptable since a relative reference allows for a lower dependency and it is in accordance to O'Neill et al. [39]. This also covers the acceptable error given by Zhan and Chong [65] for water temperatures in heating circuits up to 100 °C. As an additional validation, KPIs like the rise time, settling time, and control quality are assessed and compared to the results given by the authors above.

Furthermore, Zhan and Chong [65] define data transmission requirements according to the level of detail at 2^n data points per hour. At level 7, the highest given LOD, a temporal resolution of, 128 data points per hour are required corresponding to one data point every 28 s. They also find that most studies (on MPC in buildings) assume constant room temperature setpoints or the setpoint is considered as model input.

4.1.2. Building energy system use case

This use case comprises a temperature control for an admixing circuit as it is a common hydraulic system in heating, ventilation, and air conditioning systems. A sche-matic view of the experimental setup and the specific components is depicted in Fig. 4. In this setup, heat is generated in the distribution circuit by an electrical heater and transferred to the consumer circuit, a short air channel system, via an air-water heat exchanger. On the water side, as there is no distributor, the hydraulic pump in the consumer circuits drags hot water directly from the heat generation depending on the valve's position. On the air side, the airflow is generated by a fan unit. The water temperatures are measured at the illustrated locations in Fig. 4.

To keep the setup simple, both fan and pump are set to fixed setpoints of 1710 rpm rotational speed and 4.5 m^3/h volume flow, respectively. The electrical heater is set to regulate the primary supply temperature to 75 ˚C. Accordingly, the control task consists of regulating the supply temperature for the consumer by adjusting the valve position which is operated by a steadily controlled actuator. The valve position at 0% corresponds to all water flowing through the bypass while at 100% the bypass is closed.

The field devices, more specifically, the temperature sensors, the fan, and the valve actuator, are wired to an edge device. At the edge device, the analog values from the field devices are digitalized and sent to the platform while incoming control signals from the platform are converted back to analog signals and sent to the devices. The communication between edge device and platform is realized via MQTT.

For our tests, the position of the valve actuator is controlled by a PID controller based on the simple-pid Python module. The valve setpoint corresponds to the manipulated variable of the PID controller while the control variable corresponds to the supply water temperature before the heat exchanger. All interactions between the control module and the real system, e. g. fetching the current value of the supply temperature and returning the setpoint for the valve, are performed using the provided platform APIs. Hence, the PID controller can be operated on any computing device since it is bound neither to the field nor to the cluster where the platform is running on. In our experiments, the controller runs on a computer detached from the cluster's network. Upon start, the controller authenticates with the IDAM: It fetches an access token. Subsequently, it accesses the current value of the control variable at Orion, calculates the new manipulated variable, and sends a PATCH request back to Orion to inform it about the new setpoint. The new setpoint is passed on to the IoTA JSON, to the MQTT broker, the edge device, and last to the actuator itself. If the controller's access token expires, it can fetch a new access token by using its refresh token. For communication with the platform, the controller uses the software library FiLiP [52]. This Python library has been developed to standardize and simplify communication with FIWARE components.

To validate that the detached controller is able to control the heating circuit system, we conduct a step response test where the water supply temperature is changed from 45 ˚C to 55 ˚C. The results are shown in section 5.1.

4.2. Microgrid use case

The electrical network use case presents the operation of an islanded microgrid which is controlled by a cloud controller. A microgrid is an islanded grid that is disconnected from the bulk power system. The microgrid in this work includes real PMUs and other, virtual devices for monitoring and controlling purposes. The dynamics of the microgrid itself are simulated in a real-time environment. The simulated microgrid which is illustrated in Fig. 6 includes buildings, transmission lines, distributed energy resources (DER), such as wind parks and photovoltaic plants. The buildings represent system loads, while the DERs represent generating units with battery storage systems. First, the requirements for a control of microgrids are stated. Second, the setup and its communication flows are described.

4.2.1. Microgrid use case requirements

In conventional hierarchical controls for microgrids, fast control objectives are realized through strategies that do not require communication. In this work, a communication-based approach via a cloud controller is examined. In general, the controller needs to fulfil tasks of conventional hierarchical controllers for microgrids with droop and secondary controllers. E. g., the controller needs to keep the power balance and control the system frequency and voltage. Therefore, the controller's performance, in more detail the performance of the communication, determines the control quality of the microgrid. Cloud control through a single control layer with high performance communication simplifies the control structure and enables additional capabilities, such as integration of more advances objectives. In this work, power balance, frequency, and voltage control are taken into account as requirements to be met. More sophisticated objectives are not subject of this paper.

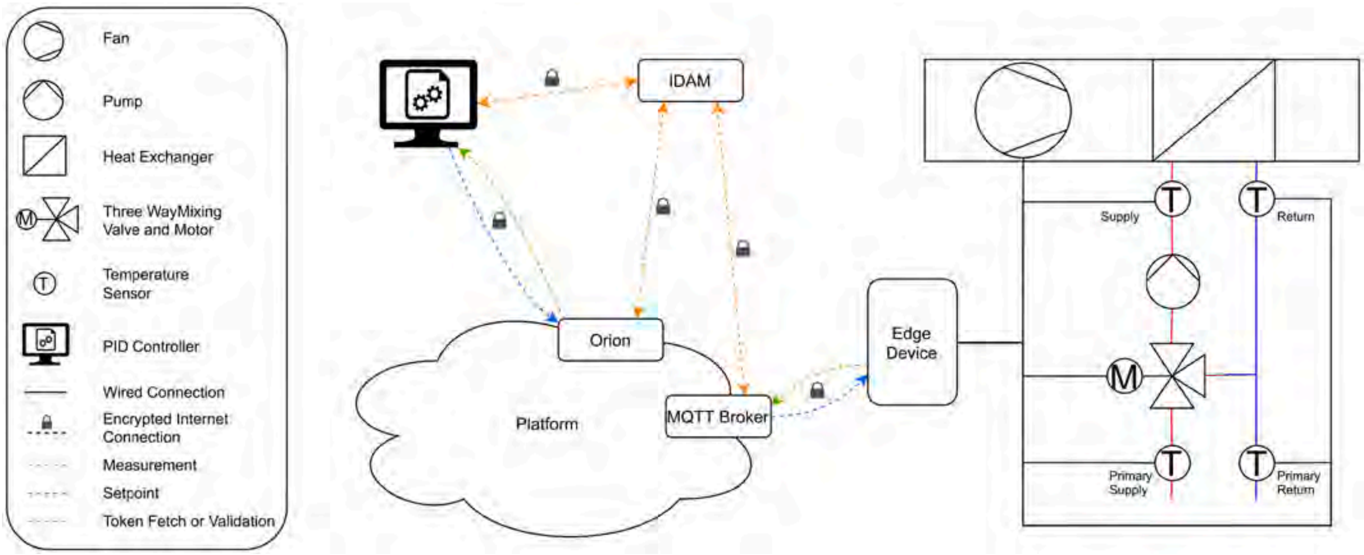

Fig. 4. Schematic of the heating circuit, its automation network, and the cloud communication with the controller.

