



Microgrid management system based on a multi-agent approach: An office building pilot

Luis Gomes^{a,*}, Zita Vale^b, Juan M. Corchado^{c,d,e}

^a GECAD – Research Group on Intelligent Engineering and Computing for Advanced Innovation and Development Polytechnic of Porto (IPP), Porto, Portugal

^b Polytechnic of Porto (IPP), Porto, Portugal

^c BISITE Research Group, University of Salamanca, Salamanca 37007, Spain

^d Air Institute, IoT Digital Innovation Hub, Salamanca 37188, Spain

^e Department of Electronics, Information and Communication, Osaka Institute of Technology, Osaka 535–8585, Japan

ARTICLE INFO

Article history:

Received 19 August 2019

Received in revised form 11 November 2019

Accepted 16 December 2019

Available online 21 December 2019

Keywords:

Microgrid deployment

Microgrid management

Multi-agent systems

Real-time energy management

ABSTRACT

Microgrids bring advantages to end-users and to the smart grid environment. However, adequate management software, enabling bringing to the field new energy management concepts, is not available yet. Small, single-tasked, software is usually proposed and tested while a clear overall system architecture for microgrid management required to take full advantage of the microgrids' potential. Previous publications usually focus on energy-related problems and do not provide an efficient and viable solution for players' representation and microgrid operation. This paper proposes a complete architecture for a microgrid management system based on a multi-agent approach – μ GIM – allowing the easy implementation of different energy strategies. The μ GIM agents can independently manage local resources while able to collaborate and/or compete with other agents. Designed to run in single-board computers, μ GIM agents are light-weighted and easily deployed in buildings. To demonstrate these capabilities, the paper details and presents a microgrid deployment using μ GIM solution.

© 2020 Elsevier Ltd. All rights reserved.

1. Introduction

The new paradigm of power energy systems enables and benefits small and medium players that have active participation roles in the smart grid context. These players can see their energy bill increased or decreased according to the efficiency of their participation towards common and/or global goals. The ways to participate in the smart grid are vast, complex, and time-consuming. Therefore, such participation must be enabled and seamless for the users.

In the new power energy system paradigm, Demand Side Management (DSM) can be used to eliminate grid peaks and to accommodate a higher penetration of renewable energies [1]. For this to be possible, new approaches must be researched to provide end-users with these abilities.

Microgrids appear in the smart grid context as “electricity distribution systems containing loads and distributed energy resources (such as distributed generators, storage devices, or controllable loads) that can be operated in a controlled, coordinated way either while

connected to the main power network or while islanded” [2]. These smaller grids bring management advantages and are being deployed around the world, where some examples can be found in University of California in San Diego [3], in the Brooklyn area in New Jersey in the United States of America [4], in Sendai in Japan [5], and in the Nanji island in China [6].

Regarding energy management systems, single-board computers (SBC) can bring new possibilities for development and deployment [7]. Their capacity to integrate an operating system (OS) enables easy access to a high level of computation while providing input/output pins similar to a microcontroller. Therefore, some works use SBC for deployment. In Ref. [8], a fire-safety sensor is deployed in a Raspberry Pi enabling web services and a TCP/IP-based sensor communication. In a more recent work presented in Ref. [9], SBC are used to provide a hierarchical and distributed multi-agent system for microgrid control merging the standard IEC 61850, FIPA specifications and data distribution service (DDS) standards.

The main contributions of this paper are an agent-based architecture for end-user representation, a multi-agent system for microgrid management, and a light-weight solution that can be easily deployed in SBC. This work differs from others by providing a unique agent-architecture that can model any end-user, avoiding

* Corresponding author.

E-mail addresses: lufog@isep.ipp.pt, jafps@isep.ipp.pt (L. Gomes), zav@isep.ipp.pt (Z. Vale), corchado@usal.es (J.M. Corchado).

the development and deployment of multiple agents with different software; one agent can represent them all. Another key aspect of this proposed solution, it is the ability of agents to be executed as standalone systems. The standalone implementation allows the proposed solution to manage the building resources without being a part of a multi-agent system.

This paper proposes a complete architecture for a microgrid management system based on a multi-agent approach. This architecture is focused on the end-user and enables the monitoring, management, and control of energy resources. The proposed architecture allows the integration of sensors to enable context-awareness and efficient resource optimization. The proposed platform architecture is named μ GIM, from microgrid intelligent management. The proposed modular platform/architecture enables the integration of different energy strategies that provides users with efficient management, intelligence and active participation in the smart grid. This paper will present the proposed platform and will not focus on modular energy strategies.

The proposed architecture and platform were developed and implemented in SBC and deployed in an office building, where the platform is tested, and the data can be analyzed. Some of these data are shown in this paper. The complete dataset can be accessed in Refs. [10] and [11]. The office building where μ GIM platform is deployed has offices rented; renters are responsible for their own energy management. The results show the ability of the μ GIM platform to efficiently monitor and control an office microgrid.

This paper is divided into six main sections. After this Introduction section, Related Work is presented in Section 2. The proposed μ GIM Platform is described in Section 3. The office building deployment is described in Section 4, and the main results are presented in Section 5. Finally, Section 6 presents the main conclusions of this work.

2. Related work

MAS are commonly applied in the smart grid context, as seen in Ref. [12]. Their ability to individually manage a part of the system while integrating a community enables the smart grid's players to be represented by agents; where each player has its own goals, but it also can collaborate or compete with other agents for common goals.

A MAS for smart grid management is proposed in Refs. [13,14], integrating MASCEM [15] and MASGriP [16] – a MAS for competitive electricity market simulation and a MAS for smart grid simulation, respectively. Later, MASGriP was implemented in the Multi-Agent System Layer proposed in Ref. [17] to monitor and control a university campus microgrid. A MAS for microgrid balance between consumption and generation is proposed in Ref. [18], where day-ahead forecasts are used for market participation and 5 min forecasts are used to balance the energy. The system uses a real-time digital power system simulator (RTDS) for grid simulation and a MAS for grid/player representation and management. A distributed voltage control system using a MAS is proposed in Ref. [19]. The MAS uses blackboard service for agent collaboration reducing data's communication between agents. Also, cooperative frequency control for islanded microgrids is proposed in Ref. [20], where a hierarchical MAS frequency control is used.

Although MAS are used for microgrids, they are usually applied in small and specific problems, not behind able to monitor and manage a player infrastructure. Also, it is common to use MAS with multiple agent types where each agent is responsible for a part of the system, a special task, or an individual energy resource. In Refs. [21] and [22] a MAS is proposed for building representation in a microgrid where five agent types are used: central coordinator

agent, building management agent, renewable energy resources agent, battery bank agent, and service agent. Five different types are also proposed in Ref. [23]: load agent, renewable distributed energy resource agent, conventional distributed energy resource agent, energy storage agent, and utility grid agent. A total of ten agents are proposed in Ref. [24]: distributed generation agent, load agent, storage agent, SCADA agent, generation coordination control agent, economic dispatch control agent, energy forecast agent, system operator agent, and static switch agent. More agent types are proposed in Refs. [25], [26] and [27]. This paper proposes a MAS with only one agent type that can represent any microgrid's player. The use of the same agent type improves scalability and deployment of the system.

To improve the deployment of the proposed MAS, SBC are used, enabling the easy installation of devices in the players' infrastructure. SBC can even be installed inside the electrical enclosure enabling efficient deployments in buildings. The use of SBC in the smart grid context is not a novelty; they are been applied to solve several tasks throughout the management layers of the smart grid. For instance, in Ref. [28], multiple Raspberry Pi are used to detect anomalies in phasor measurement units. BeagleBone Black SBC are used in Ref. [29] to implement the multi-level microgrid control proposed in Ref. [30]. In Ref. [29] RS485 and Controller Area Network (CAN) standards are used for resource integration, and the system is implemented in a Power Park with multiple microgrids.

To find complete solutions, using MAS and SBC, we need to look at broader projects, such as RIAPS and VOLTRON. Currently, these are the two main solutions related to the system proposed in this paper.

The Linux Foundation launched in the middle of 2018 the LF Energy that is an open-source coalition for multi-vendor collaboration and open-source progress in the sectors of energy and electricity [31]. The LF Energy launched four open-source projects, and one of those projects is the Resilient Information Architecture Platform for Smart Grid (RIAPS) that provides core services for buildings and microgrids. The RIAPS is presented as a software platform for building distributed real-time embedded applications [32].

The RIAPS is based on an extension of F6COM computation model [33,34]. It is a decentralized solution where nodes can join and leave groups anytime – using OpenDHT (open Distributed Hash Table) for discovery service [32].

The RIAPS platform has two node types: the control node, and the target node. The control node should be placed in the network control room and is mandatory in the RIAPS platform. For other deployments, the target node should be used. Each node is an actor (e.g. consumer or generator), and all of them are synchronized by time.

The platform provides an already set up control node running in an amd64 Xubuntu 18.04 virtual machine and a target node running in an armhf Ubuntu 18.04 for BeagleBone Black, a single-board computer from the BeagleBoard.org Foundation. Both can be freely downloaded from their webpage.¹ Instructions are available for users that want to use RIAPS in other environments.

