Hybrid Context-awareness Modelling and Reasoning Approach for Microgrid's Intelligent Control

Soumoud Fkaier[1,2,3], Mohamed Khalgui[2,3] and Georg Frey[1]
[1]*Chair of Automation and Energy Systems, Saarland University, Saarbruecken, Germany*
[2]*Tunisia Polytechnic School, Carthage University, Tunis, Tunisia*
[3]*INSAT LISI Lab, Carthage University, Tunis, Tunisia*

Keywords: Microgrid, Context-awareness, Context Modelling, Context Reasoning.

Abstract: Modern microgrids are promoting the integration of the information and communication technologies (ICT) in order to enhance the emerging advanced power management functionalities such as the integration of renewable energy sources, distributed storage optimization, demand-response strategies, electric vehicles charging, power generation rate forecasting and scheduling, etc. For this, sophisticated sensing and smart metering infrastructures are incorporated in the used equipment as well as in the involved subsystems. Hence many contextual data are become more and more available and its taking into consideration in the control tasks are likely to provide promising results. However, making the microgrid control system understand the data and take the proper decisions based on the identified context is not an easy task to perform. In fact, recognizing the relations and meanings of the sensed data is difficult and complex due to the heterogeneity and intricacy of the involved parts. Hence, providing context-aware modelling and reasoning mechanisms for microgrids becomes necessary. In this context, this paper contributes with two main solutions. First, a microgrid's formalized design providing an easy and understandable view of the system is provided. This definition respects the separation of concerns principle in order to tame the complexity of the complicated system. Second, an ontology-based context modelling and a rule-based context reasoning in the framework of microgrids are provided. To show the suitability of the proposed processes, a formal case study is carried out. The proposed processes are proved to be less resources consuming compared to some of the existing works.

1 INTRODUCTION

A microgrid is a smaller scale version of the traditional electricity grid that can operate independently or in a connected mode with the main power grid. In order to meet the requirements of its users, microgrids fulfil a set of advanced control and management tasks such as the integration of renewable energy sources, distributed storage optimization, demand-response strategies, electric vehicles charging, power generation forecasting and scheduling, etc. Such functionalities are become possible thanks to the integration of ICTs that allow establishing a bidirectional information flow. In fact, the great development of advanced sensing and electric metering devices has paved the way for applying pervasive computing also in the electricity management systems (Fkaier et al., 2016b). Hence, big data flows are available at any time to the control agents of microgrids (Suslov et al., 2016).

Although maximal information and details about the system state are useful, their management and understanding are not easy to perform by microgrid's controller. In fact, controllers have to be aware about the context of every collected data in order to use it properly and exploit it in a fruitful way. Identifying the proper context in a polyvalent system such as a microgrid is really challenging because of the interrelated multidisciplinary "subsystems" constituting the microgrid: microgrids are in general composed of generation subsystem (such as photovoltaic panels or diesel generators), storage subsystem (such as electrical batteries), and consumption subsystem (such as home area buildings). Deducing a conclusion about the state of the microgrid at a given moment based on large sets of sensed data from the mentioned subsystems is difficult since every subsystem can have many states and each state can be inner to the subsystem itself or influenced by other subsystems.

In the context-awareness computing scope, many

Fkaier, S., Khalgui, M. and Frey, G.
Hybrid Context-awareness Modelling and Reasoning Approach for Microgrid's Intelligent Control.
DOI: 10.5220/0009780901160127
In *Proceedings of the 15th International Conference on Software Technologies (ICSOFT 2020)*, pages 116-127
ISBN: 978-989-758-443-5

methods and tools have been proposed in previous works in order to provide solutions to extract accurate contexts. It is acknowledged that the most efficient ones are those that build upon the ontology of well-specific fields. Despite its robustness in the modelling of a system facets and actors, ontology based methods are still less performant in terms of the consumed computational power. This feature is important especially when it comes to control systems that may require real-time functionalities such as microgrids.

In order to overcome the difficulties mentioned above, the current paper proposes an efficient context-aware modelling and reasoning process for microgrids control. First, a formalization of the definition of a microgrid is provided. The definition allows to have a clear view on the subsystems composing microgrids and the relations between them. It is based on the separation of concerns principle that allows to reduce the complexity through separating the different topics (generation, consumption, storage, and prosuming). This definition paves us the way for defining a context modelling of the microgrid system. We introduce a context model based on ontology. It is performed using the Web Ontology Language (OWL) (Zamazal, 2020) and it provides semantic microgrid context representation. Then, a rule-based reasoning mechanism is proposed to be the mechanism for context extraction. We performed a formal case study to validate the suitability of the proposed approach.

The outline of this paper is organized as follows. Section 2 provides an overview of the related works. Section 3 introduces the proposed microgrid definition. Section 4 presents the defined ontology-based microgrids context modelling and rule-based context reasoning. Section 5 provides the case study and the performance evaluation. Finally, Section 6 concludes the paper and presents future perspectives.

2 RELATED WORKS

In this section, an overview of the works in the field of context-awareness and microgrids is provided.

2.1 Context Awareness, Context Modelling and Reasoning

Context-aware computing is the type of computing that takes into account the user activity and/or the changes in its surrounding environment (Khabou et al., 2019). It was first limited to the mobile applications that try to predict the location of the device (Fkaier et al., 2017). Then it was extended to cover many fields ranging from health care systems to smart cities (Fkaier et al., 2016a). In general, the development of context-aware applications has many steps and rely on different modules according to the studied case; but the main steps are resumed in the so-called "context life cycle" which is composed of four phases: (i) context acquisition, (ii) context modelling, (iii) context reasoning, and (iv) context dissemination. The first and last phases are considered conventional phases since most of developers agree on the used techniques and tools. However, the second and third phases are still under research because there are open questions about the improvements of the possible processes, mechanisms, and methods. Performance and suitability of one method over the other are still the subject of discussions and are likely to be them selves context-dependent. Every method, as it is the case of every approach, has its advantages and drawbacks.

The modelling phase is considered as the context representation phase which helps to understand the context details, properties, and relationships. Many solutions exist such as the key-value, markup, graphical, object-oriented, multidisciplinary, domain-focused, spatial, and ontology-based context modelling. The selection of one technique is generally based on the focus of the modelling, for example, whether the relationships and dependencies are to be emphasized or other feature such as the mobility, etc.