The main challenge to run communication-based monitoring and controlling for such a microgrid in a platform is to handle measurements from devices with high reporting rates. The measurements need to be communicated and processed in order to derive the control setpoints, which are then sent back to the actuators in the microgrid. Microgrids including only power electronic inverters possess very low inertia and thus have high dynamics. In other words, small changes in load or generation can quickly lead to significant power imbalance and to subsequent problems with the stability of the system and to violations of power electronics hardware limits. Therefore, the delays in the control loop must not be too large, otherwise the operation of the entire microgrid can be compromised in different ways. If there is too much delay, the microgrid’s status can become critical shortly after a disturbance occured.

The requirements for the delays in the control are determined by the stability of the control loop and through limitations of the power electronics hardware. Additionally, the performance of the cloud-based controller should be at least comparable to conventional local primary controllers usually proposed for microgrids, despite the delays. However, the tuning of the controller towards superior performance is out of scope of this paper. The requirements are derived as follows:

- According to Serban et al. [47], the maximum allowed communication delay for controls in islanded microgrids varies in a wide range and is dependent on diffent conditions and participaating devices. It spans from 60 ms to 2 s of fixed delay in different reviewed applications.
- Current limits of the considered power electronics hardware determines another limit on the communication delay. The communication delay is crucial, especially when facing large disturbances. If the controller does not respond fast the power electronic devices are subject to the changes and could face high currents that could damage them.
- In order to compare the controller performance to different conventional local primary controllers, time to reaching steady-state of power balance after a disturbance is taken as an indicator based on [16].

4.2.2. Microgrid and its controller

Fig. 7 presents the basic electrical structure of the microgrid. It includes two types of power electronic inverters: Grid forming and grid feeding inverters.

In case of imbalances in the network, the grid forming inverters primarily balance changes in the microgrid. The PMU devices and the inverters themselves are used to monitor the microgrid, providing information about the changes in the power balance towards the microgrid central controller (MGCC). Based on these information, the MGCC derives the setpoints for the grid feeding inverters. After they receive the setpoints, the grid feeding inverters adjust their infeed, balancing the grid and reducing the output power of the grid forming inverters. Due to high dynamics in the microgrid and the dependence on communication between the devices, the effectiveness of such controls depends strongly on the delays in the communication network, cloud processing, and possible other delays.

Similar to the heating circuit control, the MGCC is detached from the FIWARE platform and it only obtains or publishes values from the platform’s APIs using the public internet. Details about the communication are described in section 4.1.2 for the heating circuit control with the only difference that the UltraLight 2.0 protocol is used and the according IoTA is addressed. In principle, the MGCC fetches the latest measurement values, computes the setpoints and sends them back to the inverters. A communication schematic is shown in Fig. 5. The MGCC is designed as a basic PI-based control structure that calculates setpoints for feeding inverters based on the measurements of the PMUs and the

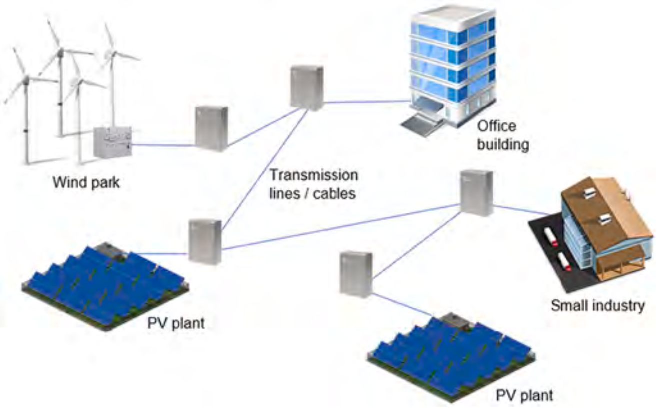

Fig. 6. Schematic of an islanded microgrid.

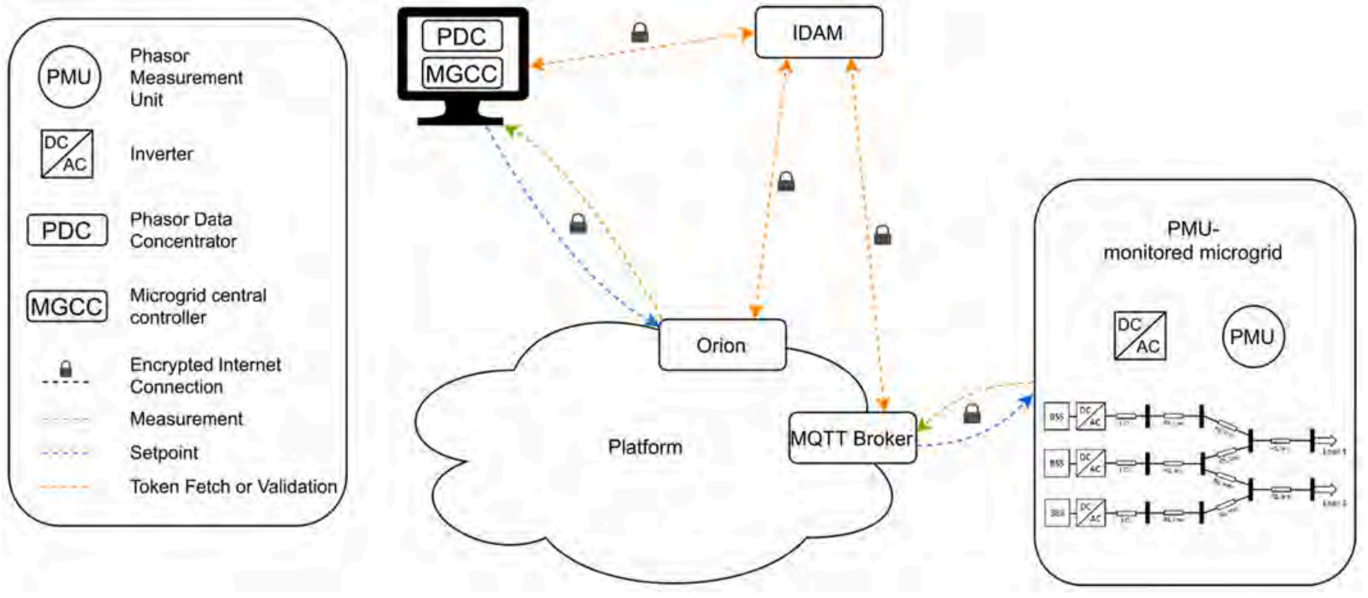

Fig. 5. Schematic of the microgrid and the cloud communication with the controller.

grid forming inverter.