Some of RIAPS applications and demonstrations can be seen in Refs. [12,35]. Particularly, in Ref. [12], RIAPS is used to implement a distributed under-frequency load shedding scheme. The implementation is tested in the IEEE 39-bus network using MATLAB software; in this case, the test is conducted outside RIAPS nodes. A distributed microgrid phase and frequency synchronization strategy is proposed in Ref. [35] and tested in a laboratory using real-time simulators (OPAL-RT 5031 and OPAL-RT 5607). In Ref. [36], RIAPS integrates an algorithm to interleaving multi-converter

¹ <https://riaps.isis.vanderbilt.edu/downloads.html>

systems to minimize harmonics in the DC bus. This work was also tested using a real-time simulator (OPAL-RT 5607). A small transactive energy demonstration, using only one consumer and one generator is presented in Ref. [37].

VOLTRON was firstly developed in 2012 by the Pacific Northwest National Laboratory, United States of America [38]. In 2014, VOLTRON was released as an open source software under Apache License (Version 2.0), the same license as RIAPS platform. Currently, VOLTRON is in its version 4. VOLTRON is an agent-based platform running in Python 2.7. Agents can be deployed in single-board computers, such as Raspberry Pi Zero W.

VOLTRON applications are distributed and centralized nodes are not needed. However, centralized agents can be deployed as service providers, such as weather services and VOLTRON Central Agent – an agent that provides a web service working as a global interface for all platform's agents. The main architecture defines three agent roles: platform agents, cloud agents, and control agents [39]. Each agent can perform one or several roles simultaneously. The platform agents provide services. The cloud agents are able to subscribe to local message topics and publish the data in remote servers. The control agent is able to control a resource while interacting with other resources.

VOLTRON agents use drivers to connect to physical resources. Modbus and BACnet are two available drives. However, being an open source solution, it allows the development of other drivers, such as the Xbee driver developed in Ref. [40].

Some of the deployments and use cases developed and implemented in VOLTRON can be seen in Ref. [41]. A hardware-in-the-loop (HIL) simulator for testing two variable-air-volume fan control is proposed in Ref. [42]. This HIL simulator was developed in the PNNL laboratories. A transactive control over rooftop air conditioning units using two demand response programs, demand limiting, and real-time pricing control, is proposed in Ref. [43]. The control is done using the building occupancy and the proposed solution was implemented in Washington State (United States of America) using single-board computers. In Ref. [44], VOLTRON is implemented in a Linux desktop machine with sixteen agents deployed in Cubietruck boards for a consensus-based distributed algorithm for energy price between consumers and generators; this work presents a simulated test with 1.400 agents. A grid deployment for transactive energy using VOLTRON agents can be seen in Refs. [45] and [46] where a total of eight Raspberry Pi boards are used – four deployed in buildings, one in the 1 MW photovoltaic array and one in a battery energy storage system. This deployment is made in the Scott Park Campus of the University of Toledo.

3. μ GIM platform

The solutions analyzed in the previous section demonstrate the current state of the art regarding ready to install platforms that can be executed in single-board computers. Specially RIAPS and VOLTRON are complete solutions that focus on energy management and can be easily deployed. The proposed μ GIM platform will differ from the previous solutions regarding their focus and actuation target. The μ GIM platform is focused in the end-user representation and can be installed as a standalone solution capable of providing energy management solutions to the end-user. One of the main novelties of μ GIM is its capability of integrating energy strategies that can be executed on the end-users' side.

While the previous solutions seemed to be built from the grid to end-users, μ GIM is designed from end-users to the grid. This new approach puts μ GIM closer to the end-user and enables the use of μ GIM without having an agent community. The μ GIM agents are designed to manage a smart building focusing on energy man-

agement, with external integration, participation, and negotiation capabilities. A μ GIM agents are capable of participating in a smart grid context by responding to demand response programs and by transacting energy. However, they use these external interactions to improve the end-users' buildings energy management, not needing or depending on these external agents. This shift of paradigm provides μ GIM with a novel solution that can be used today in our residential and nonresidential buildings providing resource optimization algorithms, while previous solutions almost demand the existence of a community in order to improve end-users' daily energy management.

Being end-user focused, μ GIM agents are able to represent an entire building/home and not smart grid or building parts, as seen in the previously mentioned examples of RIAPS and VOLTRON. The way μ GIM is able to do this will be presented in this section, while the results of such representation will be shown in Section 5.

The examples mentioned in the previous section provided case studies that usually dealt with equipment synchronization, building monitoring or large equipment – such as Heating, Ventilation, and Air Conditioning (HVAC) systems. In the case of the μ GIM platform, the priority is given to the management of small resources that are available in the end-users' building. This paradigm raises new problems regarding users' comfort, preferences, actions, and contexts.

The μ GIM agents' domain model can be seen in Fig. 1, where different colors are used to identify conceptual classes regarding building's agent, building's infrastructure, data management, management of the building, and physical interfaces. At its center is the agent/building that represents the end-user facility.

Each agent is able to represent a building; therefore, the agent can be seen as a building (central block). The agent has all the resources directly connected to it. Resources can be of different types, such as sensors for context and environment identification. Using rooms is possible to independently identify contexts inside the room and to make room optimizations. The storage system enables data storage of all resources and agent actions – locally and remotely. The Universal Asynchronous Receiver-Transmitter (UART) enables the connection of μ GIM agents in RS485 networks using Modbus/RTU protocol. Management strategies are encoded in algorithms that are executed inside the agent and that can have or not have an impact on the physical resources. Strategies can be of two types: periodic strategies are executed periodically in time, while context strategies are executed every time a particular context is met. Alarms are scheduled tasks inside the agent that execute actions; they are used for remote storage and scheduled strategies. All these aspects are explained in more detail below.

The authors developed the domain model of the μ GIM agents in the architecture presented in Fig. 2, using a component-based architecture methodology. The μ GIM agents are built on top of the Raspbian operating system. The dark orange blocks represent third-party software that is needed for the μ GIM platform but is not developed for/by μ GIM. The μ GIM agents are developed in Java 8 using the DevOps methodology for software development. Each light blue block, represented in Fig. 2, is, in fact, an individual thread that is executed in parallel. Internally, threads share data among them. The Java language was used because of the JADE framework and the ability to run in multiple operating systems. Not being the fastest programming language, Java is not the only language used in μ GIM agents; energy strategies can be developed or use other frameworks running in faster programming languages, such as the use of C in the GNU Linear Programming Kit (GLPK) solver.

However, μ GIM agents can use other languages for optimization and energy management, such as the case of C language used for is computational speed.

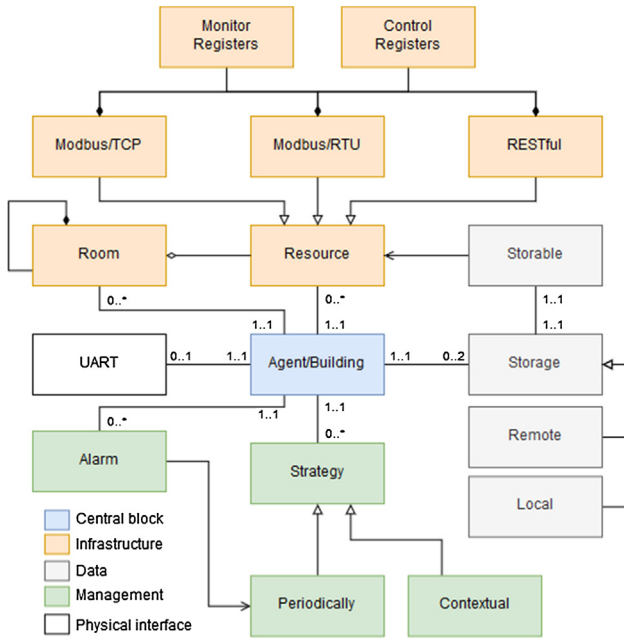


Fig. 1. Agent domain model.

For network discovery, μ GIM uses the JGroups block. This enables virtual tunneling in the network where agents can discover each other. In the JADE framework – the framework used for the proposed MAS – each agent must connect to the framework. JGroups enables the broadcast of the framework location, its Internet Protocol (IP) address, port, and name. The JADE Controller block can assume one of two modes per agent: Agent core, and MAS core. The agent core is assumed by default and indicates that that agent is part of a MAS. The MAS core mode indicates that that agent is

also the provider of the MAS framework where other agents will connect to. JADE's Agent Management System (AMS) and Directory Facilitator (DF) agent will be executed in the MAS core SBC. Per microgrid, only one MAS core must be used. If MAS core agent fails, the MAS will stop working because all agents will lose the connection AMS and DF agents. This JADE limitation should be overpassed to provide agent migration in case of failures; JGroups can be used to overcome JADE failures.

The Internal Memory block enables an agent memory unit that saves key/value objects where the key is a String and the value is an object. This block enables the sharing of data between threads that are not directly connected.

The Logging block is essential for development and logs all error and warning occurrences in the agent. In μ GIM, the Logging block can be configured to show the logs directly in the command line or to save the logs in files. The use of files is needed to enable the creation of command-line user interfaces. The GUI block (Graphical User Interface) is designed to have two possible configurations: local, and remote. Both can be used simultaneously. The local GUI uses a command-line GUI – see Fig. 3. The local GUI makes sense when a screen is available in the agent location. The remote GUI instantiates a μ GIM RESTful API (Application Programming Interface) that provides the monitoring and control of the μ GIM agent. From the design point-of-view, the remote GUI makes more sense because it provides a better visualization to users and enables the use of the interface in a web browser; mobile or in a computer.

