A Smart Adaptive Switching Module Architecture Using Fuzzy Logic for an Efficient Integration of Renewable Energy Sources. A Case Study of a RES System Located in Hulubești, Romania

Simona-Vasilica Oprea 1, Adela Băra 1,*, Ştefan Preda 1 and Osman Bulent Tor 1,2

1 Department of Economic Informatics and Cybernetics, Bucharest University of Economic Studies, Romana Square 6, 010374 Bucharest, Romania; simona.oprea@csie.ase.ro (S.-V.O.); stefan.preda@oracle.com (S.P.); osman.tor@epra.com.tr (O.B.T.)
2 Engineering, Procurement, Research and Analysis—EPRA, Çankaya/Ankara 06800, Turkey
* Correspondence: bara.adela@ie.ase.ro; Tel.: +40-752-29-4422

Received: 2 July 2020; Accepted: 27 July 2020; Published: 29 July 2020

Abstract: Electricity generation from renewable energy sources (RES) has a common feature, that is, it is fluctuating, available in certain amounts and only for some periods of time. Consuming this electricity when it is available should be a primary goal to enhance operation of the RES-powered generating units which are particularly operating in microgrids. Heavily influenced by weather parameters, RES-powered systems can benefit from implementation of sensors and fuzzy logic systems to dynamically adapt electric loads to the volatility of RES. This study attempts to answer the following question: How to efficiently integrate RES to power systems by means of sustainable energy solutions that involve sensors, fuzzy logic, and categorization of loads? A Smart Adaptive Switching Module (SASM) architecture, which efficiently uses electricity generation of local available RES by gradually switching electric appliances based on weather sensors, power forecast, storage system constraints and other parameters, is proposed. It is demonstrated that, without SASM, the RES generation is supposed to be curtailed in some cases, e.g., when batteries are fully charged, even though the weather conditions are favourable. In such cases, fuzzy rules of SASM securely mitigate curtailment of RES generation by supplying high power non-traditional storage appliances. A numerical case study is performed to demonstrate effectiveness of the proposed SASM architecture for a RES system located in Hulubești (Dâmbovita), Romania.

Keywords: smart adaptive switching module; fuzzy logic; sensors; microgrid; RES integration

1. Introduction and Literature Survey

Microgrids are complex, variable size power systems encapsulating interconnected distributed energy resources (especially RES), storage devices, and loads, including electric vehicles (EVs), that normally synchronously connected to main power grid, but can also operate in island mode (off-grid) [1]. They are developed within a defined electrical boundary, such as campuses, business/industrial parks, and remote residential or touristic areas, etc. The electricity consumption in microgrids should be differently approached whether they are connected to the grid or not [2]. A comprehensive set of definition and classification of microgrid emphasizing challenges and opportunities of the converter-based microgrids is provided in [3]. More definitions of the microgrid concept and depiction of components show its complexity and potential [4,5]. Since feed-in tariff rates are higher than the local rates, priority is to consume as much as possible from the local sources. The on-grid systems typically do not have storage devices, thus consuming when RES is available...
is essential to reduce the electricity costs. The principle of consuming when RES is available also applies to off-grid systems, as it reduces the electricity consumption at night, allowing acquisition of a smaller-size storage system, thus substantially reducing the costs. Another challenge for off-grid systems is to consume during the day to reduce the consumption at night and prevent the storage system from operating at low battery level. Thus, even if a RES system can provide electricity when batteries are 45% loaded, the load should be avoided in order to preserve the batteries lifetime being advisable to resupply when the batteries level is over 60% or 70%. As for the hybrid systems, it is preferable to consume during the day and operate with the smaller-size storage system and preserve the batteries bank lifetime.

The role of the Information and Communications Technology (ICT) solutions in managing electricity is thoroughly underlined in [6]. This process initially started with simple applications such as MS Excel and continued with more complex software, varying from stand-alone applications, Enterprise Resource Planning (ERP), business intelligence, data warehouses, On-Line Transaction Processing (OLTP), big data, Internet of Things (IoT) to cloud computing. Once the technologies evolve, ICT tools become more complex and can provide enhanced control with the utilization of fuzzy logic and machine learning for managing the electricity consumption.