5. Results and discussion

In this section, te results of each use case are presented and discussed.

5.1. *Results of the cloud controlled heating circuit*

The results of the step response test are shown in Fig. 8. The temperature step is applied at approximately 15:08 after the supply temperature reached equilibrium at 45 ˚C for 30 min. The setpoint for the supply temperature is then changed to 55 ˚C. The valve opens up and the primary supply temperature drops until approximately 66 ˚C. This is because there is no storage capacity on the primary side. The electrical heating controller notices the drop, consequently raises the electrical power and the primary supply temperature reaches the setpoint of 75 ˚C 12 min after the valve is opened.

The PID controller for the supply temperature control is tuned manually at the start temperature of 45 ˚C and the parameters are kept constant during the experiment. For the control variable, according to O'Neill et al. [39] and covering aspects from Zhan and Chong [65], the range of tolerance is set to +/− 1% with respect to the setpoint or +/− 0.55 K. The controller is able to adjust and keep the temperature within this tolerance band.

As can be seen from Fig. 8 and in accordance with [62], there is no consecutive duration of 15 min with an error greater than +/− 0.5 K before and after the step is applied. The supply temperature oscillates with a maximum overshoot of 1.57 K or 2.85% from the reference (note that the undershooting amplitude is with −2.50 K or 4.5% to the reference higher due to the drop in primary temperature).

After a settling time of roughly 11.3 min or 678 s (the relatively slow settling time is additionally influenced by the slow adjustment of the primary supply temperature control), the control variable stays within the defined tolerance and the rise time amounts to 33 s. The fuzzy control of Zhang et al. [67] achieves 230 s rise time and 23.4% overshoot, their reference prediction fuzzy adaptive control achieves 340 s rise time and 0.8% overshoot. Compared to these results, the control in this application peforms quite well, especially considering that the results of Zhang et al. have been achieved in simulations.

The control quality according to Haissig [25] is calculated to 22.2%. This is significantly lower compared to the results by Haissig. However, Haissig controlled the room temperature and the period under consideration is significantly longer.

Overall, the controller is capable of keeping the control variable within the tolerance and the settling time is of equal magnitude as the dynamic of the electrical heater. Therefore, even with the manual parameter tuning in one operating point, the control shows sufficient quality and the FIWARE platform proves to be suitable for BES control applications. Yet, especially when using online data, adaptive and automatic tuning offer great potential in optimizing the control. In fact, studies have shown that MPC has a superior performance over PID control [15,26,30,46,66]. However, da Silva et al. [49] conclude from their studies that potential performance increase heavily relies on the system model mismatch, e. g. given low MPC model quality, a PID still outperforms an MPC and is more suitable for robust solutions due to lower complexity. Sturzenegger et al. [54] find that savings in operational expenses are often compensated by the additional engineering, commisioning and hardware required for an MPC implementation, therefore not justifying the implementation effort in the end. As reported in Henze et al. [27], building and calibrating the models account for 70% of the total effort. Thus, the enhanced data availability enables an automized process to extract an accurate model and an overall higher building performance in IoT-based BES, which has led to the research field of data-driven MPC [10] and is focus of future research.

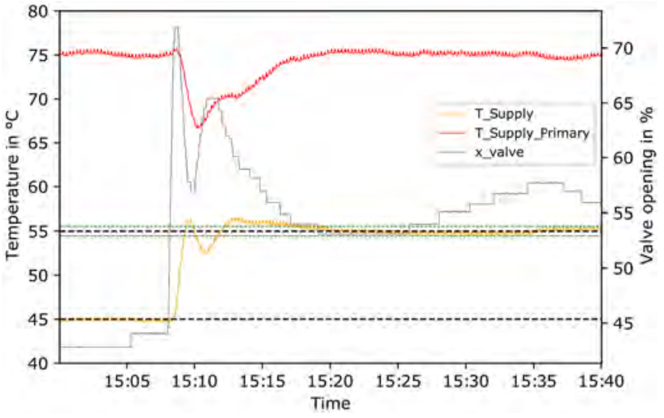

Fig. 8. Results of a step response test from 45 ˚C to 55 ˚C for a cloud controlled heating circuit.

5.2. *Results of the cloud controlled microgrid*

In the assumed scenario, a change of a load from 10 kW to 30 kW creates an active power imbalance in the microgrid. Phasor measurement values are published to the MGCC with a reporting rate of 100 Hz. The measurements are synchronized with respect to their timestamps in the phasor data concentrator (PDC) before they are fed to the controller.

Fig. 9 (top) shows the total load in the microgrid represented by the active and reactive powers. The active power load increases at 0.5 s. After that, both powers slightly fluctuate due to voltage fluctuations. This is expected in case of large disturbances in low inertia systems. One can see in Fig. 9 (middle) that the active power of the forming inverter initially increases quickly after the change of load at 0.5 s. At around 0.82 s, its power starts to decrease, while the active power of the grid

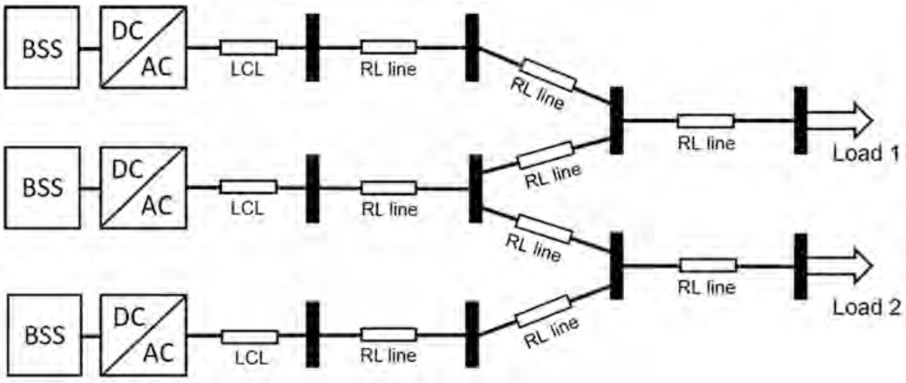

Fig. 7. Microgrid network with DC/AC inverters, battery storage systems (BSS), LCL filters, RL lines and loads.