The Data Storage is divided into two types; agents must have these two configuration types. The local storage uses a database installed in the localhost and uses the SBC internal memory. The remote storage uses an external server. The combination between the two types enables the existence of data inside the SBC – with a high-speed reading rate – and the existence of historical data in a server without the memory limitation of SBC. The μ GIM agents can also read from the remote storage system.

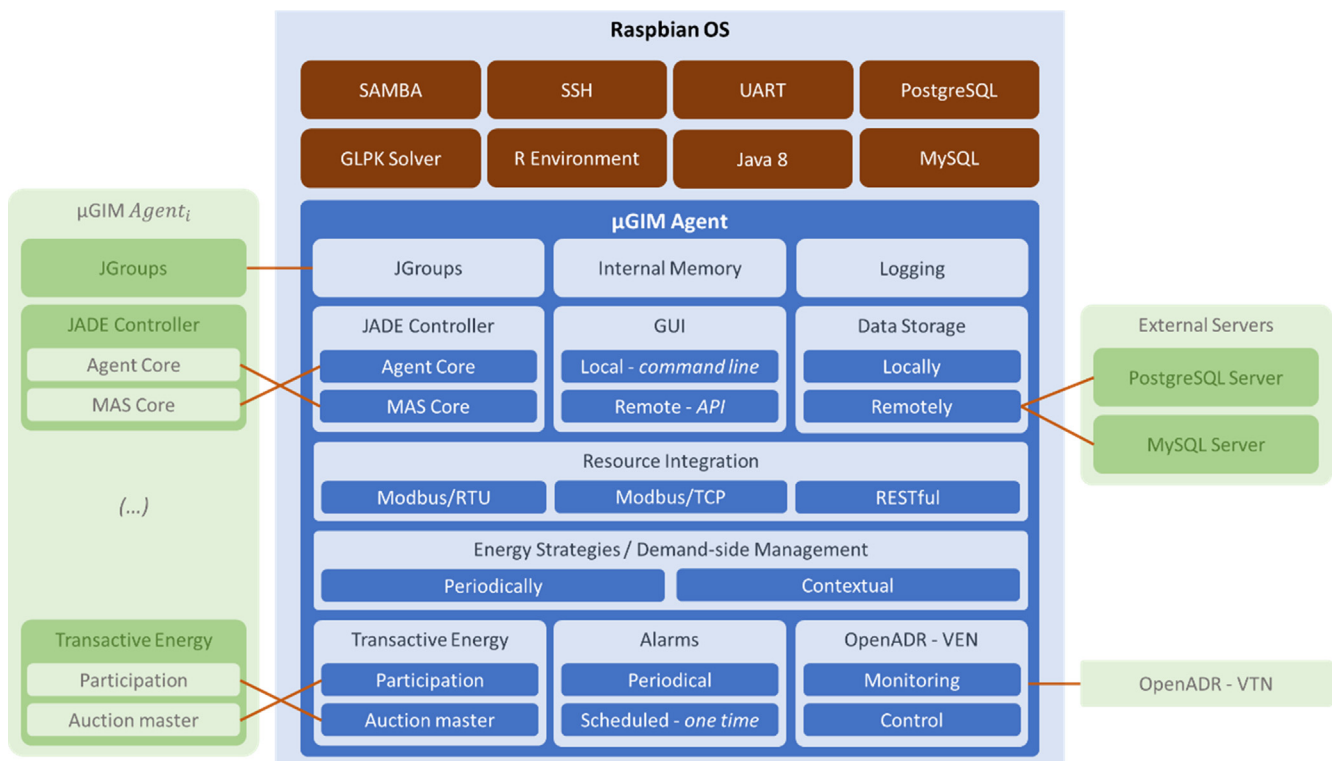


Fig. 2. μ GIM Agent Architecture.

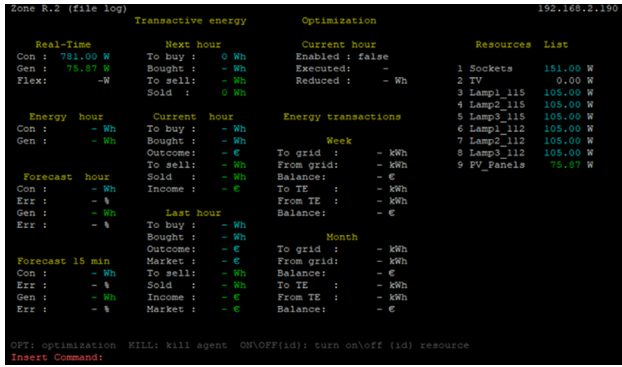


Fig. 3. Agent graphical user interface.

The Resource Integration block is responsible for managing the integration of physical resources, such as electrical loads and sensors. In the μ GIM agents, each physical resource is a thread that monitors and controls the configured resource; the control is not mandatory, and it depends on the resource configuration. Currently, μ GIM provides drivers for Modbus/RTU, Modbus/TCP, and RESTfull. Modbus is highly disseminated in energy equipment, such as energy analyzers and inverters. The possibility of including resources with RESTfull APIs enables the integration of the Internet of Things (IoT) devices.

The Energy Strategies block enables Demand-Side Management (DSM). Algorithms for DSM and energy-related algorithms are represented in this block. Strategies can be periodic or contextual. Each energy-related strategy is a computational instance of the Energy Strategy interface.

In the μ GIM platform, agents can participate in Transactive Energy local auctions, enabling energy transactions between neighbors. Each agent can assume at least one of two modes: participant, and auction master. In a μ GIM microgrid configuration, only one auction master is allowed. Section 3.5 details the agents' participation in energy transactions.

The Alarms block enables internal alarms in the agent, where each alarm executes a specific function, such as the execution of a forecasting algorithm. Alarms can be Periodic or Scheduled, where periodic alarms are executed periodically (e.g. every hour), and scheduled alarms are executed at a certain time (e.g. a particular optimization algorithm can be executed every day at 10 a.m.).

OpenADR 2.0b [46] block implements a Virtual End Node (VEN) enabling Monitoring reports and the participation on demand response programs; using the Control block. This block is currently under development and more tests and validations must be made.

3.1. Multi-Agent approach

The μ GIM platform is designed for microgrid management with individual players representation. Therefore, a centralized solution is not suitable. The MAS approach is no novelty in the application of microgrids or smart grids, as seen in previous works, such as Refs. [48–50]. A vast survey can also be seen in Ref. [51]. Therefore, the use of such approach seemed right to meet the goals of the μ GIM platform.

The multi-agent approach used in the μ GIM platform enables the agent representation of each microgrid player within an SBC. All agents have their internal and individual goals efficiently managing the players'/buildings' energy. They can communicate and interact with the other microgrid players to achieve their goals.

The μ GIM platform is FIPA compliant, in opposition to RIAPS and VOLTTRON solutions that are not FIPA compliant. This compliance is provided by the used JADE framework.

3.2. Single-Board computers

SBC offers new possibilities for software deployment. The small size and computational power available in SBC are advantages that can be used for energy management systems. Besides the already mentioned RIAPS [32] and VOLTTRON [39], other solutions used SBC for energy solutions, such as dRF-Pi solution [52], and publications such as in Refs. [28,53], and [30]. The versatility, low-price, low-size and proven used of SBC for energy solutions were the main motivations that support the choice of developing μ GIM in SBC.

The μ GIM platform was designed and developed for Raspberry Pi 3 Model B. The reason behind this decision regards the support and community size of Raspberry Pi as well as having all the necessary connections in the board (multiple USB, HDMI, and RJ45). However, other SBC were successfully tested and can be used, such as NanoPi M1 Plus and BeagleBone; some of them with great specification sets. For instance, NanoPi NEO Core 2 is a 40x40 mm board with 1 GB DDR3 RAM, 8 GB eMMC storage (which can be extended using an SD card), an Allwinner H5 64-bit 1.5 GHz quad-core processor, 68 General Purpose Input/Output (GPIO) pins and able to run Ubuntu Core operating system. The size, available GPIO, and the computation power enable the deployment of energy management systems in boards as small as 40x40 mm.

Even the card size of Raspberry Pi 3 Model B is small enough to enable its deployment almost anywhere. With 85 × 56 mm size, Raspberry Pi 3 can be deployed inside the electrical enclosure or near the electrical enclosure – minimizing the distance from energy analyzers that can be placed in the electrical enclosure and that can be integrated using Modbus/RTU protocol.

3.3. Resource integration

The resource integration, in the μ GIM agents, enables the monitoring and management of the players' resources. Nowadays, IoT devices must be considered. Their massification makes them common devices in buildings. Also, they can be used in energy management systems, as seen in Refs. [54–56]. Therefore, the integration of IoT devices must be taken into consideration.

In the μ GIM agents, RESTfull resource type can be used to integrate IoT devices in the platform. For IoT devices that cannot be directly integrated, it is possible to install in the SBC one of the home automation systems available, some good choices being the following open-source systems: Home Assistant, Domoticz, and openHAB.

The protocols Modbus/RTU and Modbus/TCP are highly used in energy equipment, such as energy analyzers and photovoltaic inverters. For this reason, they are available in μ GIM agents to enable a direct connection between the agent and energy equipment. Because μ GIM is end-user focused, protocols such as IEC 61850 are not yet available.