Several home automation solutions provide insights on the smart home technologies [7]. An affordable IoT-based load control solution using communication protocols that focuses on remotely controlling various appliances using sensors and relays is described in [8]. However, this approach is more focused on home security issues such as preventing fire, locking, etc. Another smart home solution using wireless sensor networks that concentrate on controlling loads to reduce the energy consumption is provided in [9]. Also, recent developments related to the smart home, data security and indoor quality have been reported [10].

To handle variability characteristics of the RES systems at the microgrid level, a fuzzy logic-based control of the storage system and generating units is described in [11] to minimize active power exchange, control the frequency in island operation [12], and peak load sharing [13]. Also, considering the changes in consumption, an energy management system using a fuzzy logic controller for a microgrid with RES and storage facilities as backup aiming to integrate local RES is presented in [14]. In this approach, the focus is more on dynamically adapting RES modes to integrate them either at the microgrid or grid level. A fuzzy logic algorithm is also developed and implemented to integrate RES in [15]. Moreover, an energy management approach for residential consumers with a multi-agent system to control loads is proposed to reshape the load curve by adequately scheduling appliances considering the consumers’ preferences [16]. The schedule is designed using a fuzzy logic controller that considers the technical constraints, preferences and minimizes the costs. A more recent multi-agent based research, which also uses machine learning and fuzzy control, focuses on integrating RES at the microgrid level [17]. The fuzzy logic method for energy management is implemented for a house that has a hybrid solar system in [18]. This study explores the possibility of efficient usage of the solar energy ensuring the adequate battery loading. The major drawback is that it only considers the amount of solar energy generated and locally consumed, ignoring the data collected by wind, solar irradiance and temperature sensors.

The fuzzy logic method is suitable in those areas where accurately determining a value or rigorously studying a process is not an end in itself, in other words, an inevitable inaccuracy is accepted [19]. In fact, in practice, there are few situations when an exact determination is required; in most cases, some tolerance in controlling processes or determining certain sizes is accepted. Thus, using the fuzzy logic to efficiently control the load by switching on/off the appliances connected to a RES system is well suited.

The fuzzy logic is recommended to make decisions in so-called “grey” values at the expense of known true, false or one and zero values [20]. This is also the case for RES where the input parameters have “grey” values, such as the wind is intense, less intense or burst, the intensity of the solar irradiance is low, medium, high, etc. The fuzzy logic has a potential implementation in various areas such as
the RES operation with a certain level of uncertainty. Such an example is given in [21] where fuzzy logic is used for a wind turbine to stabilize the voltage generated by the turbine, overcoming the environmental disturbance or the fluctuations of the weather parameters.

The equilibrium that needs to exist between generation and demand in the context of the volatility of the RES output is highlighted in [22]. The authors underline the necessity of a Smart Adaptive Switching Module (SASM) to manage and plan the energy demand. The study emphasizes that SASM suits well in current trend of using the fuzzy logic method in the energy management. Thus, a fuzzy logic management system, from the perspective of using and managing multiple energy sources (photovoltaic—PV, wind, battery storage, fuel cell, etc.), is proposed in [23] as a fair approach in managing the energy flows.

RES operation is characterized by an uncertainty of the generated power mainly due to weather conditions. From this perspective, Demand Side Management (DSM) in microgrids (or RES-powered systems) is necessary to compensate this uncertainty [24]. In DSM area, it is valuable to mention that the energy management includes energy conservation which basically reduces uncertainty and increases the energy efficiency [25]. Thus, the SASM can also contribute to improve the energy conservation by switching on adequate loads.

The fuzzy logic method for optimization provides approximate results with an outstanding degree of accuracy. However, the implementation of the fuzzy method is not recommended particularly by the accuracy of the results; the main reason for using this method is that it can be implemented when other methods fail, especially in those systems in which the dependence on the input parameters is unknown and there is no input-output function. A case that recommends applying the fuzzy logic method is when one or more output factors are not known precisely, or they are ambiguously known. Hence, the SASM fits well in the fuzzy logic paradigm. Referring to the weather input parameters, they cannot be estimated precisely, but they are determined approximately based on weather forecast [26]. Also, the intensity of the solar irradiance is not precisely known because of the uncertainty of the various factors including the weather parameters which have the most pronounced degree of ambiguity. This uncertainty leads to fluctuations in the generated power due to heating or ventilation issues of the components. For example, in case of 3000 rotations/minute turbine, the dynamics of the airflow and the effect of the cooling sub-systems are difficult to predict because it depends on the shape of the carcasses, the existing air currents, the distance of the equipment to other bodies, etc. The ambiguity introduced by the heating and ventilation of the PV panels, which are directly exposed to environmental factors, should be listed. Thus, these aspects recommend using a fuzzy logic to enhance the SASM performance.