The reasoning phase is considered as the conclusion/knowledge extraction based on the modelled contexts (Cheng et al., 2018). In this phase also many techniques can be used such as the probabilistic logic-based, Bayesian networks, ontology-based, case-based, rule-based, supervised as well as unsupervised learning, and fuzzy logic based (Al-Barazanchi and Vural, 2015).

2.2 Microgrids, Smart Grids, and Context Awareness

Context-awareness is acknowledged to be one important mainstay paradigm for the development of smart software applications (this is applicable also to the case of smart grids and smart microgrids). Context-awareness was involved in various microgrids functionalities. In the following, a short overview of the different use cases of context-awareness in the field of smart and micro grids is presented.

(Donohoe et al., 2015) present a survey of the requirements and challenges of context-aware smart grids. (Radzi et al., 2019) introduce context-awareness traffic scheduling algorithm for smart grid's network. (Obaidat et al., 2018) use the context-awareness concepts to create a framework for the intelligent data management in smart grids. (Gomes et al., 2017) as well as (Stavropoulos et al., 2014) use context-awareness for the development of energy management systems of households. (Degha et al., 2019) propose a solution for efficient energy management based on ontological context to smart building. A smart energy equipment management solution for smart cities based on distributed context-awareness is proposed by (Choi et al., 2018). (Donohoe et al., 2013) analyze the adoption of context-awareness paradigm for microgrids storage using electric cars. (Zhuikov and Kyselova, 2013) introduce a context-aware control process for microgirds. (Kyselova et al., 2016) propose a context-aware framework for energy management systems. (Reddy and Krishna, 2018) use context-awareness in the load balancing of power in smart grids. (Meng and Lu, 2015) introduce a context-aware service customization strategy for smart homes.

Despite their importance, none of the above-cited researches have defined a holistic context model that could be used within the microgrids control strategies. There is a need to an understandable model that covers all the factors and perspectives that might influence the context. In addition, efficient reasoning approach should be provided.

2.3 Summary

As it can be seen from the previous subsections, context-awareness is a huge computation filed that has a large variety of techniques and tools and can have multiple use cases in the one field (i.e., smart electricity management). A study of the manifold context modelling techniques leads to conclude that the ontology-based modelling is a very powerful method. In fact, in comparison with other models the ontology-based modelling technique is better in terms of reusability, formality, and interoperabiliy (Li et al., 2015). It also provides better expressive context representations and can be independent from applications (Alegre et al., 2016). Thus, the current paper adopts the ontology paradigm to represents the context modelling thanks to the mentioned advantages.

However, concerning the reasoning some researchers argue that ontology reasoning might be suitable for some kinds of applications and not for some others. (Li et al., 2015) and (Sezer et al., 2017) argue that the ontology-based reasoning can be complex and ambiguous. More importantly, ontology-based reasoning cannot be suitable for time-critical applications (Wang et al., 2004). However, rule-based reasoning technique can be combined with ontology-based models and provide promising efficiency. Machine/deep learning is an important technique that can be used for reasoning and it is suitable for many use cases. However, for microgirds it might have some limitations. First, in order to accomplish its task as expected, machine learning requires large amounts of historic data to train. In this case, providing **good and enough** training data for each particular microgrid is required (given that not all microgrids have the same composition of subsystems as described in Section 3, i.e., their behaviour is not the same). Therefore, from one hand it is required to provide the appropriate training data of the complicated parts, and from the other hand this process should be re-done for each different microgrid and for each time there is a change in a microgrid structure. However, it is easier to define rules, which makes the implementation quicker.

For control systems, such as microgirds, it is useful even obligatory to provide understandable explanation of the interpretations made by the reasoning unit, especially when it comes to security and financial issues. Machine learning builds upon the "black box" models which are not enough clear to be understandable by humans. Therefore, it is wiser to rely on a clear controllable approach. In conclusion, given that microgrids are a kind of control systems, rule-based approach represents a better solution.

In summary, given the above presented analysis of the state of the art, the ontology-based modelling technique along with the rule-based reasoning technique are adopted in the contribution of this paper.

3 MICROGRID DEFINITION

In order to clearly introduce the microgrid context ontology, it is necessary first to define the conceptual model of a microgrid. A microgrid is defined as a small scale electricity grid system that is composed of a set of subsystems. These subsystems influence each-other and their operations are performed under the control and supervision of a control unit called the microgrid controller as depicted in Fig. 1. An information and communication infrastructure (composed mainly of sensors, measurement units, and smart metering devices) enables the exchange of bidirectional information flows between the involved actors.

Four main subsystems can be defined to describe the involved disciplines in the microgrid system: (i) generation, (ii) consumption, (iii) storage, and (iv) prosuming. Each subsystem is characterized by specific properties, has well-defined equipment as well as tasks, and has optimization challenges.

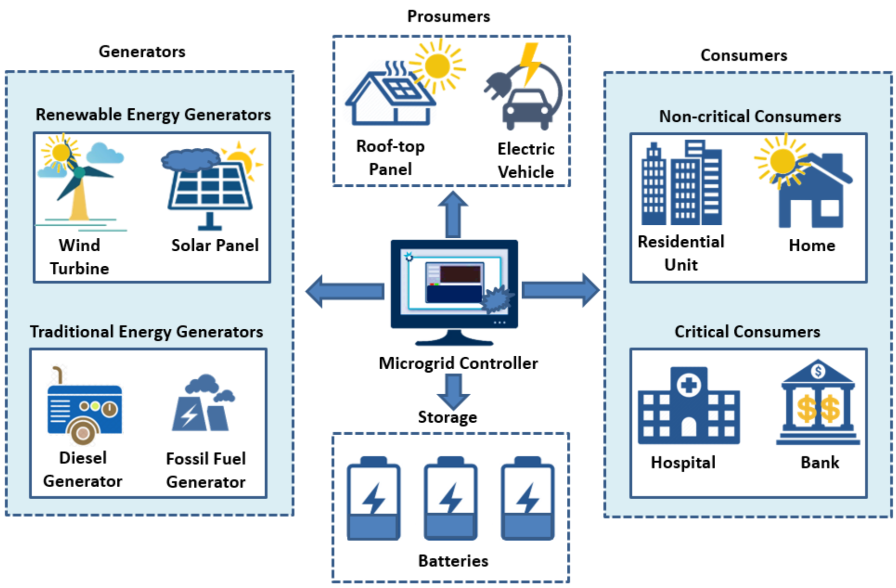

Figure 1: Subsystems composing a micorgrid.