3.4. Energy strategies

To provide a suitable solution for end-users' representation capable of managing energy resources, it is needed to integrate energy management solutions, such as demand side management [57], active participation in demand response programs [58], energy forecasting [59], scheduling algorithms [60] and learning algorithms [61]. Energy management related algorithms are seen in μ GIM platform as Energy Strategies.

Energy strategies can assume one of two types: periodic, or contextual. Periodic strategies are strategies that are executed periodically without context relation; for instance, forecasting algorithms for consumption and/or generation. The contextual strategies are executed when a certain context occurs; for instance, energy bal-

ancing algorithms to match consumption and generation can be executed when the consumption and generation do not match.

For periodic strategies, the system uses a variation of Linux's crontab schedule notation (Fig. 4), where strategies can be configured to be executed at a certain time. The use of percental value, in the notation of Fig. 4, indicates multiple values – for instance, the notation `* 8% ***` is executed at 8 a.m., 4 p.m., and 12 p.m. The last value of the notation is an adjustment, in minutes, to be made in the time. For instance, the notation `15% **** -5` is executed every hour at the following minutes: 10, 25, 40, and 55.

Contextual strategies are associated with restrictions (mandatory) and conditions (optional). All the restrictions of a strategy must be fulfilled. If a strategy has conditions, then at least one of them must be fulfilled. The agents' contextual strategies are only executed if all restrictions are fulfilled and at least one condition is fulfilled.

By default, μ GIM agents do not provide energy strategies. Energy strategies were designed and developed as blank/empty containers that can be created by any developer or user. Energy strategies are open to any developer and contribution. Currently, and to provide a range of options, the μ GIM platform allows the use of several technologies that were tested and that can be used in energy strategies, such as GNU Linear Programming Kit (GLPK) solver, Java Optimization Modeler (JOM) library, R project, Deep Learning for Java (DL4J) and TensorFlow. The energy strategy described in Section 4.2, of this paper, was developed using the JOM library and GLPK solver.

3.5. Smart grid participation

In the μ GIM platform, players/agents can actively participate in the microgrid and smart grid. This is possible with the OpenADR and Transactive Energy blocks. The OpenADR block implements the VEN OpenADR 2.0b profile specifications [47]. A Virtual Top Node (VTN) can directly interact with the agent and with the agent's resources, depending on the agent configuration. OpenADR is an important enabler for the end-user active participation in the smart grid [62].

Currently, μ GIM also provides a simple peer-to-peer local auction for energy transactions. By using the agent's configuration file, an agent can be configured to be a participant of transactive energy, participating in the local auction. The μ GIM platform enables the use of one of the four available auction types: English, Dutch, Blind, and Vickrey. An agent uses energy strategies to participate in transactive energy (e.g. sell everything, sell nothing, do aggressive bids, buy at any cost). The peer-to-peer transactions are a μ GIM feature that will not be handled in this work, begin under development and not finished.

Transactive energy participation is configured with periodic strategy notations. Therefore, it is possible to have several auctions running at different times and using different auctions types.

4. Microgrid deployment

This section presents the μ GIM platform deployed in an office building where more than one entity is responsible to manage energy resources. The present section is divided in two: the building deployment where the deployment of the microgrid is detailed, and the Zone R.2 resource management in which a particular energy strategy placed in Zone R.2 agent is presented.

4.1. Building deployment

The μ GIM platform is deployed in an office building that already had a Supervisory Control and Data Acquisition (SCADA) system

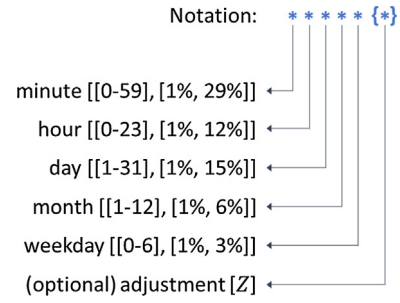


Fig. 4. Periodic strategies notation.

with several energy analyzers, Programmable Logic Controllers (PLCs), dimmable lamps using Digital Addressable Lighting Interface (DALI) protocol and one grid-connected photovoltaic inverter. Fig. 5 shows the SCADA system existent in the building.

The office building has sixteen offices that can be rented. The building has a total of forty photovoltaic panels with and individual peak of 250 W; totalizing a generation peak of 10 kW. The offices can only be rented in pairs or triples, because of electrical restrictions. Each renter has access/control to four photovoltaic panels, making each renter responsible for managing 1 kW of peak generation. The building's manager manages the remaining photovoltaic panels – which are not rented. The building manager is noted as Z.0 and the four renters are noted as L.1, L.2, L.3, and R.2.

Fig. 6 shows a satellite image of the office building with a top layer identifying the inside rooms and zones used. Each zone is managed by a unique μ GIM agent. As can be seen, the two last right offices are not considered in this deployment. The common areas and toilets are managed by the building's manager/agent, as well as the kitchen (room 10) and rooms 11 and 16, which are not rented.

Agents are running in Raspberry Pi boards. Fig. 7 shows the used Raspberry Pi version in each agent and how they are connected. In this microgrid configuration, the building's agent (Z.0) will accommodate the MAS Core and is the Auction master. To test the performance of the μ GIM agents, the Z.0 agent is running in a Raspberry Pi Zero W; a board with a size of 65x30 mm using a Wi-Fi connection.

JGourps toolkit connects all the agents using a tunnel in the computer network. After the first discovery, the Building agent broadcasts the existence of a MAS Core in its IP, sending the MAS name and port. The green circle represents the MAS where all agents are connected. Within the MAS, there is the auction master link between all agents willing to participate in the peer-to-peer energy transactions.

4.2. Zone R.2 resource management energy strategy

In Zone R.2 agent, the energy strategy proposed in Ref. [63] is used in order to test the ability of μ GIM agents to execute energy management algorithms. The currently presented energy balancing algorithm is not this paper focus; it will be presented only to validate the abilities of μ GIM agents.

$$\begin{aligned} \text{minimize} = & \sum_{i=1}^{n_{var}} |P_i^{var}| \times W_i^{var} + \sum_{i=1}^{n_{dis}} |P_i^{dis}| \times W_i^{dis} + w \\ & \times \left[P^{Gen} - P^{Cons} + \sum_{i=1}^{n_{var}} P_i^{var} + \sum_{i=1}^{n_{dis}} P_i^{dis} \right] \end{aligned} \quad (1)$$

The algorithm implemented in the Zone R.2 agent will balance the agent consumption (P^{Cons}) with the on-site generation (P^{Gen}), using the objective function of Equation (1). The algorithm is exe-

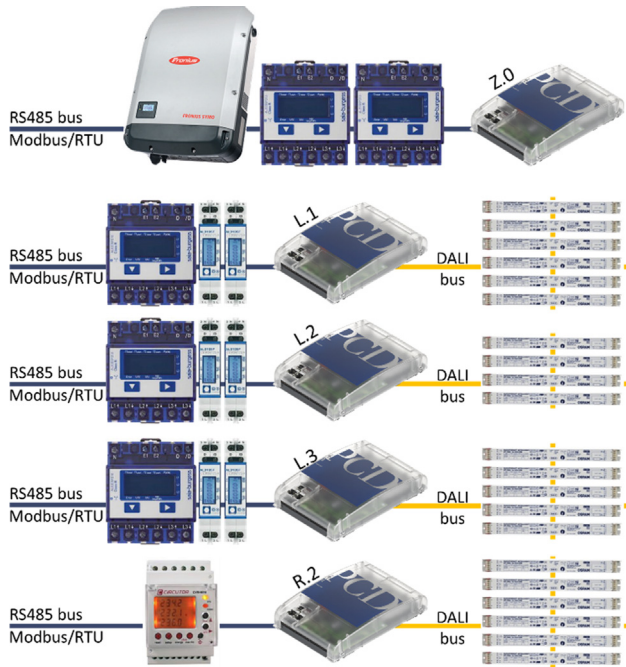


Fig. 5. Hardware deployment for monitoring and control.



Fig. 6. Building zones/agents deployment.

cuted when are imbalanced consumption and generation and every four seconds when the energy is balanced. The four-second execution enables the agent to change its consumption according to the resources' priorities; therefore, a continuous execution is needed even when the energy is balanced. The algorithm considers two resource types, depending on its control: variable, and discrete.