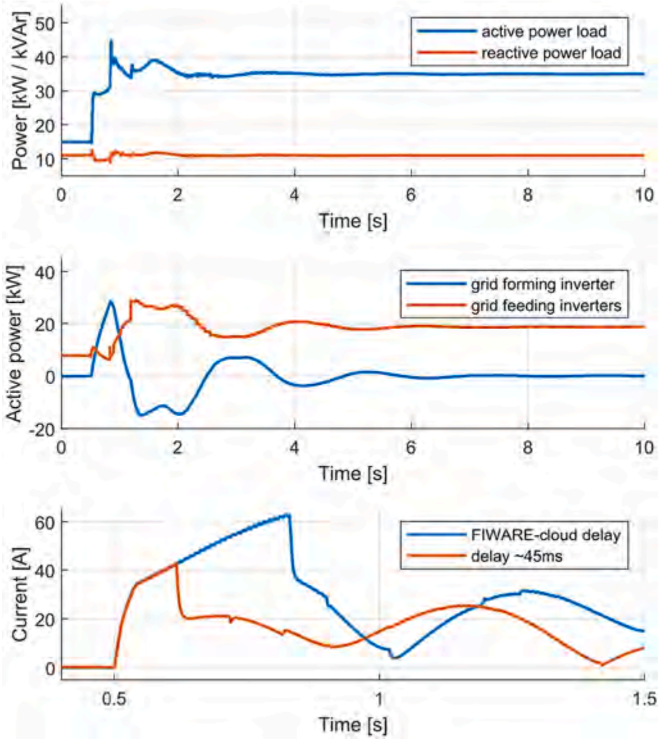

Fig. 9. Top: Load changes in the microgrid (large disturbance at 0.5 s), Middle: Power infeed of the inverters after disturbance. Bottom: Output current of the grid forming inverter).

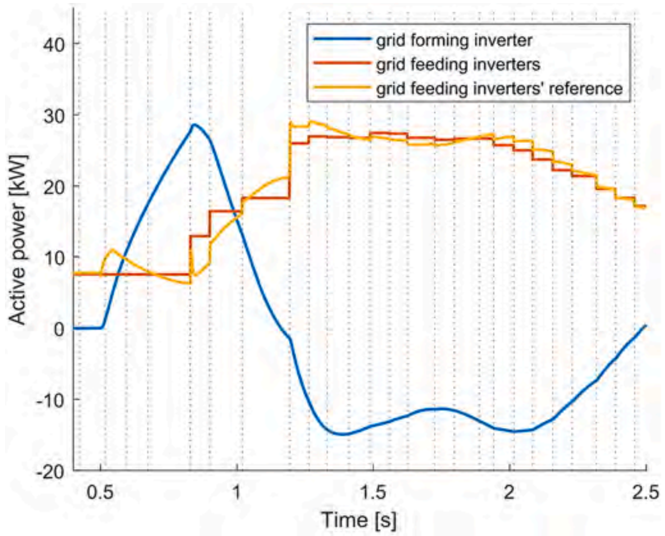

Fig. 10. Power output of inverters vs. reference power. Dashed lines indicate instants with new setpoints received from the MGCC.

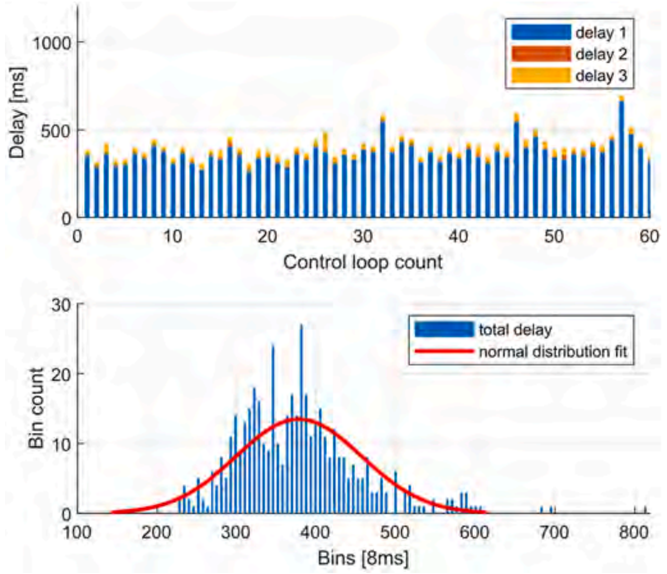

Fig. 11. Top: Three main delays in the control loop operation (60 samples for better visibility). Delay 1: From measurement timestamp to PDC. Delay 2: From PDC to controller output. Delay 3: From controller output to setpoints implementation. Bottom: Distribution of the total control loop delay.

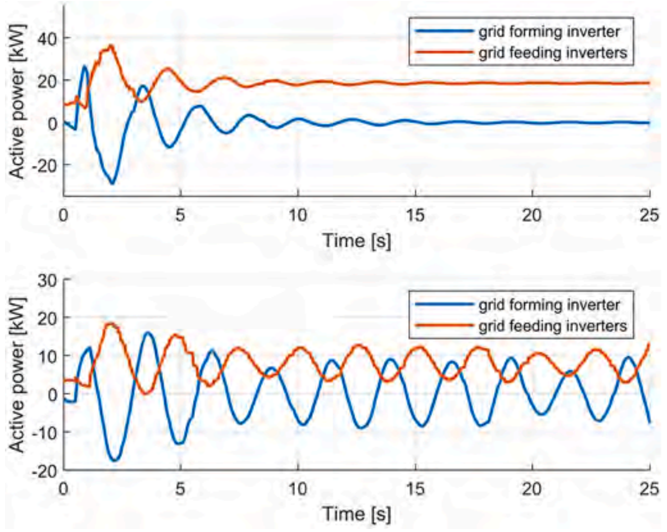

Fig. 12. Output active power of inverters in case of an emulated additional delay of 50 ms (top) and 100 ms (bottom).

feeding inverters increase in order to cover the imbalance. To observe the changes in the grid forming inverter more precisely, one can see its output current as blue line in Fig. 9 (bottom). It grows quickly after the load change and starts to decrease at 0.82 s, exactly when the grid feeding inverters receive a new setpoint from the MGCC and start to supply more active power. In Fig. 10, the discrete values of subsequent setpoints received by the grid feeding inverters are presented. The setpoints received at 0.82 s cause injection of more active power. However, due to the delays in the control loop, after the disturbance at 0.5 s, there are still new setpoints received, which do not yet request increase in power injection. Moreover, Fig. 10 indicates that the setpoints are received with irregular time delays, which is mostly caused by intermittent communication through the public internet.