3.1 Electricity Generation Subsystem

Traditionally, the electricity is generated in big facilities in high voltage power then transmitted as medium voltage power, and distributed as low voltage power. However, with the advent of smart grids the generation process has been changed where especially clean energy sources are integrated. To reach intelligent control of the electricity usage, detailed information about the generation must be provided to the controller so that it can make right decisions like the switching between the on/off-grid modes. The awareness about the available generators and their minimal/maximal production abilities are important for the awareness consolidation.

To make the task of perception of electricity generation information easier, two main generation types are defined: renewable and non-renewable. **Renewable energy generation** relies on the natural renewable energy sources such as the wind, the water, and the sun. Properties of the available equipment like the solar photovoltaic panels or wind turbines are tremendous in the context information creation. Capacity, size, number, and utilization rate can all impact the context. More importantly, it is true that renewable energies have many advantages like reducing the emission of greenhouse gas but they have non-trustful behaviour. Despite the weather forecasting and prediction mechanisms, the generation amounts remain unknown or at least non-precise. Therefore, clear view about the considered renewable energies may help in the control task. **Non-renewable energy generation** relies on some materials such as the oil or coal to produce electricity. Despite the bad impact on the globe safety, non-renewable energies guarantee a trustful generation behaviour. Therefore they are in general used as the backbone of the power grid. The percentage of reliance on these energies may help in many tasks such as the tariff definition.

3.2 Electricity Consumption Subsystem

Electricity is nowadays required in nearly all the fields of the people's daily life. Information about the consumption profiles of the consumers supplied within one microgrid is required to determine the control contexts. Also analysing the human activities like the working hours, the holidays, the social events, etc., helps in the context recognition. Similarly, the existence of industrial establishments, big shopping centers, or even product stores among the consumers is necessary to take into account in the control strategy.

It is required that a microgrid provides supply for its consumers, however in some cases (peak hours consumption) shortages can occur, consequently not all consumers can be satisfied. In that case, a microgrid controller need to understand the context and must serve consumers according to their emergency level and priorities. To make the context perception easier for the controller, consumers are classified into two groups: critical and non-critical consumers. **Critical consumers** are consumers whom the shortages or faults in the electricity supply to their activities imply catastrophic losses in terms of people lives, money, or security issues (for example hospitals, banks, and police offices). Electricity should be continuously supplied to such kind of consumers. Thus, considerable awareness should be attached to this information. **Non-critical consumers** are consumers whom the supply can be deffered to later time, and short blackouts do not induce losses rather they induce a deterioration of the quality-of-service.

3.3 Prosumers Subsystem

Equipments and tools of managing the renewable energy are witnessing continuous evolution which facilitate their adoption and integration in many emerging use cases such as the electricity generation from the roof-top photovoltaic panels. Hence, the house lord can play the role of electricity producer and consumer at the same time (from where the term “prosumer” is born). Another use case, is that the produced electricity can be supplied to electric cars as a type of storage. Hence, enabling such advanced functionalities to the users can be the source of sudden big energy demand (in case of simultaneous bad weather and high use) or also the release (during good weather and low use).

The behaviour of prosumers is generally unpredictable, at least if we assume that the metering and sensing infrastructure ensures security and privacy of the user’s data which makes difficult forecasting its activity. That is why being aware of such kind of users allows to take into account any potential sharp demand-response changes.

3.4 Storage Subsystem

Storage systems play a pivotal role in almost all functionalities of the microgrid control. It allows to store electricity and provide it back during the peak shaving process (Dongol et al., 2018), dynamic demand-response mechanisms, load shifting, etc., in order to guarantee better supply continuity and better disturbances avoidance. It is necessary to a microgrid controller to be aware about the size, the type, the capacity, the distribution and other parameters of storage equipment in order to be ensure reliable functionalities.

4 CONTEXT MODELLING AND REASONING APPROACH FOR MICROGRIDS

In this section, we provide the definition of the proposed microgrid’s context model as well as the proposed reasoning mechanism. But before starting, it is necessary to show how the context information is to be used in the control process.

Fig. 2 depicts the abstract architecture view of the microgrid control system which is composed of three main layers: (i) Physical Layer: which represents the process and field components of the subsystems defined in Section 3. (ii) Context Acquisition Layer: which is the layer subject of the current paper contributions. Finally, (iii) Control Layer: which holds the controller of the microgrid. In the following, the functionalities of the second layer, i.e., Context Acquisition Layer, are introduced.

Functionalities of the physical layer such as gathering data from the heterogeneous sources, as well as functionalities of the control layer such as defining reconfiguration and adaptation policies are out of the scope of this paper.

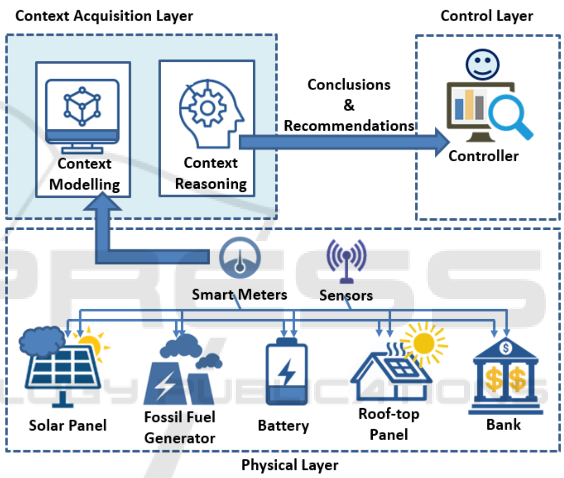

Figure 2: Microgrid control system abstract architecture.

4.1 Ontology-based Microgrid Context

The purpose behind defining the microgrid context model using the ontology is to provide an easy understandable model. Also, ontology based models are easily extensible, so whenever new concepts need to be added the basic model could be reused. Moreover, although the contexts in a microgrid are not easy to define and even complex, the ontology allows us to derive the formal semantic knowledge about the microgrid subsystems, their components, and constraints.