In Equation (1), n_{var} represents the number of loads capable of load reduction and n_{dis} represents the number of loads capable of load curtailment, while resource priorities are represented by W_i^{var} and W_i^{dis} , for each resource type. The resources' priority is the combination of user preferences and sensor data, according to the formulation in Ref. [63]. For instance, the room's current temperature is correlated with the user's desired room temperature to provide the priority for cooling and heating resources. w is a weight that prioritizes the balance between generation and consumption.

$$wp \times P^{Gen} \geq P^{Cons} - \sum_{i=1}^{n_{var}} P_i^{var} - \sum_{i=1}^{n_{dis}} P_i^{dis} \quad (2)$$

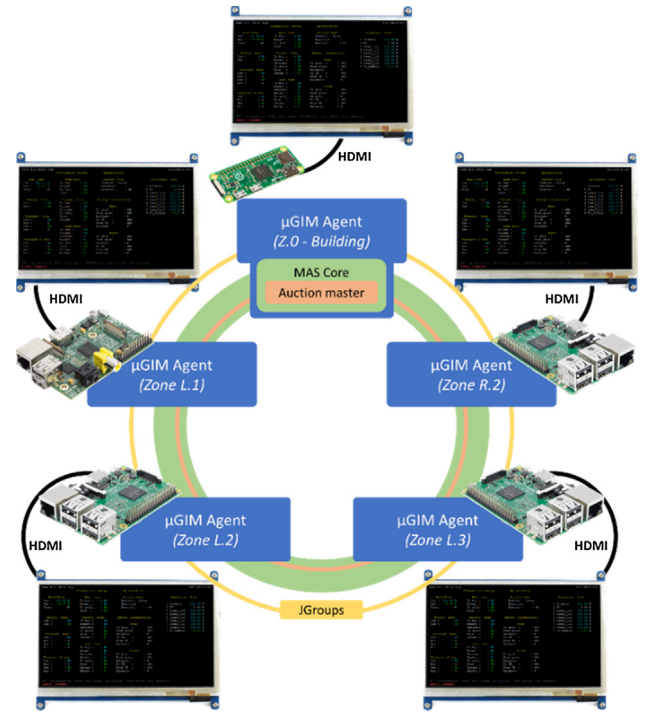


Fig. 7. Agents/zones connectivity and communication.

$$P_i^{dis} = P_i^{dis_cons} \times X_i^{dis}, \forall i \in \{1, \dots, n_{dis}\} \quad (3)$$

Equations (2) and (3) are two constraints. Equation (2) ensures the balance between generation and consumption. The weight variable wp is used in the generation; this will create a range where a small unbalance is accepted. Equation (3) represents the curtailment program for discrete resources. X_i^{dis} indicates if resource i is turned on or off by the optimization. $P_i^{dis_cons}$ represents the power consumption of a resource. $P_i^{dis_cons}$ can be a positive or negative number. If consumption is higher than generation, it will assume a positive value (i.e. the resource consumption), otherwise, it will assume a negative value (i.e. the resource consumption multiplied with -1); this enables the algorithm to decrease and increase the agent consumption according to its real-time generation.

The GLPK solver was used in this energy strategy to provide the best optimization of resources. The GLPK solver is being executed inside the SBC (i.e. Raspberry Pi 3 Model B). The objective function and constraints are modeled using the JOM library in order to simplify the use of the solver, providing mathematical modeling of the presented equations.

This energy strategy is used in Zone R.2 considering 10 controllable resources and 11 sensors. The sensors' data are used to calculate the resources' priority, and the controllable resources are used to adjust the energy consumption. The agent also integrates uncontrollable resources that are not considered in the optimization algorithm but are considered in the energy strategy. The execution triggers and conditions are specified in the configuration file of the agent, as seen in Ref. [63].

5. Results

The results show the efficiency of the μ GIM monitor capabilities and its ability to adequately manage energy resources using sensors for contextual management. The results are obtained from the real-time operation of μ GIM agents.

The shown data are publishing as open access datasets in Refs. [10,11]. The first dataset represents a week with measures every 10 s; starting on Wednesday, 10 April 2019, and ending on Tuesday, 15 April 2019. The charts have in the xx axis the hours of each day, the days can be identified by the generation peaks. The monitoring data are shown in Figs. 8–13. The data showed a significant generation decrease in the weekend (fourth and fifth days in each chart).

The monitoring data were read from the PostgreSQL remote storage server. Meaning that the data were read in the agent, stayed in the local storage database in the agent, and then sent to the remote storage server.

Fig. 8 shows the energy data from the Zone Z.0 agent. This agent is responsible for the building, managing the common areas as well as rooms 10, 11 and 16. Being the building's owner, this agent is also responsible for managing a 6 kW photovoltaic panels array. From consumptions it is possible to observe the presence of persons even during the weekend (fourth and fifth days of the graphs). Only on Sunday the consumption of the common right area is smaller. Because it has a peak of photovoltaic of 6 kW, this agent has more generation than consumption during the middle of the day.

Zone L.1 data are shown in Fig. 9. During the recorded week this agent is not self-sustainable, always demanding energy from the grid. The major consumption of this agent results from the electrical sockets (brown line). By matching the consumptions of the lamps with the weekend consumption of Fig. 8 it is possible to see that the person(s) that used the building during the weekend was working in Zone L.1.

Fig. 10 shows the energy data from the Zone L.2 agent, where a server room is placed. For the agent of Zone L.2, the main consumption comes from the HVAC system that is connected to the server room; ensuring a stable temperature along the days. The presented graphs represent power and not energy. By looking at the power, it is noticed that the agent has small periods of time that it has more generation than consumption, but looking at energy per hour, the agent always demands energy from the grid.

Fig. 11 shows the data of the Zone L.3 agent. This agent has periods, during the middle of the day, where consumption is lower than generation. The consumption is stable, but it is visible the HVAC system left turned on during the first night, from the first day to the second day. A curiosity of this zone is the no use of lamps. The persons in this zone prefer not having artificial light while working.

The most self-sustainable agent is the Zone R.2 agent shown in Fig. 12. With the lowest consumption, this agent has the champion of injecting energy into the grid. The use of the lights during the day reflects the use of the zone during the weekdays.

Zone R.2 has 11 sensors installed, some of these data can be seen in Fig. 13. All sensors are available in the published dataset [10]. In the last two days (Monday and Tuesday) a shot-time peak of light occurs in the morning near 8:15. This light peak is caused by the cleaning services collecting the garbage. During the week, it is also visible a trend related to the increase of the temperature, above normal, when there are persons inside the room.

The data from Fig. 14 are part of the dataset published in Ref. [11]. The dataset has data regarding the Zone R.2 agent during a 24-hour period with the balancing algorithm described in Section 4.2 of this paper. The measurements of the dataset were taken every 10 s, during the 24-hour period. Fig. 14 shows the balancing algorithm working between 16:00 and 18:00 of a weekday. The consumption, represented by the blue line, follows the generation, represented by the green line. The algorithm, executed in GLPK solver using 10 resources and 11 sensors, is able to calculate the resources' priorities according to the user's preferences and optimize the consumption following the generation profile, decreasing

the need for external energy supply. In Fig. 14, the consumption decreases at the same rate as the generation, but near 16:50 the consumption increases because generation also increases, meaning that the algorithm is able to decrease and increase the consumption according to the generation available onsite. During the period shown in Fig. 14, only at 17:11 it was necessary to buy energy from the grid to serve the high consumption peak, resulted from the fan heater; almost immediately turned off by the balancing algorithm.

6. Discussion and conclusion

The deployment and daily use of microgrids are a reality. They bring advantages for smart grids and have positive impacts in isolated areas. However, there is a lack of the offer of complete solutions for microgrid operation. There are in the market solutions for equipment synchronization and inverters that made microgrids possible, but management solutions are not easy to find and usually operate using centralized control. Currently, there is a significant gap between academic concepts and the physical implementations of microgrids; there are a few microgrids that try to minimize this gap, but they are not yet commonly used in practice.

The proposed solution, μ GIM, brings a layered and dynamic architecture that can serve the real implementations while keeping a door open for the integration and adoption of new and disruptive approaches. μ GIM is a totally distributed agent approach solution that looks and treats a microgrid as a community of individual players that can collaborate and compete in the microgrid environment.

Similar solutions such as VOLTTRON and RIAPS are discussed in this paper. These open-source solutions are similar to μ GIM regarding their ability to run in single-board computers, their multi-agent approach and their end-user participation in the microgrid. However, the μ GIM platform is designed having the individual end-user as the central focus, providing better resource integration and individual operation inside a microgrid environment. The μ GIM agents can be deployed as a standalone solution in an isolated building outside a microgrid and the user can take advantage of its management and optimization capabilities, while VOLTTRON and RIAPS were built for multi-agent systems only, not providing standalone solutions for a single end-user. The ability to work as a standalone solution allows the adoption of μ GIM agents in end-users that are not yet integrated in a multi-agent μ GIM microgrid. Otherwise, the dissemination of μ GIM will be limited to the number of players in microgrids using the μ GIM platform.

This paper shows the promising results of two open access datasets that use raw data from the μ GIM platform. The third dataset regarding transactive energy can also be found in Ref. [64]. Detailed data and the capability of providing a balancing algorithm in real-time demonstrate clear evidence of the efficiency and potential of μ GIM for player and microgrid operation. Collaborative and cooperative agent behavior is not discussed in this paper – because the focus is the μ GIM platform architecture and how it can be deployed in real microgrids.

The μ GIM platform is not, at this time, an open-source software because the code is not published for everyone. However, and following the VOLTTRON and RIAPS roadmap, the authors are planning to make μ GIM platform also available for public access using an open source repository with the appropriate documentation for μ GIM deployment. Making μ GIM an open source software can also benefit its development; because more developers can contribute to μ GIM, namely in what regards IoT resource integration, energy strategies, and transactive energy.