The electricity from PV panels is produced during the day, usually from around 9 am to 4 pm, with a generation peak at mid-day. Thus, its output cannot be synchronized with the consumption and it is required to store the surplus and handle the consumption with an intelligent hybrid, advanced or a so-called smart grid inverter [27]. The concept is to evolve from the classical storage in batteries, charging batteries till are full, to the direction to diverting energy flow from the renewable energy sources to batteries and to consumers, to allocate even a part of energy to batteries, other part to consumers and even a third part to be injected to grid.

To compare SASM with an advanced or smart hybrid inverters or controllers, the usage of smart hybrid inverters is exemplified briefly as follows. They operate in different modes considering various priorities [28] (such as: self-use: load > battery > grid, force time use: battery > load > grid (when charging) or load > battery > grid (when discharging), feed in priority: load > grid > battery, back up mode: battery > load > grid, peak and valley mode: battery > load > grid (when charging) or load > grid (when discharging)). However, SASM connects appliances to grid depending on the available of energy (or expected amount of energy), ensuring that batteries do not discharge more than a certain threshold and efficiently use the available energy.

Basically, a SASM with fuzzy logic and an advanced or smart hybrid inverter have a common goal of using renewable energy when it is available. Although the smart hybrid inverter achieves this goal
by prioritizing the energy flow between the grid, battery, and appliances, the SASM with a fuzzy logic maximizes the usage of RES generation through controlled switching of proper electric appliances.

In this paper, a SASM architecture is proposed to efficiently control the consumption and maximize the usage of the RES potential with a fuzzy logic. The proposed SASM has three main features:

- It efficiently controls the consumption by switching the appliances to benefit from RES potential;
- Optimization at power level is adaptive as RES potential depends on the following conditions: weather parameters, power forecast, battery level, etc.;
- SASM is smart in the sense that it uses machine learning and fuzzy logic technologies.

Advantages of the proposed SASM are summarized below:

- It maximizes the usage of available RES generation through adaptive switching of electric appliances;
- It protects battery banks in the microgrid by controlling its discharging level. The SASM can be set to prevent the battery bank from being unloaded below a desired level (i.e., 60%);
- It is adaptive to smaller size battery banks;
- It allows optimizing load versus available renewable generation, thereby, a faster return on investment (ROI);

The originality of this study includes:

- RES operation is optimized taking into account ambiguous values of several input parameters through a fuzzy logic;
- Efficiency of the RES operation is increased in an adaptive manner;
- SASM architecture is composed of three variants for controlling the loads: (i) electrical network topology; (ii) electrical switches; (iii) sensors and smart appliances;
- Practical implementation of the methodology by categorization of appliances and computing resources such as PC, Arduino, Raspberry Pi, sensors, communication protocols, etc.

This paper is organized as follows. Methodology of the proposed SASM approach is described in Section 2. A numerical case study, which is performed to demonstrate effectiveness of the proposed SASM architecture for a RES system located in Hulubești (Dâmbovița), Romania, is presented in Section 3. Conclusion drawn from the study is summarized in Section 4.

2. Methodology

2.1. Fuzzy Logic Principals

As stated in the introduction, the SASM fits well in the fuzzy logic paradigm. Classical logic is based on two levels: either false or true. In practice, however, we often have intermediate levels between false and true. For example, we may say that a glass of water is full or empty. However, we may also say that it is almost full, almost empty or half empty. Similarly, we may say that outside is very cold, cold, less cold, warm, warmer, very hot or canicular. Hence, we can fit the temperature in 7 intervals, and the imprecision is even higher, considering that for each person these intervals of temperature can be different depending on perception.