At a given moment, the microgrid controller may not need every context detailed information, but it is of great usability to provide a holistic context model. Most of the previous works provide a context modelling for some specific use cases like the context-aware load-balancing, the context-aware electric vehicle charging, the context-aware power generation

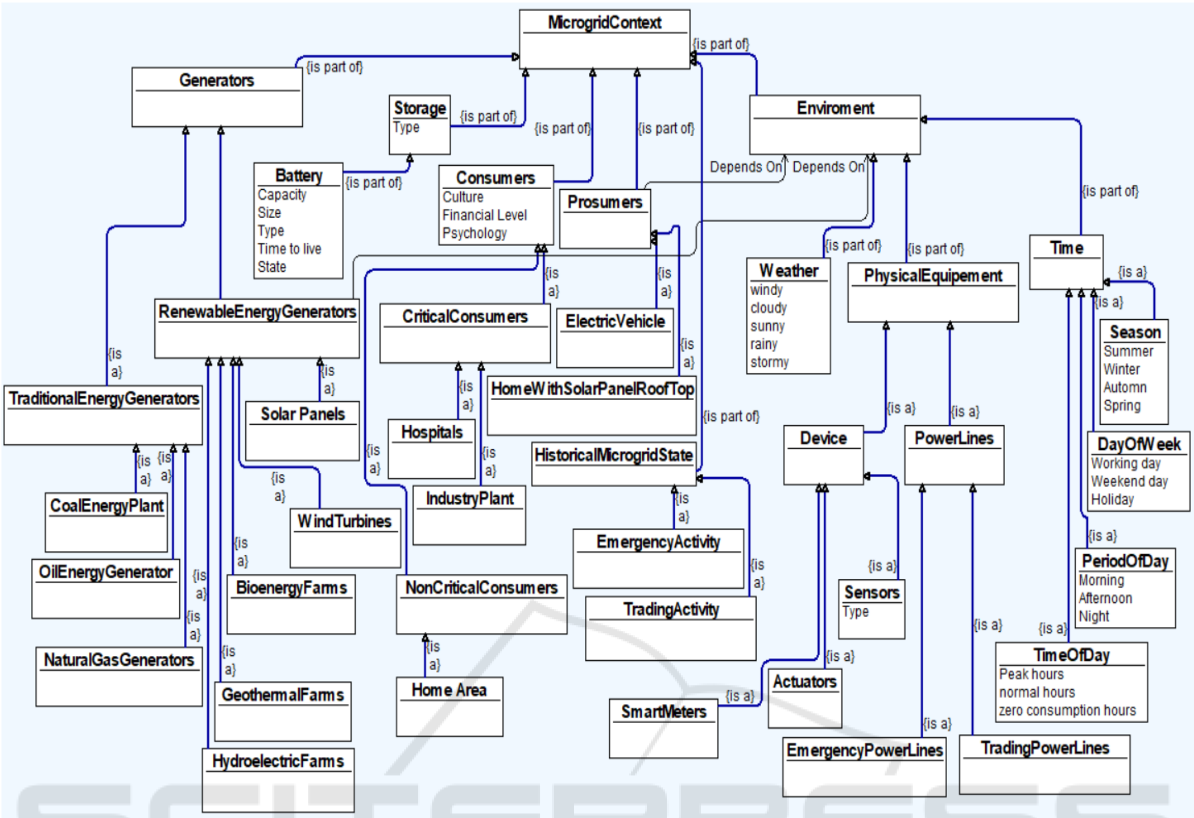

Figure 3: Microgrid Context Model Using OWL.

forecasting. However, none of them provide a context definition for the whole microgrid. It is difficult to model every single case and every single scenario, especially with the expanded subsystems as presented in Section 3. Nevertheless, we provide a holistic context model for microgrids context-awareness which incorporates the most used (conventional) facets. Whenever necessary, the model could be extended, i.e., new ontology entities and deeper details could be added.

Fig. 3 depicts the proposed microgrid context ontology. The model is presented based on the OWL (Web Ontology Language). The considered entities are the entities related to the control task and that can affect the intelligent functionalities such as the load balancing, the peak shaving, the generation optimization, the storage optimization and others. The microgrid context is formed mainly with six main parts: (i) generators, (ii) storage, (iii) consumers, (iv) prosumers, (v) environment, and (vi) historical state.

Generators Ontology: This ontology entity is used to represent the electricity generators within a microgrid system. It is the generalization of the renewable and traditional energy generators where each kind is a generalization of some specific classes of generation such as the *Solar Panels* or the *Coal Energy Plant*. For space constraints, we detail one example of context information of solar panels. It is important for the controller to get some parameters such the voltage at maximum power point (VMP), the open circuit voltage (VOC), the fill factor (FF), etc. Also the number of panels, their size, etc.

Storage Ontology: This ontology entity is used to represent the electricity storage system in a microgrid. The *Battery* specific class is a conventional type of electricity storage. Information required are for example the power requirement range (expressed in kW), the standing time interval (in hours), the nominal AC coupling voltage (VAC), the dimensions, the percentage of over and under voltage allowance, etc.

Consumers Ontology: This ontology entity is used to represent the consumers of electricity within a microgrid. In general, consumers can be characterized by their culture (for example consumers in a developed country do not consume the same amount of consumers in developing ones). Even the financial level and psychology (how is the susceptibility to accept and use new restrictions and technologies, etc.) can influence the consumption rates. Consumers are classified into two classes according to their priority: *CriticalConsumers* and *NonCriticalConsumers*.

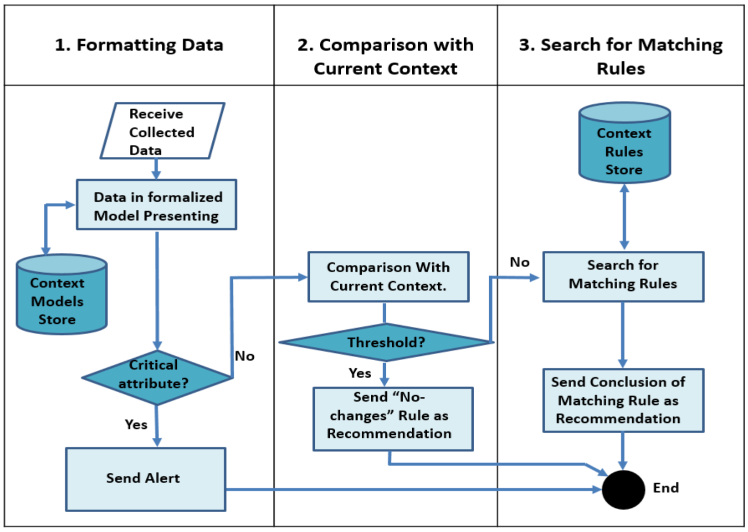

Figure 4: Context Reasoning Process.