The novelty of the proposed μ GIM platform and its capabilities are the main contributions of this paper. The shown results are

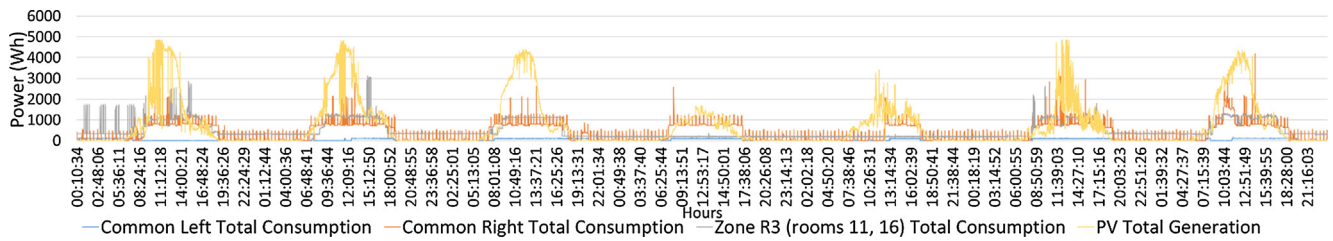


Fig. 8. Energy data read in Zone Z.0 agent.

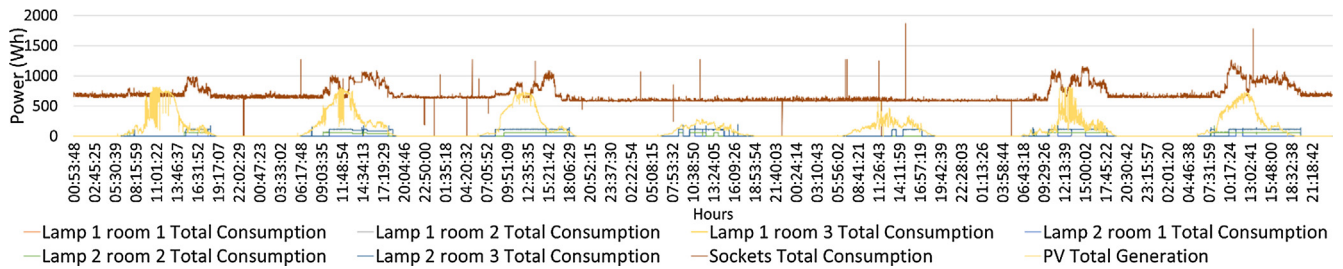


Fig. 9. Energy data read in Zone L.1 agent.

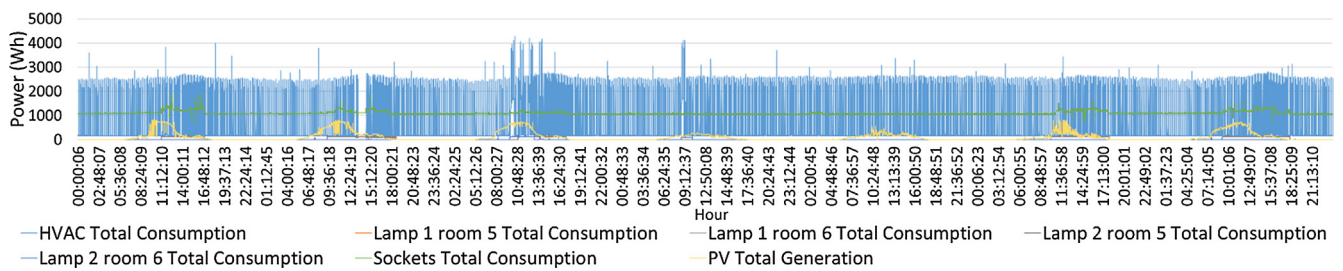


Fig. 10. Energy data read in Zone L.2 agent.

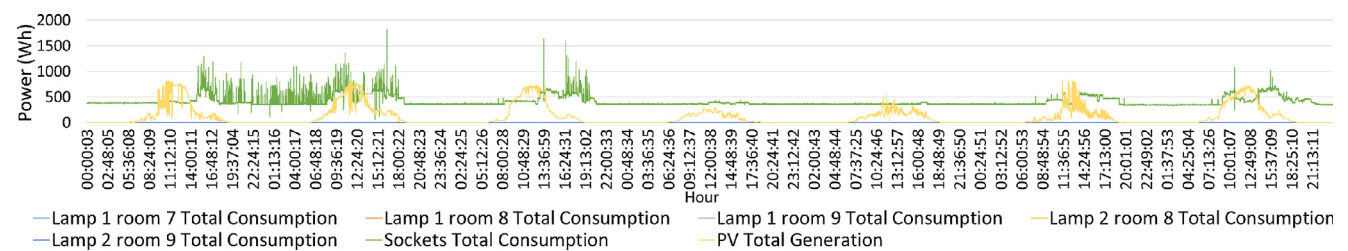


Fig. 11. Energy data read in Zone L.3 agent.

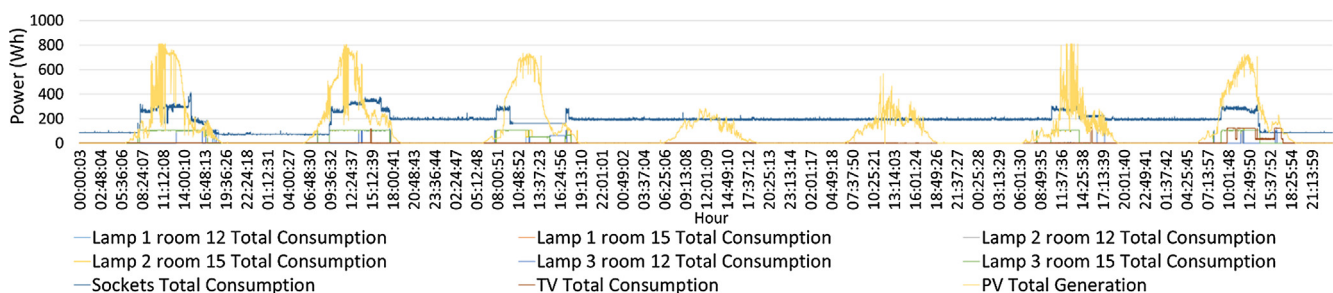


Fig. 12. Energy data read in Zone R.2 agent.

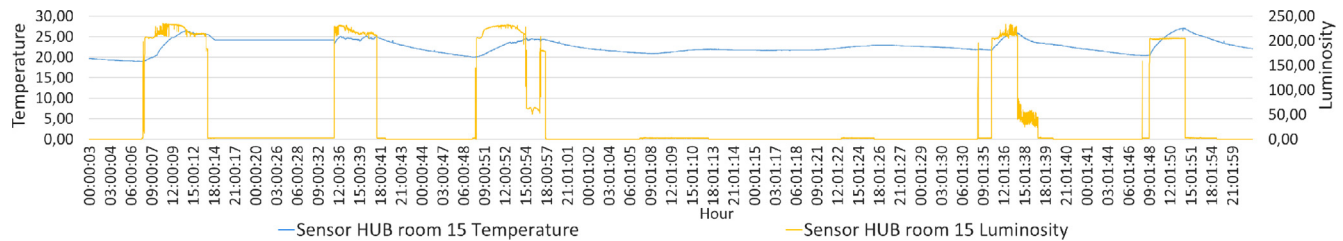


Fig. 13. Sensor data read in Zone R.2 agent.

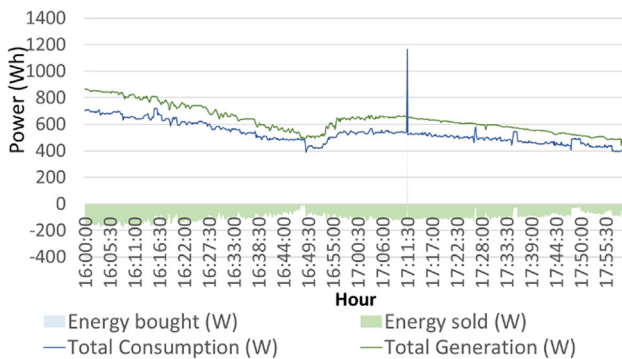


Fig. 14. Balance algorithm in Zone R.2 agent.

highly promising. By using single-board computers, smaller as Raspberry Pi Zero W, the μ GIM agents demonstrate to be very hardware efficient while capable to run 24/7 without significant issues. The small size of the hardware used allows the deployment of μ GIM agents virtually anywhere.

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.

Acknowledgments

This work has been developed under the MAS-SOCIETY project – PTDC/EEI-EEE/28954/2017 and has received funding from FEDER Funds through COMPETE program and from National Funds through FCT under the projects UID/EEA/00760/2019 and SFRH/BD/109248/2015.