Due to the influence of delays in the presented microgrid use case, several timestamps are analyzed in order to assess the performance of the FIWARE platform. In the control loop, one can distinguish different sources of delays between an event in the network (large change of the load) and the implementation of the setpoints by the grid feeding inverters. Such a delay of the entire control loop can include: (i) Delay due to the acquisition window in the PMU devices, (ii) delay in the processing of the algorithm in the PMU, (iii) delay from communication latency between the PMU and the platform, (iv) delay due to saving and reading of measurement data into and from the database, (v) delay in the processing of the phasor data concentrator, (vi) delay in the processing of the controller, and (vii) delay of communication latency between the controller and the devices when transmitting the setpoints.

Four timestamps during the operation are recorded: (i) Timestamps of the measurement devices, (ii) timestamps at the input of the PDC (before the controller), (iii) timestamps at the output of the controller, and (iv) timestamps at the moment of execution of the setpoints by the actuators. Thus, three delays between these timestamps can be calculated precisely. Fig. 11 (top) presents these three main delays in the control loop of the MGCC.

Fig. 11 (bottom) presents a histogram of 8 ms long bins of total delay and the fitting normal distribution function curve. The delay refers to the time of the entire control loop in the operation for the presented scenario. The average delay over a test of 3 min is equal to 381 ms with an average output controller sampling at 85 ms.

The MGCC is able to maintain stability in the microgrid even though the total measured delay is significant due to the communication with the platform over the public internet. Fig. 12 shows the waveforms of the inverters' active power in case the control setup is delayed by additional 50 ms and 100 ms, respectively. In the first case (50 ms), the inverters are still able to converge to a steady state; however, in case of the additional 100 ms delay, the system is only marginally stable and any further increase of delay can lead to globally instable behavior. We identify the maximum allowed delay in our tests at mean value around 480 ms (380 ms mean + 100 ms emulated additional delay) with a standard deviation of 79 ms, which is in accordance with [47]. We do not focus on further investigation of how stochasticity of the delay impacts the control performance and these values are considered marginal. Derivation or analytical identification of the exact number of the marginal delay is very challenging due to the variable nature of the delays and is not subject of this work.

With the assumed low-voltage power electronics devices, we consider hardware limits of 70 A output AC current. Thus, we conclude that the currently observed delays do not cause a violation of this constraint, as can be seen from Fig. 9 (bottom).

In Fig. 11 (top), one can observe that the vast majority of the overall delay is between the timestamps of the measurement devices and the PDC at the input of the controller. In case this delay is limited to an order of magnitude of the opposite way communication (from the controller to the actuators at around 25 ms), the MGCC would be able to operate far from its marginal delay area. Moreover, in case of such delay reduction, the sub-transient behavior of the current can be further curtailed, leading to much safer operation for the power electronics hardware. The current in such a case (giving approx. 45 ms total delay) is presented in Fig. 9 (bottom).

Comparing the performance of reaching steady-state between conventional primary droop controls, as described in [16], and the presented control in this work, it can be observed that both control strategies can achieve steady-state within seconds. As mentioned before, the presented communication-based control can reach any setpoint requested by the cloud controller, including, e. g. economically optimal ones. This distinguishes the presented cloud control from conventional local controllers, such as droop controllers, that require low-bandwidth communication for the secondary control layer in order to restore nominal values in the grid.

Concluding, the requirements for the microgrid cloud control use case are met. Thus, the FIWARE platform proves to be suitable for microgrid control applications.

6. Conclusion and outlook

In this work, we presented a FIWARE-based IoT platform that integrates control experiments with real and hybrid (real and simulation-based) applications in smart building and smart grid domains. The suitability of the platform enabling cloud control with security measures applied is determined through meeting use case specific requirements while the whole setup relies entirely on open source software.

For instance, in a typical building energy system application, we investigate the supply temperature control of a conventional heating circuit via PID control. Latter was deployed on some remote computer, detached from the platform. The communication between the heating circuit, the platform, and the controller was realized via the public internet and secured through the use of encryption, authentication, and authorization measures. Overall, the controller was capable to keep the control variable within the tolerance and the settling time is of equal magnitude as the dynamic of the electrical heater. Therefore, even with the manual parameter tuning in one operating point, the control is sufficient. However, especially when using online data, adaptive and automatic tuning offer great potential in optimizing the control and their use should be investigated further in future applications.

Similarly, in the demonstrated microgrid control experiment, both platform and controller managed the high frequency of incoming data, 100 phasors per second. The biggest delay arose between the measurement devices and the arrival of the measurement in the platform. In this work, the mean value maximum allowed delay is identified to 480 ms with a standard deviation of 79 ms. Within these delays, the controller still achieved to maintain the stability of the microgrid. Yet, the delays are still vital to the stability of the microgrid. If the delays increase the stability of the microgrid cannot be guaranteed and current hardware limitations could be exceeded and devices can take damage. Hence, further applications should consider measures to improve the performance of the communication between a field device and a secured platform.

In general, for future applications, the dense availability of data through the growing use of the IoT would make it possible to consider more sophisticated controls, e.g. data driven control and adaptive control. Through the use of these more sophisticated controls, the energy performance gap, not only in buildings but in a broad range of energy use cases, could be filled or narrowed. Since the deployment location of the control algorithm is not limited anymore, low performance hardware can be installed to transfer data from the building towards the cloud and reduce capital spending. Additionally, taking into account that microgrid controllers can receive any setpoint from anywhere, this bear the possibility to operate microgrids not only stable but also energy and economically optimal.

The use of open standards and other open source applications like FIWARE benefits data exchange. Elevating field information from the field layer to a platform layer with standardized interfaces benefits software development. E. g., because software developers do not have to learn use case specific interfaces and protocols. The availability of information combined with modern control strategies, open source tools, and standardized, secure interfaces and information exchange can generate new business cases which then can be quickly spread all over the world and accelerate the reduction of residual energy demands and enable sector-coupling. However, there is still a lack of standardization in the field of data modeling. Only through the use of generic data models there will be benefits from the use of standardized data protocols and structures: Interoperability through standardization.

Funding

This research is possible thanks to the support of the German Federal Ministry for Economic Affairs and Climate Action through 03EN1030B and its predecessor 03ET1561B in which most of the fundamental work was executed.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

We would like to thank our project partners from Technical University Dresden (03EN1030A and 03ET1561A) for the collaboration. Further, we would like to thank AutomationOne GmbH who developed and constantly works on improving the used edge device. Last, we would like to thank Michelle and Lucca whose support in the construction of the heating circuit test stand made the experiments possible.