Prosumers Ontology: This ontology entity is used to represent the prosumers such as the *ElectricVehicle* and *HomeWithSolarPanelRoofTop* specific classes. Information about how many home has the roof-top solar panels and what are their characteristics is necessary. Also the number of potential usage of electric vehicles and whether it disposes of solar panels are useful to know.
Environment Ontology: This ontology entity is used to describe the environmental influencers on the context of control of the microgrid. Three main conditions are taken into account which are the timing, weather forecast, and physical equipment. The time is represented through the specific class *Time* which is a generalization of the *Season*, *DayOfWeek*, *PeriodOfDay*, *TimeOfDay* which impact directly the scheduling, the generation, and the load balancing tasks.

The *Weather* specific class also is important to the renewable energy generation and that is generally feeded by the weather forecast service providers. The *PhysicalEquipment* specific class represents the physical sources of the context or let say the low level context. It is classified into two types of microgrid equipment: *Devices* such as the sensors, actuators, and smart meters, and the *power lines* especially the emergency and trading power lines. Being aware about the emergency power lines for example, helps to set the emergency reconfigurations in case of internal faults or also in case of faults in neighbor microgrids.
Historical Microgrid State Ontology: This ontology entity is used to describe the previous activities that a microgrid has performed. Historical information has an influence on the decision making for current and future configurations/control tasks.

There are dependencies between the different entities, we present among them the dependency of the renewable energy generators and the prosumers to the environment ontology.

4.2 Rule-based Context Reasoning

In this section, the proposed lightweight context reasoning process is introduced. The process flow is described with three steps as depicted in Fig. 4: (1) formatting the collected data, (2) comparison with the current context, and (3) search for matching rules.

Before providing the pseudo-code of each step, it is noted that we assume that at any given instant, the microgrid has a context (i.e., a state) that can be changed according to the results of the provided reasoning process. It is also assumed that the new found conclusions are forwarded to the controller of the microgrid (control layer in Fig. 2), and the latter has to make decisions based on its control protocols (related to tariff calculation, sustainability policies, security policies, optimization policies, etc.). The role of the proposed processes in this paper is to get and recognize context changes.

4.2.1 Formatting Data

Sensors and metering devices provide real numerical values that are not meaningful or at least not helpful in deducing the semantic relations between the facts related to the context entities (as shown in the ontology definition in Section 3). That is why, we propose the following process to represent the sensed data.

We note that, a Context Attributes Model Store (CAMS) is provided as input to the process. The store contains context attributes models where each model represents one context attribute and its related labels as depicted in Fig. 5. A label is a description of the range of values of the attribute. For example, *temperature* is a context attribute of the weather ontology. The interval of values [0, 10] is described as low temperature, [10, 25] is medium temperature, and [25, 35] is high temperature. Hence, *low, medium,* and *high* are defined as labels. In the same way, the consumption rate (in kW) of one residential unit is an attribute of the non-critical consumers entity that belongs to the consumers ontology. The values between [0, 70] is labelled as very low, between [70, 250] is labelled as low, between [250, 600] is labelled as medium, and between [600, 1200] is labelled as high.

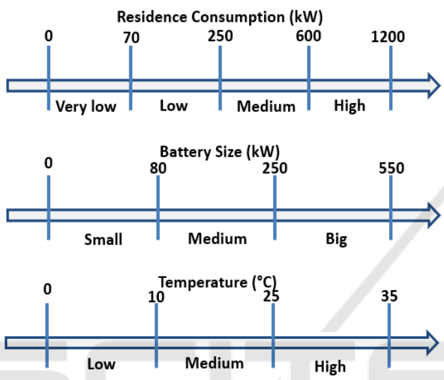

Figure 5: Context attributes model.

Some of the context attributes are given a critical importance and receiving an information related to one of them implies the immediate sent of an alert to the controller. For example, if the sensor of fires detects a fire the process has to end immediately and send an alert message to the controller. Same case for situations related to the storms or blackouts.

Data formatting pseudo-code is as follows:

Step 1: Parse received values and check if one of the critical attributes contains a value.

Step 2: For each received attribute, read the related context attribute model from the context attribute model store CAMS and decide which label is to be assigned to the attribute by comparing the value to the minimum and maximum values of each range, i.e., we say that the label of a range of one model is assignable to the input x_i if:

$$x_i \in [RangeMinValue, RangeMaxValue] \qquad (1)$$

Step 3: Generate a context row using the retrieved labels. A context row, denoted by CR is defined as a set of context items denoted by ci, hence $CR = \{ci_1, ci_2, ..., ci_n\}$ where n is the number of context items. A context item is defined as a tuple $ci =< type, label, value >$ where *value* is the collected/sensed/measured value, *label* is the label of the context attribute determined in the previous step, and *type* is the semantic type of the input and it is presented by its root through the ontologies o_i, entities e_j, and attributes a_k. For example, the context item of the temperature will be as follows: $ci =< o_{environment}.e_{weather}.a_{temperature}, medium, 15 >$. The context item of the residential unit consumption will be as follows: $ci =< o_{consumers}.e_{noncritical}.e_{residence}.a_{consumption}, low, 100 >$.

4.2.2 Comparison with Current Context

Given that context awareness should be provided in real time in order to be efficient, many calculations have to be processed. However, the input data could be the same or slightly different from the current one in many situations. So calculations will produce nearly same results which have not an impact on the context. For example, for the demand response handling task, knowing the period of the day (morning /afternoon or night) may help to reduce the calculations during some hours of each period. For instance, supposing that we are in the night, the periodic supervision of the consumption rates indicates that it is a low consumption. Thus, there is a need to a method allowing to avoid triggering the reasoning process that might produce similar results.

To avoid leading non-useful computations, we propose to create an intermediate step that compares the new context values with the current ones. The comparison is based on a predefined similarity threshold of the values of the context row. The threshold is defined according to the assumptions and preferences of the system owners, for example 10%.