References

- [1] L. Martirano, E. Habib, G. Parise, G. Greco, M. Manganelli, F. Massarella, L. Parise, Demand side management in micro-grids for load control in nearly zero energy buildings, *IEEE Transactions on Industry Applications* PP (Feb. 2017), 1–11.
- [2] C. Marnay, S. Chatzivasileiadis, C. Abbey, R. Iravani, G. Joos, P. Lombardi, P. Mancarella, and J. von Appen, "Microgrid Evolution Roadmap," in *Proc. 2015 International Symposium on Smart Electric Distribution Systems and Technologies (EDST)*, pp. 139–144.
- [3] B. Washom, J. Dilliot, D. Weil, J. Kleissl, N. Balac, W. Torre, and C. Richter, "Ivory Tower of Power: Microgrid Implementation at the University of California, San Diego," *IEEE Power and Energy Magazine*, vol. 11, pp. 28–32, July–Aug. 2013.
- [4] LO3ENERGY, "EXERGY – Business Whitepaper", 24 April 2018 (accessed 29/03/2019) <https://lo3energy.com/wp-content/uploads/2018/04/Exergy-BIZWhitepaper-v11.pdf>
- [5] K. Hirose, "Behavior of the Sendai Microgrid during and after the 311 Great East Japan Disaster," in *Proc. 2013 35th International Telecommunications Energy Conference, SMART POWER AND EFFICIENCY*, pp. 1–6.
- [6] Bo Zhao, Jian Chen, Leiqi Zhang, Xuesong Zhang, Ruwen Qin, Xiangning Lin, Three representative island microgrids in the East China Sea: Key technologies and experiences, *Renewable and Sustainable Energy Reviews* 96 (2018) 262–274, <https://doi.org/10.1016/j.rser.2018.07.051>.
- [7] Steven J. Johnston, Philip J. Basford, Colin S. Perkins, Herry Herry, Fung Po Tso, Dimitrios Pezaros, Robert D. Mullins, Eiko Yoneki, Simon J. Cox, Jeremy Singer, Commodity single board computer clusters and their applications, *Future Generation Computer Systems* 89 (2018.) 201–212, <https://doi.org/10.1016/j.future.2018.06.048>.
- [8] Vladimir Vujović, Mirjana Maksimović, Raspberry Pi as a Sensor Web node for home automation, *Computers & Electrical Engineering* 44 (2015) 153–171, <https://doi.org/10.1016/j.compeleceng.2015.01.019>.
- [9] M.H. Cintuglu, T. Youssef, O.A. Mohammed, Development and application of a real-time testbed for multiagent system interoperability: A case study on hierarchical microgrid control, *IEEE Transactions on Smart Grid* 9 (3) (May 2018) 1759–1768, <https://doi.org/10.1109/TSG.2016.2599265>.
- [10] L. Gomes, "uGIM: week monitorization data of a microgrid with five agents (10/04/19–16/04/19)", (Version 1.0), Zenodo (2019).
- [11] L. Gomes, Dataset of uGIM deployed in an office building, Zenodo (2019), <https://doi.org/10.5281/zenodo.2565289>.
- [12] F.H. Malik, M. Lehtonen, A review: Agents in smart grids, *Electric Power Systems Research* 131 (2016) 71–79, <https://doi.org/10.1016/j.epsr.2015.10.004>.
- [13] Pedro Oliveira, Zita Vale, Hugo Morais, Tiago Pinto, Isabel Praça, A multi-agent based approach for intelligent smart grid management, *IFAC Proceedings Volumes* 45 (21) (2012) 109–114, <https://doi.org/10.3182/20120902-4-FR-2032.00021>.
- [14] L. Gomes, P. Faria, H. Morais, Z. Vale and C. Ramos, "Distributed, Agent-Based Intelligent System for Demand Response Program Simulation in Smart Grids," in *IEEE Intelligent Systems*, vol. 29, no. 1, pp. 56–65, Jan.–Feb. 2014. doi: 10.1109/MIS.2013.2
- [15] Zita Vale, Tiago Pinto, Isabel Praça, Hugo Morais, MASCEM: Electricity markets simulation with strategic agents, *IEEE Intelligent Systems* 26 (2) (2011) 9–17, <https://doi.org/10.1109/MIS.2011.3>.
- [16] P. Oliveira, T. Pinto, H. Morais and Z. Vale, MASGrid – A Multi-Agent Smart Grid Simulation Platform, 2012 IEEE Power and Energy Society General Meeting, San Diego, CA, 2012, pp. 1–8. doi: 10.1109/PESGM.2012.6345649
- [17] L. Gomes, P. Faria, Z.A. Vale, J. Silva, Energy analyzer emulator for microgrid implementation and demonstration and respective gateway, *IEEE Transactions on Industry Applications* 55 (1) (2019) 134–144, <https://doi.org/10.1109/TIA.2018.2866451>.
- [18] T. Logenthiran, D. Srinivasan, A.M. Khambadkone, H.N. Aung, Multiagent system for real-time operation of a microgrid in real-time digital simulator, *IEEE Transactions on Smart Grid* 3 (2) (June 2012) 925–933, <https://doi.org/10.1109/TSG.2012.2189028>.
- [19] N. Yorino, Y. Zoka, M. Watanabe, T. Kurushima, An optimal autonomous decentralized control method for voltage control devices by using a multi-agent system, *IEEE Transactions on Power Systems* 30 (5) (Sept. 2015) 2225–2233, <https://doi.org/10.1109/TPWRS.2014.2364193>.
- [20] W. Liu, W. Gu, W. Sheng, X. Meng, Z. Wu, W. Chen, Decentralized multi-agent system-based cooperative frequency control for autonomous microgrids with communication constraints, *IEEE Transactions on Sustainable Energy* 5 (2) (April 2014) 446–456, <https://doi.org/10.1109/TSTE.2013.2293148>.
- [21] A. Anvari-Moghaddam, A. Rahimi-Kian, M.S. Mirian, J.M. Guerrero, A multi-agent based energy management solution for integrated buildings and microgrid system, *Applied Energy* 203 (2017) 41–56, <https://doi.org/10.1016/j.apenergy.2017.06.007>.
- [22] A. Anvari-Moghaddam, J. M. Guerrero, A. Rahimi-Kian and M. S. Mirian, "Optimal real-time dispatch for integrated energy systems: An ontology-based multi-agent approach," 2016 IEEE 7th International Symposium on Power Electronics for Distributed Generation Systems (PEDG), Vancouver, BC, 2016, pp. 1–7. doi: 10.1109/PEDG.2016.7526997
- [23] R. de Azevedo, M.H. Cintuglu, T. Ma, O.A. Mohammed, Multiagent-based optimal microgrid control using fully distributed diffusion strategy, *IEEE Transactions on Smart Grid* 8 (4) (July 2017) 1997–2008, <https://doi.org/10.1109/TSG.2016.2587741>.
- [24] M. Mao, P. Jin, N.D. Hatziaargyriou, L. Chang, Multiagent-based hybrid energy management system for microgrids, *IEEE Transactions on Sustainable Energy* 5 (3) (July 2014) 938–946, <https://doi.org/10.1109/TSTE.2014.2313882>.
- [25] C. Dou, D. Yue, X. Li, Y. Xue, MAS-based management and control strategies for integrated hybrid energy system, *IEEE Transactions on Industrial Informatics* 12 (4) (Aug. 2016) 1332–1349, <https://doi.org/10.1109/TII.2016.2569506>.
- [26] M. Yazdani, A. Mehrizi-Sani, Distributed control techniques in microgrids, *IEEE Transactions on Smart Grid* 5 (6) (Nov. 2014) 2901–2909, <https://doi.org/10.1109/TSG.2014.2337838>.