Appendix A. Appendix

Table 2
Image name and version overview for used open source components available at https://hub.docker.com/.

Component Name	Image Name and Version
Orion Context Broker	fiware/orion:3.1.0
MongoDB	mongo:4.4
CrateDB	crate:4.5.0
QuantumLeap	orchestracities/quantumleap:0.8.1/
RedisDB	redis:6.2.6
MQTT Broker	karltga/mosquitto-go-auth-oauth2:v1.2/
IoT Agent JSON	fiware/iotagent-json:1.19.0
IoT Agent UltraLight 2.0	fiware/iotagent-ul:1.18.0
Keycloak / IDAM	jboss/keycloak:15.0.0
PEP Proxy	quay.io/gogatekeeper/gatekeeper:1.3.5/

References

[1] A. Ahmed, M.M. Khan, W. Ahmed, Cloud based network management and control for building automation, in: 2016 19th International Multi-Topic Conference (INMIC), 2016, pp. 1–6, https://doi.org/10.1109/INMIC.2016.7840123.
[2] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, M. Ayyash, Internet of things: a survey on enabling technologies, protocols, and applications, in: IEEE Communications Surveys & Tutorials 17, 2015, pp. 2347–2376. URL:.
[3] E. Al-Masri, K.R. Kalyanam, J. Batts, J. Kim, S. Singh, T. Vo, C. Yan, Investigating messaging protocols for the internet of things (IoT), IEEE Access 8 (2020) 94880–94911, https://doi.org/10.1109/ACCESS.2020.2993363.
[4] S.A. Al-Qaseemi, H.A. Almulhim, M.F. Almulhim, S.R. Chaudhry, IoT architecture challenges and issues: lack of standardization, in: 2016 Future Technologies Conference (FTC), 2016, pp. 731–738, https://doi.org/10.1109/FTC.2016.7821686.
[5] Q. Alfalouji, T. Schranz, A. Kümpel, M. Schraven, T. Storek, S. Gross, A. Monti, D. Müller, G. Schweiger, IoT middleware platforms for smart energy systems: an empirical expert survey, Buildings 12 (2022) 526. URL:.
[6] K. Amasyali, N.M. El-Gohary, A review of data-driven building energy consumption prediction studies, Renew. Sust. Energ. Rev. 81 (2018) 1192–1205. URL:.
[7] V. Araujo, K. Mitra, S. Saguna, C. Åhlund, Performance evaluation of FIWARE: a cloud-based IoT platform for smart cities, J. Parallel Distrib. Comput. 132 (2019) 250–261. URL:.
[8] D. Bandyopadhyay, J. Sen, Internet of Things: Applications and Challenges in Technology and Standardization, URL: `https://link.springer.com/article/10.1007/s11277-011-0288-5`, 2011 https://doi.org/10.1007/s11277-011-0288-5.
[9] A. Banks, E. Briggs, K. Borgendale, R. Gupta, MQTT Version 5.0, URL: `http://docs.oasis-open.org/mqtt/mqtt/v5.0/cos01/mqtt-v5.0-cos01.pdf`, 2018. Latest version: `https://docs.oasis-open.org/mqtt/mqtt/v5.0/mqtt-v5.0.html`. Last accessed: 2021-10-18.
[10] J. Berberich, J. Köhler, M.A. Müller, F. Allgöwer, On the design of terminal ingredients for data-driven MPC, IFAC-PapersOnLine 54 (2021) 257–263. URL: `https://www.sciencedirect.com/science/article/pii/S240589632101329X` https://doi.org/10.1016/j.ifacol.2021.08.554.
[11] S. Blechmann, K. Jansen, B. Scheuffele, M.H. Schraven, A. Kümpel, M. Pietersz, D. Müller, Best practices zur gateway-Entwicklung für internet of things - Anwendungen in der Gebäudeautomation, Technical report, RWTH Aachen University, 2021. URL:.
[12] J.M. Cantera Fonseca, F.G. Márquez, T. Jacobs, FIWARE-NGSI v2 Specification, URL: `https://fiware.github.io/specifications/ngsiv2/stable/`.
[13] F. Cirillo, G. Solmaz, E.L. Berz, M. Bauer, B. Cheng, E. Kovacs, A standard-based open source IoT platform: FIWARE, IEEE Int. Things Mag. 2 (2019) 12–18. URL:.
[14] B. Dave, A. Buda, A. Nurminen, K. Främling, A framework for integrating BIM and IoT through open standards, Autom. Constr. 95 (2018) 35–45. URL:.
[15] R. De Coninck, L. Helsen, Practical implementation and evaluation of model predictive control for an office building in Brussels, Energy Build. 111 (2016) 290–298. URL: `https://www.sciencedirect.com/science/article/pii/S0378778815303790` https://doi.org/10.1016/j.enbuild.2015.11.014.
[16] S. Eberlein, K. Rudion, Optimisation, benchmark testing and comparison of droop control variants in microgrids, IET Smart Grid 4 (2021) 536–548. URL: `https://onlinelibrary.wiley.com/doi/abs/10.1049/stg2.12036` https://doi.org/10.1049/stg2.12036. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1049/stg2.12036.
[17] ETSI, Context Information Management (CIM); NGSI-LD API, URL: `https://www.etsi.org/deliver/etsi_gs/CIM/001_099/009/01.04.02_60/gs_CIM009v010402p.pdf`, 2021. rGS/CIM-0009v142.
[18] FIWARE, FIWARE GitHub Repositories, URL, `https://github.com/FIWARE`, 2021. Last accessed: 2021-09-24.
[19] FIWARE, What Is FIWARE - About Us, URL, `https://www.fiware.org/about-us/`, 2021.
[20] E. Fotopoulou, A. Zafeiropoulos, F. Terroso-Sáenz, U. Simsek, A. González-Vidal, G. Tsiolis, P. Gouvas, P. Liapis, A. Fensel, A. Skarmeta, Providing personalized energy management and awareness services for energy efficiency in smart buildings, Sensors 17 (2017) 2054. URL: `https://www.mdpi.com/1424-8220/17/9/2054` https://doi.org/10.3390/s17092054. number: 9 Publisher: Multidisciplinary Digital Publishing Institute.
[21] A. Ghosal, M. Conti, Key management systems for smart grid advanced metering infrastructure: a survey, IEEE Commun. Surv. Tutor. 21 (2019) 2831–2848, https://doi.org/10.1109/COMST.2019.2907650, conference Name: IEEE Communications Surveys Tutorials.
[22] J.M. Guerrero, M. Chandorkar, T.L. Lee, P.C. Loh, Advanced control architectures for intelligent microgrids—part I: decentralized and hierarchical control, IEEE Trans. Indus. Electron. 60 (2013) 1254–1262, https://doi.org/10.1109/TIE.2012.2194969, conference Name: IEEE Transactions on Industrial Electronics.
[23] M. Haghgoo, A. Dognini, T. Storek, R. Plamanescu, U. Rahe, S. Gheorghe, M. Albu, A. Monti, M. Dirk, Open smart energy eco-system for the future, IOP Conf. Ser. 588 (2020), 022048. URL:.
[24] M. Haghgoo, A. Dognini, T. Storek, R. Plamanescu, U. Rahe, S. Gheorghe, M. Albu, A. Monti, D. Müller, A cloud-based service-oriented architecture to unlock smart energy services, Energy Inform. 4 (2021) 8. URL: https://doi.org/10.1186/s42162-021-00143-x.
[25] C. Haissig, Adaptive fuzzy temperature control for hydronic heating systems, in: Proceedings of the 1999 IEEE International Conference on Control Applications (Cat. No.99CH36328) vol. 1, 1999, pp. 582–588, https://doi.org/10.1109/CCA.1999.806711.
[26] I. Hazyuk, C. Ghiaus, D. Penhouet, Model predictive control of thermal comfort as a benchmark for controller performance, Autom. Constr. 43 (2014) 98–109. URL:.
[27] G.P. Henze, D.E. Kalz, S. Liu, C. Felsmann, Experimental analysis of model-based predictive optimal control for active and passive building thermal storage inventory, HVAC&R Res. 11 (2005) 189–213. URL:.
[28] A. Kümpel, F. Stinner, B. Gauch, M. Baranski, D. Müller, A Representative Simulation Model for Benchmarking Building Control Strategies, Banff, AB, Canada, URL: `http://www.iaarc.org/publications/2019_proceedings_of_the_36th_isarc/a_representative_simulation_model_for_benchmarking_building_control_strategies.html`, 2019 https://doi.org/10.22260/ISARC2019/0033.