Let $X = \{x_1, x_2, x_3, ..., x_n\}$ be the sensed values of the input context row and $V = \{v_1, v_2, v_3, ..., v_n\}$ be the values of the current context row CR. To see how much the new sensed values are different from the current ones the following formula is applied:

$$P = \sum_{j=1}^{n} (|x_j - v_j| / [(x_j + v_j)/2] \times 100)/n \qquad (2)$$

For example, let the current context row values be $V = \{35, 1, 1100\}$ and let the sensed values be $X = \{30, 1, 1000\}$. The percentage P is 8.3% which is less than the threshold 10%. In this case, it is recommended to not re-trigger the search for rules and rather "no changes recommendation" can be sent to the controller because the difference is negligible and does not imply a context change.

If the difference is greater than the threshold then a search for the matching context rule is necessary.

4.2.3 Search for Matching Rules

In order to find the proper context, a context rule store (CRS) is defined. A rule has the form of *condition* $\longrightarrow$ *conclusion*, where the condition is a conjunction of a set of premises. The context items of the input context row are used as premises of the rules, and the conclusions of rules will be sent to the controller as recommendations of reconfigurations. In this phase of the process, premises are given priorities allowing to facilitate the rule matching process as detailed in the following. The pseudo-code of the matching rule search is given as follows:

Step 1: Select the set of triggerable rules. This is done based on which attribute has changed its values (i.e., we select rules containing as premises the related attributes).

Step 2: If only one rule is selected in Step 1 then go to Step 3. Else if more than one rule is selected in Step 1 then compare the number of premises of the rules. The longest rule is the winner rule. Else if there are more than one winner rule then;

Check the priorities of the attributes. Select the rule containing the attribute having the highest priority. If all attributes are of equal priorities then calculate the Euclidean distance between the input context row values X and the conditions of the selected rules R_k as follows

$$d(X,R_k) = \sqrt{\sum_{i=1}^{n}(x_i - r_{ki})^2} \tag{3}$$

Where r_{ki} is the mid-range of the relevant attribute range. The winner rule is the one having the minimal distance to X.

Step 3: Trigger winner rule. If all rules selected in Step 1 are triggered, then finish. Else, go to Step 2.

The conclusions of all triggered rules will be sent to the controller as recommendations/suggestions that help on establishing the proper control strategies. For example, a recommendation to sell energy when there is surplus of renewable energy generation and low consumption rates, etc.

5 FORMAL CASE STUDY

Let us consider a microgrid called *mg* that has these features: *mg* relies on the utility main power grid and a ten solar panels plant. *mg* should supply one residential area (composed only of homes) and one hospital. *mg* has also one battery as a storage system. Using the proposed microgrid context ontology, in the current case the controller of *mg* should handle: (i) the renewable energy generation subsystem, (ii) the consumers subsystem, and (iii) the storage subsystem. The traditional electricity generators and the prosumers are not considered. Table 1 depicts some of the context attributes stored in CAMS.

Table 1: Context Attribute Labels.

Attribute name	Values
Connection to main grid	on, off
Solar panel production	high, medium, low
Battery reserve	empty, full, half full
Battery activity	charging, discharging, idle
Consumer priority	critical, non-critical
Weather	windy, cloudy, sunny, rainy, stormy
Season	summer, winter, autumn, spring
Day of the week	working day, week-end day, holiday
Period of the day	morning, afternoon, night
Time of the day	peak, normal, zero-consumption hours
Consumption	very low, low, medium, high, very high

Let the current context row of the microgrid, denoted by *CR* be composed with the set of context items defined in Table 2.

Table 2: Current Context Items.

Id	tuples
ci_1	$< o_{HistState}.e_{GridConn}.a_{mode}, OnMode, 1 >$
ci_2	$< o_{Generators}.e_{Renew}.e_{PV}.a_{ProdRate}, medium, 50\% >$
ci_3	$< o_{Storage}.e_{Battery}.a_{Reserve}, halfFull, 50\% >$
ci_4	$< o_{Environment}.e_{Weather}.a_{Sunlight}, sunny, 80\% >$
ci_5	$< o_{Environment}.e_{Time}.a_{Season}, summer, 3 >$
ci_6	$< o_{Environment}.e_{Time}.a_{DayOfWeek}, workingday, 1 >$
ci_7	$< o_{Environment}.e_{Time}.a_{PeriodOfDay}, morning, 1 >$
ci_8	$< o_{Consumers}.e_{NonCritical}.a_{Residence}, low, 110 >$

Let the input vector of collected data be as follows I= { (Connection to the grid = true), (SolarPanelProduction= 1000 Wh), (BatteryReserve= 20%), (Weather= 60% sunny), (DateTime= 04.07.2010:11:30:00), (ConsumptionRate= 350 kW)}. Following the proposed approach, the first step, which is the data formatting step, processes the data vector I using the CAMS and results on the following context row, denoted by $R = \{ci_1, ci_2, ci_3, ci_4, ci_5, ci_6, ci_7, ci_8\}$:

Table 3: Generated Context Items.

Id	tuples
ci_1	$< o_{HistState}.e_{GridConn}.a_{mode}, OnMode, 1 >$
ci_2	$< o_{Generators}.e_{Renew}.e_{PV}.a_{ProdRate}, medium, 50\% >$
ci_3	$< o_{Storage}.e_{Battery}.a_{Reserve}, empty, 20\% >$
ci_4	$< o_{Environment}.e_{Weather}.a_{Sunlight}, sunny, 80\% >$
ci_5	$< o_{Environment}.e_{Time}.a_{Season}, summer, 3 >$
ci_6	$< o_{Environment}.e_{Time}.a_{DayOfWeek}, workingday, 1 >$
ci_7	$< o_{Environment}.e_{Time}.a_{PeriodOfDay}, morning, 1 >$
ci_8	$< o_{Consumers}.e_{NonCritical}.a_{Residence}, medium, 350 >$

Since any of the attributes is critical, no alerts will be sent to the controller and the process goes to the next step which is the comparison with the current context step. A quick parsing of the attributes returns those that have changed their values which are the battery reserve and the consumption level. Therefore, calculating the percentage of the change is to be done.