- [27] E. Amicarelli, Q. T. Tran and S. Bacha, Multi-agent system for day-ahead energy management of microgrid, 2016 18th European Conference on Power Electronics and Applications (EPE'16 ECCE Europe), Karlsruhe, 2016, pp. 1–10. doi: 10.1109/EPE.2016.7695257
- [28] S. Drakontaidis, M. Stanchi, G. Glazer, J. Hussey, A. S. Leger and S. J. Matthews, Towards Energy-Proportional Anomaly Detection in the Smart Grid, 2018 IEEE High Performance extreme Computing Conference (HPEC), Waltham, MA, 2018, pp. 1–7. doi: 10.1109/HPEC.2018.8547695
- [29] L. Zhang, J. Xiao, P. Wang and X. Pan, Design and implementation of communication network for modular microgrid based power park, 2016 IEEE Innovative Smart Grid Technologies - Asia (ISGT-Asia), Melbourne, VIC, 2016, pp. 160–165. doi: 10.1109/ISGT-Asia.2016.7796379
- [30] J. Xiao, P. Wang, L. Setyawan, Q. Xu, Multi-level energy management system for real-time scheduling of DC microgrids with multiple slack terminals, IEEE Transactions on Energy Conversion 31 (1) (March 2016) 392–400. <https://doi.org/10.1109/TEC.2015.2488639>.
- [31] The Linux Foundation, The Linux Foundation Launches LF ENERGY, New Open Source Coalition, 12 July 2018. Available (accessed 12/04/2019): <https://www.linuxfoundation.org/press-release/2018/07/the-linux-foundation-launches-lf-energy-new-open-source-coalition/>
- [32] S. Eisele, I. Mardari, A. Dubey and G. Karsai, RIAPS: Resilient Information Architecture Platform for Decentralized Smart Systems, 2017 IEEE 20th International Symposium on Real-Time Distributed Computing (ISORC), Toronto, ON, 2017, pp. 125–132. doi: 10.1109/ISORC.2017.22
- [33] W. R. Otte, A. Dubey, S. Pradhan, P. Patil, A. Gokhale, G. Karsai, and J. Willemsen, F6COM: A Component Model for Resource-Constrained and Dynamic Space-Based Computing Environment, in Proceedings of the 16th IEEE International Symposium on Object-oriented Real-time Distributed Computing (ISORC'13), Paderborn, Germany, Jun. 2013.
- [34] A. Dubey, G. Karsai, N. Mahadevan, A component model for hard real-time systems: CCM with ARINC-653, Software: Practice and Experience 41 (12) (2011) 1517–1550.
- [35] Y. Du, H. Tu, S. Lukic, A. Dubey and G. Karsai, "Distributed Microgrid Synchronization Strategy Using a Novel Information Architecture Platform," 2018 IEEE Energy Conversion Congress and Exposition (ECCE), Portland, OR, 2018, pp. 2060–2066. doi: 10.1109/ECCE.2018.8557695
- [36] H. Tu et al., An Adaptive Interleaving Algorithm for Multi-Converter Systems, 2018 9th IEEE International Symposium on Power Electronics for Distributed Generation Systems (PEDG), Charlotte, NC, 2018, pp. 1–7. doi: 10.1109/PEDG.2018.8447801
- [37] S. Eisele, A. Dubey, G. Karsai and S. Lukic, "WiP Abstract: Transactive Energy Demo with RIAPS Platform," 2017 ACM/IEEE 8th International Conference on Cyber-Physical Systems (ICCPs), Pittsburgh, PA, 2017, pp. 91–92.
- [38] B. Akyol, C.H. Allwardt, Z.W. Beech, J.B. Chapman, J.N. Haack, S. Katipamula, R. G. Lutes, K.E. Monson, "VOLTRONtm 2016", June 2016, PNNL-25499 (accessed 23/03/2019) https://voltron.org/sites/default/files/publications/PNNL-25499_VOLTRON_2016.pdf
- [39] VOLTRON team, "Agents in the Platform", (accessed 12/04/2019) <https://voltron.readthedocs.io/en/develop/overview/agents-overview.html>
- [40] M. Karami, G.V. Mc Morrow, L. Wang, Continuous monitoring of indoor environmental quality using an Arduino-based data acquisition system, Journal of Building Engineering 19 (2018) 412–419. <https://doi.org/10.1016/j.jobe.2018.05.014>.
- [41] S. Katipamula, J. Haack, G. Hernandez, B. Akyol, J. Hagerman, VOLTRON: An open-source software platform of the future, IEEE Electrification Magazine 4 (4) (Dec. 2016) 15–22. <https://doi.org/10.1109/MELE.2016.2614178>.
- [42] Sen Huang, Weimin Wang, Michael R. Brambley, Siddharth Goyal, Wangda Zuo, An agent-based hardware-in-the-loop simulation framework for building controls, Energy and Buildings 181 (2018) 26–37. <https://doi.org/10.1016/j.enbuild.2018.09.038>.
- [43] Sen Huang, Jianming Lian, He Hao, Srinivas Katipamula, Transactive control design for commercial buildings to provide demand response, IFAC-PapersOnLine 51 (34) (2019) 151–156. <https://doi.org/10.1016/j.ifacol.2019.01.058>.
- [44] H. Pourbabak, J. Luo, T. Chen, W. Su, A novel consensus-based distributed algorithm for economic dispatch based on local estimation of power mismatch, IEEE Transactions on Smart Grid 9 (6) (Nov. 2018) 5930–5942. <https://doi.org/10.1109/TSG.2017.2699084>.
- [45] D. Raker, A. Sellers, R. Kini, M. Green, T. Stuart, R. Ellingson, R. Khanna, M. Heben, Grid Integration of Building Systems and 1 MW Photovoltaic Array using VOLTRON, 2017 IEEE 44th Photovoltaic Specialist Conference (PVSC), Washington, DC, 2017, pp. 2926–2930. doi: 10.1109/PVSC.2017.8520928
- [46] D. Raker, R. Kini, R. Huntsman, M. Green, O. Spaci, T. Stuart, R. Ellingson, R. Khanna, M. Heben, Transactive Mitigation Of Variability In The Output Of 1 MW Photovoltaic Array Using VolttronTM, 2018 IEEE 7th World Conference on Photovoltaic Energy Conversion (WCPEC) (A Joint Conference of 45th IEEE PVSC, 28th PVSEC & 34th EU PVSEC), Waikoloa Village, HI, 2018, pp. 1462–1467. doi: 10.1109/PVSC.2018.8548242
- [47] U. Herberg, D. Mashima, J. G. Jetcheva and S. Mirzazad-Barijough, OpenADR 2.0 deployment architectures: Options and implications, 2014 IEEE International Conference on Smart Grid Communications (SmartGridComm), Venice, 2014, pp. 782–787. doi: 10.1109/SmartGridComm.2014.7007743
- [48] Y. Han, K. Zhang, H. Li, E.A.A. Coelho, J.M. Guerrero, MAS-based distributed coordinated control and optimization in microgrid and microgrid clusters: A comprehensive overview, IEEE Transactions on Power Electronics 33 (8) (Aug. 2018) 6488–6508. <https://doi.org/10.1109/TPEL.2017.2761438>.
- [49] Z. Li, C. Zang, P. Zeng, H. Yu and H. Li, MAS based distributed automatic generation control for cyber-physical microgrid system, in IEEE/CAA Journal of Automatica Sinica, vol. 3, no. 1, pp. 78–89, 10 January 2016. doi: 10.1109/JAS.2016.7373765
- [50] F. Chen, M. Chen, Q. Li, K. Meng, J.M. Guerrero, D. Abbott, Multiagent-based reactive power sharing and control model for islanded microgrids, IEEE Transactions on Sustainable Energy 7 (3) (July 2016) 1232–1244. <https://doi.org/10.1109/TSTE.2016.2539213>.
- [51] F.H. Malik, M. Lehtonen, A review: Agents in smart grids, Electric Power Systems Research 131 (2016) 71–79. <https://doi.org/10.1016/j.epsr.2015.10.004>.
- [52] dEF-Pi team, "dEF-Pi: Distributed Energy Flexibility Platform & Interface", [accessed on 13/06/2019] <https://fan-ci.sensorlab.tno.nl/builds/defpi-documentation/master/html/>.
- [53] S. J. Matthews and A. S. Leger, Leveraging single board computers for anomaly detection in the smart grid, 2017 IEEE 8th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON), New York, NY, 2017, pp. 437–443. doi: 10.1109/UEMCON.2017.8249031
- [54] L. Gomes, P. Faria, F. Silva and Z. Vale, Energy Resources Management Enabled by Internet of Things Devices, 2019 IEEE-APS Topical Conference on Antennas and Propagation in Wireless Communications (APWC), Granada, Spain, 2019, pp. 258–263. doi: 10.1109/APWC.2019.8870442
- [55] M. Bisadi, A. Akrami, S. Teimourzadeh, F. Aminifar, M. Kargahi, M. Shahidehpour, IoT-enabled humans in the loop for energy management systems: Promoting building occupants' participation in optimizing energy consumption, IEEE Electrification Magazine 6 (2) (June 2018) 64–72. <https://doi.org/10.1109/MELE.2018.2816844>.
- [56] F.G. Brundu et al., IoT software infrastructure for energy management and simulation in smart cities, IEEE Transactions on Industrial Informatics 13 (2) (April 2017) 832–840. <https://doi.org/10.1109/TII.2016.2627479>.
- [57] A. Mahmood, A. Ahmad, H.T. Javed, Z. Mehmood, Z.A. Khan, U. Qasim, N. Javaid, A Survey of 'User Comfort' in Home Energy Management Systems in Smart Grid, 2015 IEEE 29th International Conference on Advanced Information Networking and Applications Workshops, Gwangju, 2015, pp. 36–43. doi: 10.1109/WAINA.2015.124
- [58] M. H. Riaz, M. Zeeshan, Hasan-ul-Banna, T. Kamal and S. A. H. Shah, Demand side management using different energy conservation techniques, 2017 International Multi-topic Conference (INMIC), Lahore, 2017, pp. 1–4. doi: 10.1109/INMIC.2017.8289460
- [59] E. Vinagre, L. Gomes, Z. Vale, "Electrical Energy Consumption Forecast Using External Facility Data", in: 2015 IEEE Symposium Series on Computational Intelligence, Cape Town, 2015, pp. 659–664. <http://doi.org/10.1109/SSCI.2015.101>.
- [60] M.A.A. Pedrasa, T.D. Spooner, I.F. MacGill, Coordinated scheduling of residential distributed energy resources to optimize smart home energy services, IEEE Transactions on Smart Grid 1 (2) (Sept. 2010) 134–143. <https://doi.org/10.1109/TSG.2010.2053053>.
- [61] Jui-Sheng Chou, Duc-Son Tran, Forecasting energy consumption time series using machine learning techniques based on usage patterns of residential householders, Energy 165 (2018) 709–726. <https://doi.org/10.1016/j.energy.2018.09.144>.
- [62] M. Wei, S.H. Hong, M. Alam, An IoT-based energy-management platform for industrial facilities, Applied Energy 164 (2016) 607–619. <https://doi.org/10.1016/j.apenergy.2015.11.107>.
- [63] L. Gomes, J. Spinola, Z. Vale, J.M. Corchado, Agent-based architecture for demand side management using real-time resources' priorities and a deterministic optimization algorithm, Journal of Cleaner Production 241 (2019). <https://doi.org/10.1016/j.jclepro.2019.118154>.
- [64] L. Gomes, "uGIM: a week with peer-to-peer transactions (03/06/2019 - 09/06/2019)", (Version 0.1.0) [Data set]. Zenodo, 2019. Doi: 10.5281/zenodo.3371222.