[29] A. Kümpel, T. Storek, M. Baranski, M. Schumacher, D. Müller, A cloud-based operation optimization of building energy systems using a hierarchical multi-agent control, J. Phys. Conf. Ser. 1343 (2019), 012053. URL:.

[30] D. Lindelöf, H. Afshari, M. Alisafaee, J. Biswas, M. Caban, X. Mocellin, J. Viaene, Field tests of an adaptive, model-predictive heating controller for residential buildings, Energy Build. 99 (2015) 292–302. URL: `https://www.sciencedirect.com/science/article/pii/S037877881500328X` https://doi.org/10.1016/j.enbuild.2015.04.029.

[31] Y. Lu, Industry 4.0: a survey on technologies, applications and open research issues. Journal of industrial information, Integration 6 (2017) 1–10. URL:.

[32] R. Martínez, J.Á. Pastor, B. Álvarez, A. Iborra, A testbed to evaluate the FIWARE-based IoT platform in the domain of precision agriculture, Sensors 16 (2016) 1979. URL:.

[33] P. Martins, S.I. Lopes, A. Curado, Designing a FIWARE-based smart campus with IoT edge-enabled intelligence, in: Á. Rocha, H. Adeli, G. Dzemyda, F. Moreira, A. M. Ramalho Correia (Eds.), Trends and Applications in Information Systems and Technologies vol. 1367, Springer International Publishing, Cham, 2021, pp. 557–569. URL:.

[34] J. Mayer, V. Pampana, M. Bernard, D. Bytschkow, T. Stohl, P. Gupta, M. Duchon, Holonic architectures for IoT-empowered energy management in districts, in: 2021 IEEE 7th World Forum on Internet of Things (WF-IoT), 2021, pp. 189–194, https://doi.org/10.1109/WF-IoT51360.2021.9595252.

[35] D. Minoli, K. Sohraby, B. Occhiogrosso, IoT considerations, requirements, and architectures for smart buildings—energy optimization and next-generation building management systems, IEEE Int. Things J. 4 (2017) 269–283, https://doi.org/10.1109/JIOT.2017.2647881, conference Name: IEEE Internet of Things Journal.

[36] S.P. Mohanty, U. Choppali, E. Kougianos, Everything you wanted to know about smart cities: the internet of things is the backbone, IEEE Cons. Electron. Mag. 5 (2016) 60–70, https://doi.org/10.1109/MCE.2016.2556879, conference Name: IEEE Consumer Electronics Magazine.

[37] F. Montori, L. Bedogni, L. Bononi, A collaborative internet of things architecture for smart cities and environmental monitoring, IEEE Int. Things J. 5 (2018) 592–605, https://doi.org/10.1109/JIOT.2017.2720855. conference Name: IEEE Internet of Things Journal.

[38] B. Morvaj, L. Lugaric, S. Krajcar, Demonstrating smart buildings and smart grid features in a smart energy city, in: Proceedings of the 2011 3rd International Youth Conference on Energetics (IYCE), 2011, pp. 1–8.

[39] Z. O'Neill, Y. Li, K. Williams, HVAC control loop performance assessment: a critical review (1587-RP), Sci. Technol. Built Environ. 23 (2017) 619–636. URL: https://doi.org/10.1080/23744731.2016.1239466. publisher: Taylor & Francis _eprint:.

[40] D. Pan, X. Wang, F. Liu, R. Shi, Transient stability of voltage-source converters with grid-forming control: a design-oriented study, IEEE J. Emerg. Sel. Top. Power Electron. 8 (2020) 1019–1033, https://doi.org/10.1109/JESTPE.2019.2946310, conference Name: IEEE Journal of Emerging and Selected Topics in Power Electronics.

[41] R. Plamanescu, M. Albu, S. Gheorghe, S. Bugnar, M. Coroiu, PMU cloud-based applications for power systems insight, in: 2019 54th International Universities Power Engineering Conference (UPEC), 2019, pp. 1–5, https://doi.org/10.1109/UPEC.2019.8893631.

[42] R. Plamanescu, S. Gheorghe, M. Sarb, A. Istenes, M. Albu, A synchronized measurements Fiware platform for smart grid applications, in: 2019 11th International Symposium on Advanced Topics in Electrical Engineering (ATEE), 2019, pp. 1–5, https://doi.org/10.1109/ATEE.2019.8724897 (iSSN: 2159-3604).

[43] A. Pozo, Á. Alonso, J. Salvachúa, Evaluation of an IoT application-scoped access control model over a publish/subscribe architecture based on FIWARE, Sensors 20 (2020) 4341. URL:.

[44] B.L. Risteska Stojkoska, K.V. Trivodaliev, A review of internet of things for smart home: challenges and solutions, J. Clean. Prod. 140 (2017) 1454–1464. URL:.