The corresponding numerical values of the contexts are given as follows: V= {1, 50%, 50%, 80%, 3, 1, 1, 110}, and X= {1, 50%, 20%, 80%, 3, 1, 1, 350}. The difference calculation gives $P = \sum_{j=1}^{8}(|x_j - v_j|/[(x_j + v_j)/2] \times 100)/8 = 23.75\%$. Given that the threshold of the difference between the current and new context is of 10%, it is necessary to move to the next step which is the search for the matching context rules. For this, the CRS is defined as depicted in Table 4.

Applying the first step of the third phase results in the recognition of R1, R7 and R8 as triggerable rules. Then, according to the second step of this search a comparison between the selected rules in terms of number of premises give the following results: (R1, 2), (R7, 3), and (R8, 3). There is no longest rule because R7 and R8 have the same size of premises.

In this case, a comparison of the premises priority is performed. In this case study, more importance is given to the integration of renewable energy. Thus, whenever a rule is dealing with renewable energies it has more priority over other rules. As conclusion, R8 is the first rule to trigger. Triggering R8 gives as a result *Switch on to main grid*. Then, R7 and R1 are triggered successively. Therefore these recommendations are sent to the controller of *mg*: switch on to main grid and start charging batteries.

6 PERFORMANCE EVALUATION

6.1 Numerical Analysis

A major challenge of today's smart applications is its level of awareness and its efficiency in terms of both: (i) reliable behaviours, and (ii) computational resources consumption. For this, efficiently manage and exploit the collected (and/or sensed/measured) data is a key factor that allows to assess the quality of an approach. In this context, the proposed context reasoning in this paper introduces a comparison phase as an intermediate phase between the data representation phase and the rules matching phase at the aim of reducing possible non-needed computations, i.e., those that end with the same conclusion. Therefore, to make our process smart, a threshold of difference is defined to decide about the similarity of the current context values and the new ones, i.e., how much current and new contexts are different.

To show more the benefits of the contribution, let us consider the current context row $A = \{ci_1, ci_2, ci_3\}$ composed of three context items where the values are given by $V = \{35, 1, 1100\}$. The first attribute represents the percentage of the renewable energy generation and it has this context attribute labels: low for the interval [0%, 33%], medium for the interval [34%, 66%], and high for the interval [67%, 100%]. The second attribute represents the connection to the utility grid (1 for true, and 0 for false). Lastly, the third attribute represents the consumption (in Wh) and it has this context attribute labels: low for the values in [0, 1000], medium for the values [1000, 2000], and high for the values [2000, 3000].

Let us consider the following five vectors representing five periodic context measurements during one morning: $V_1 = \{32, 1, 1020\}$, $V_2 = \{40, 1, 1200\}$, $V_3 = \{35, 1, 900\}$, $V_4 = \{35, 1, 1400\}$, and $V_5 = \{35, 1, 1488\}$. Calculating the percentage difference P, between A and the collected vectors results in values less than the threshold as depicted in Fig. 6.

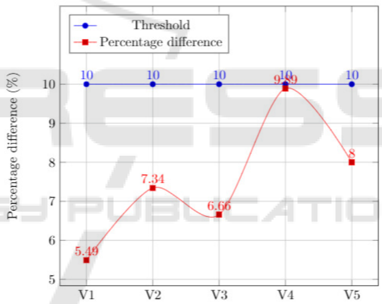

Figure 6: Comparison between current and new rows.

Given that the reconfigurations are costly, the slight fluctuations of the generation and consumption inside one interval of values or around its bounds and below the threshold value should not generate context changes. Defining a threshold values allows to determine the supportable possible fluctuations within one defined context. This phase allows to reduce considerably non-necessary rule matching processes, thus allows to improve the computational resources usage.

6.2 Discussion

The proposed approach shows the ability to express the relation of the control task to the related context. It introduces a definition of the context ontology's that may influence the behaviour of the microgrid controlling tasks and contributes to the enrichment of the development of the system intelligence. Compared to the works presented in Section 2.2, and to the best of

Table 4: Rules Samples.

Id	Rules
R1	(BatteryReserve is empty) and (Weather is sunny) ⟶ Start charging batteries
R2	(Weather is sunny) and (PeriodOfDay is morning) and (SolarPanelProduction is high) ⟶ Switch off from main grid
R3	(Weather is sunny) and (PeriodOfDay is afternoon) ⟶ Stop charging batteries
R4	(Season is winter) and (ConsumerPriority is critical) ⟶ Switch on to main grid
R5	(PeriodOfDay is night) and (BatteryActivity is discharging) ⟶ Turn batteries on idle mode
R6	(Consumption is low) and (Weather is sunny) and (Season is summer) ⟶ Switch off from main grid
R7	(Consumption is medium) and (Weather is sunny) and (Season is summer) ⟶ Switch on to main grid
R8	(Consumption is medium) and (Weather is sunny) and (SolarPanelProduction is medium) ⟶ Switch on to main grid

our knowledge, the current work is the first to define a microgrid ontology that takes into consideration the four system perspectives: generation, consumption, prosuming, and storage.

The three-steps reasoning process is lightweight and provides computational resources reduction thanks to the second step, i.e., the “comparison with current context step”. In fact, it allows to improve the semantics of the context information and avoid non-useful calculations. Compared to the works reported in (Skillen et al., 2014), (Quinn et al., 2017), our proposed reasoning method provides better resources usage thanks to the filtering of the “comparison with current context step”. Also, recognizing urgent attributes in step 1 allows to lighten the computational overhead for critical situations. Search for matching rules process execution is not triggered if critical attributes are obtained also when there are small negligible changes.

The efficiency of the proposed contributions would be more verifiable by the development of the physical and control layers which allow to provide a complete process of a context-aware application work flow.

7 CONCLUSIONS

The current paper introduces an approach of context modelling and reasoning for micrograms. A generalized and holistic design of the microgrid is presented, the design defines four subsystems: generators, consumers, prosumers, and storage. The current paper introduces a microgrid ontology based on the proposed design to present the different parts that can impact the context of the system control. The ontology is defined using the OWL language and it is general, holistic, and extensible. The proposed context model also shows the dependencies between the different subsystems of a microgird and helps to find common understanding of the microgrid context. Then, a three-steps process of rule-based context reasoning is introduced. The reasoning process has as originality the lightweight process thanks to a filtering mechanism that is processed before the rules search. This process allows to reduce computational resources usage. A formal case study is conducted and the suitability of the context deduction is demonstrated. In future work, many interesting research directions could be tackled. The definition of reconfiguration and adaptation mechanisms to be performed by the controller of the microgrid needs to be studied. Also, application of the proposed concepts to a real-world case study would be of great importance.

REFERENCES

Al-Barazanchi, S. A. M. and Vural, A. M. (2015). Modeling and intelligent control of a stand-alone pv-wind-diesel-battery hybrid system. In *2015 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT)*, pages 423–430. IEEE.

Alegre, U., Augusto, J. C., and Clark, T. (2016). Engineering context-aware systems and applications: A survey. *Journal of Systems and Software*, 117:55–83.

Cheng, X., Yang, J., and Xia, L. (2018). A service-oriented context-awareness reasoning framework and its implementation. *The Electronic Library*.

Choi, C., Esposito, C., Wang, H., Liu, Z., and Choi, J. (2018). Intelligent power equipment management based on distributed context-aware inference in smart cities. *IEEE Communications Magazine*, 56(7):212–217.

Degha, H. E., Laallam, F. Z., and Said, B. (2019). Intelligent context-awareness system for energy efficiency in smart building based on ontology. *Sustainable Computing: Informatics and Systems*, 21:212–233.

Dongol, D., Feldmann, T., and Bollin, E. (2018). A model predictive control based peak shaving application for a grid connected household with photovoltaic and battery storage. In *SMARTGREENS*, pages 54–63.

Donohoe, M., Jennings, B., and Balasubramaniam, S. (2013). Context-aware microgrid storage using electric cars. In *IEEE PES ISGT Europe 2013*, pages 1–5. IEEE.

Donohoe, M., Jennings, B., and Balasubramaniam, S. (2015). Context-awareness and the smart grid: Requirements and challenges. *Computer Networks*, 79:263–282.

Fkaier, S., Romdhani, M., Khalgui, M., and Frey, G. (2016a). Enabling reconfiguration of adaptive control systems using real-time context-aware framework. In *2016 IEEE/ACS 13th International Conference of Computer Systems and Applications (AICCSA)*, pages 1–8. IEEE.

Fkaier, S., Romdhani, M., Khalgui, M., and Frey, G. (2016b). R2tca: New tool for developing reconfigurable real-time context-aware framework—application to baggage handling systems. In *Proc. Int. Conf. Mobile Ubiquitous Comput., Syst., Services Technol.(UBICOMM)*, pages 113–119.

Fkaier, S., Romdhani, M., Khalgui, M., and Frey, G. (2017). Context-awareness meta-model for reconfigurable control systems. In *ENASE*, pages 226–234.

Gomes, L., Sousa, F., and Vale, Z. (2017). Enaplug–an environmental awareness plug to test energy management solutions for households. In *EPIA Conference on Artificial Intelligence*, pages 253–259. Springer.

Khabou, N., Rodriguez, I. B., and Jmaiel, M. (2019). An overview of a novel analysis approach for enhancing context awareness in smart environments. *Information and Software Technology*, 111:131–143.

Kyselova, A., Verbitskyi, I., and Kyselov, G. (2016). Context-aware framework for energy management system. In *2016 2nd International Conference on Intelligent Energy and Power Systems (IEPS)*, pages 1–4. IEEE.

Li, X., Eckert, M., Martinez, J.-F., and Rubio, G. (2015). Context aware middleware architectures: survey and challenges. *Sensors*, 15(8):20570–20607.

Meng, Z. and Lu, J. (2015). A rule-based service customization strategy for smart home context-aware automation. *IEEE Transactions on Mobile Computing*, 15(3):558–571.

Obaidat, M. S., Reddy, N. R., Venkata, K. P., and Saritha, V. (2018). Context-aware middleware architectural framework for intelligent smart grid data management. In *2018 International Conference on Computer, Information and Telecommunication Systems (CITS)*, pages 1–5. IEEE.

Quinn, S., Bond, R., and Nugent, C. (2017). Ontological modelling and rule-based reasoning for the provision of personalized patient education. *Expert Systems*, 34(2):e12134.

Radzi, N., Suhaimy, N., Ahmad, W., Ismail, A., Abdullah, F., Jamaludin, M., and Zakaria, M. N. (2019). Context aware traffic scheduling algorithm for power distribution smart grid network. *IEEE Access*, 7:104072–104084.

Reddy, N. R. S. and Krishna, P. V. (2018). Context aware power management in smart grids using load balancing approach. *International Journal of Advanced Intelligence Paradigms*, 11(3-4):335–347.

Sezer, O. B., Dogdu, E., and Ozbayoglu, A. M. (2017). Context-aware computing, learning, and big data in internet of things: a survey. *IEEE Internet of Things Journal*, 5(1):1–27.

Skillen, K.-L., Chen, L., Nugent, C. D., Donnelly, M. P., Burns, W., and Solheim, I. (2014). Ontological user modelling and semantic rule-based reasoning for personalisation of help-on-demand services in pervasive environments. *Future Generation Computer Systems*, 34:97–109.

Stavropoulos, T. G., Rigas, E. S., Kontopoulos, E., Bassiliades, N., and Vlahavas, I. (2014). A multi-agent coordination framework for smart building energy management. In *2014 25th International Workshop on Database and Expert Systems Applications*, pages 126–130. IEEE.

Suslov, K., Solodusha, S., and Gerasimov, D. (2016). Determination of parameters of adaptive law for the control of an off-grid power system. In *2016 5th International Conference on Smart Cities and Green ICT Systems (SMARTGREENS)*, pages 1–7. IEEE.

Wang, X. H., Zhang, D. Q., Gu, T., and Pung, H. K. (2004). Ontology based context modeling and reasoning using owl. In *IEEE annual conference on pervasive computing and communications workshops, 2004. Proceedings of the second*, pages 18–22. Ieee.

Zamazal, O. (2020). A survey of ontology benchmarks for semantic web ontology tools. *International Journal on Semantic Web and Information Systems (IJSWIS)*, 16(1):47–68.

Zhuikov, V. and Kyselova, A. (2013). Integration of context-aware control system in microgrid. In *2013 IEEE XXXIII International Scientific Conference Electronics and Nanotechnology (ELNANO)*, pages 386–390. IEEE.