[45] J. Rocabert, A. Luna, F. Blaabjerg, P. Rodríguez, Control of power converters in AC microgrids, IEEE Trans. Power Electron. 27 (2012) 4734–4749, https://doi.org/10.1109/TPEL.2012.2199334, conference Name: IEEE Transactions on Power Electronics.

[46] F. Salem, M.I. Mosaad, A comparison between MPC and optimal PID controllers: case studies, Michael Faraday IET Int. Summit 2015 (2015) 59–65, https://doi.org/10.1049/cp.2015.1607.

[47] I. Serban, S. Céspedes, C. Marinescu, C.A. Azurdia-Meza, J.S. Gómez, D. S. Hueichapan, Communication requirements in microgrids: a practical survey, IEEE Access 8 (2020) 47694–47712, https://doi.org/10.1109/ACCESS.2020.2977928, conference Name: IEEE Access.

[48] X. Shi, B. Si, J. Zhao, Z. Tian, C. Wang, X. Jin, X. Zhou, Magnitude, causes, and solutions of the performance gap of buildings: a review, Sustainability 11 (2019) 937. URL:.

[49] L.R. da Silva, R.C.C. Flesch, J.E. Normey-Rico, Controlling industrial dead-time systems: when to use a PID or an advanced controller, ISA Trans. 99 (2020) 339–350. URL:.

[50] A. Stanford-Clark, A. Nipper, MQTT - The Standard for IoT Messaging, URL: `https://mqtt.org/`, 1999. Last accessed: 2021-05-01.

[51] J.A. Stankovic, Research directions for the internet of things, IEEE Internet Things J. 1 (2014) 3–9, https://doi.org/10.1109/JIOT.2014.2312291, conference Name: IEEE Internet of Things Journal.

[52] T. Storek, S. Bano, D. Nikolay, FiLiP - Fiware library for Python, URL: `https://github.com/RWTH-EBC/FiLiP`, 2021.

[53] T. Storek, J. Lohmöller, A. Kümpel, M. Baranski, D. Müller, Application of the open-source cloud platform FIWARE for future building energy management systems, J. Phys. Conf. Ser. 1343 (2019), 012063. URL:.

[54] D. Sturzenegger, D. Gyalistras, M. Morari, R.S. Smith, Model predictive climate control of a swiss office building: implementation, results, and cost–benefit analysis, IEEE Trans. Control Syst. Technol. 24 (2016) 1–12, https://doi.org/10.1109/TCST.2015.2415411, conference Name: IEEE Transactions on Control Systems Technology.

[55] D. Syed, A. Zainab, A. Ghrayeb, S.S. Refaat, H. Abu-Rub, O. Bouhali, Smart grid big data analytics: survey of technologies, techniques, and applications, IEEE Access 9 (2021) 59564–59585, https://doi.org/10.1109/ACCESS.2020.3041178, conference Name: IEEE Access.

[56] S. Talari, M. Shafie-khah, P. Siano, V. Loia, A. Tommasetti, J.P.S. Catalão, A review of smart cities based on the internet of things concept, Energies 10 (2017) 421. URL:.

[57] F. Terroso-Saenz, A. González-Vidal, A.P. Ramallo-González, A.F. Skarmeta, An open IoT platform for the management and analysis of energy data, Futur. Gener. Comput. Syst. 92 (2019) 1066–1079. URL:.

[58] United Nations Environment Programme (2021), 2021 Global Status Report for Buildings and Construction: Towards a Zero-emission, Efficient and Resilient Buildings and Construction Sector, Technical Report. Nairobi. URL: `http://www.unep.org/resources/report/2021-global-status-report-buildings-and-construction`, 2021. library Catalog: www.unep.org Section: publications.

[59] G. Vaglica, F. Bono, G. Renaldi, A JRC FIWARE testbed for SMART building and infrastructures: Implementation of the FIWARE platform for performance testing and heterogeneous sensor nodes, in: Technical Report JRC118952; EUR 30038 EN, European Commission Joint Research Centre, 2020. LU. URL: https://data.europa.eu/doi/10.2760/314017.

[60] F. de la Vega, J.P. García-Martín, G.P. Santos, A. Torralba, Implementation of a Fiware-based integration platform and a web portal as aids to improve the control of ships navigation in a river, in: 2020 IEEE International Symposium on Systems Engineering (ISSE), 2020, pp. 1–3, https://doi.org/10.1109/ISSE49799.2020.9272242 (iSSN: 2687-8828).

[61] W. Velasquez, L. Tobar-Andrade, I. Cedeno-Campoverde, Monitoring and data processing architecture using the FIWARE platform for a renewable energy systems, in: 2021 IEEE 11th Annual Computing and Communication Workshop and Conference (CCWC), 2021, pp. 1383–1387, https://doi.org/10.1109/CCWC51732.2021.9376026.

[62] M. Wallace, R. McBride, S. Aumi, P. Mhaskar, J. House, T. Salsbury, Energy efficient model predictive building temperature control, Chem. Eng. Sci. 69 (2012) 45–58. URL:.

[63] P. de Wilde, The gap between predicted and measured energy performance of buildings: a framework for investigation, Autom. Constr. 41 (2014) 40–49. URL:.

[64] A. Zanella, N. Bui, A. Castellani, L. Vangelista, M. Zorzi, Internet of things for smart cities, IEEE Internet Things J. 1 (2014) 22–32, https://doi.org/10.1109/JIOT.2014.2306328, conference Name: IEEE Internet of Things Journal.

[65] S. Zhan, A. Chong, Data requirements and performance evaluation of model predictive control in buildings: a modeling perspective, Renew. Sust. Energ. Rev. 142 (2021), 110835. URL:.

[66] D. Zhang, X. Huang, D. Gao, X. Cui, N. Cai, Experimental study on control performance comparison between model predictive control and proportion-integral-derivative control for radiant ceiling cooling integrated with underfloor ventilation system, Appl. Therm. Eng. 143 (2018) 130–136. URL:.

[67] X. Zhang, J. Lei, M. Zhang, Research on control method of valve-controlled heating system, J. Phys. Conf. Ser. 1345 (2019) 032081. URL: https://doi.org/10.1088/1742-6596/1345/3/032081. publisher: IOP Publishing.

[68] S. Žitnik, M. Janković, K. Petrovčič, M. Bajec, Architecture of standard-based, interoperable and extensible IoT platform, in: 2016 24th Telecommunications Forum (TELFOR), 2016, pp. 1–4, https://doi.org/10.1109/TELFOR.2016.7818915.