The fuzzy logic SASM operates with fuzzy sets and membership functions, graphically represented as in Figure 1. For the outside temperature example above, the fuzzy set is denoted by \( A = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7\} \) with \( x_1 = \) very cold; \( x_2 = \) cold; \( x_3 = \) less cold; \( x_4 = \) warm; \( x_5 = \) warmer; \( x_6 = \) very hot; \( x_7 = \) canicular. \( A \) is a set of values in the universe of interest \( X \) (temperature in this case). The function which is usually denoted by \( \mu_A(x) \) shows the degree to which it belongs to a point \( x \) of the input field. Thus, for a fuzzy set \( A = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7\} \) and a membership function, we can associate the elements \( x_1, x_2, x_3, x_4, x_5, x_6, x_7 \) with the following values 0.2, 0.3, 0.5, 1, 0.4, 0.3, 0.3. In fuzzy logic \( \mu_A(x) \in [0, 1] \), being defined on an entire interval, while in classical logic \( \mu_A(x) \in \{0, 1\} \) with only two
values: 0 and 1. In [29], a fuzzy set is defined by a set of pairs formed by the points of the universe of interest and values of the membership function: $A = \{x, \mu_A(x) | x \in X\}$. Similarly, in our example, the fuzzy set is: $A = \{"very cold": 0.2, "cold": 0.3, "less cold": 0.5, "warm": 1, "warmer": 0.4, "very hot": 0.3, "canicular": 0.3\} \text{ or } A = \{x_1: 0.2, x_2: 0.3, x_3: 0.5, x_4: 1, x_5: 0.4, x_6: 0.3, x_7: 0.3\}.$

Decision making using fuzzy logic is used in complex systems in which the information is uncertain, unable to be exactly determined, and system constraints are not very well defined. According to [15], there are two approaches for using fuzzy logic to make a decision: the conventional approach, in which the defuzzification process is carried out at an early stage and a fuzzy approach in which defuzzification is accomplished in the final stages. In Figure 2, the most simplified diagram of a fuzzy logic system is depicted [30–32].

In this diagram, the fuzzification block converts the distinct input values into a fuzzy dataset (regarding temperature, values such as $-10 ^\circ C$ and $15 ^\circ C$ are converted to “cold” and “warm”). Thus, the fuzzy input values are transformed into human language (cold, warm, etc.) by mapping input values of specific ranges to human language. The rules that are applied by the deduction engine are built on the knowledge of the domain for which the fuzzy logic system is used (if the system is used to control a heating system based on outside temperature, an example would be “if it is cold, engage more power”). The fuzzy rules are also expressed in human language and define control policies that are tracked by the system. They can be extracted from the existing numerical values related to the system. The deduction engine makes decisions similar to human decision-making, whereas the defuzzification process creates distinct output values based on the fuzzy output set generated by the deduction engine. The output values are different because most physical systems interpret or receive distinct values as...
input. The block diagram in Figure 2 does not have a compensator or a controller that adjusts the output values. This is generally represented as closed-loop control systems [33–35] as in Figure 3.

![Figure 3. Example of a block diagram of a system using a fuzzy logic controller.](image)

In a closed-loop control system that uses a fuzzy controller, the distinct values from the system output are measured by a sensor and represent the input for the fuzzification, thus it generates a corresponding fuzzy set that is input for the fuzzy controller and moreover for the fuzzy deduction engine. The defuzzification process transforms the fuzzy set series created by the deduction engine into distinct values that can be used as input for the system. In this diagram, all the fuzzy components (fuzzification, controller, execution engine, defuzzification) are on the reverse loopback response.

2.2. Mamdani and Takagi-Sugeno Fuzzy Controllers

The Mamdani fuzzy controllers are systems using an input vector of \( n \) functions \( x(t) \in R^n \) to obtain at any time \( t \) a vector of \( y(t) \in R^m \) [36]. The input functions \( x(t) \) are fuzzy and characterized by uncertainty. The output is a vector of exact values, so-called crisp. The exact output values (e.g., a true signal or 1 logic) are determined by real-time operation of the RES system.

Figure 4 exemplifies the operation of a system with two input variables \( x_1, x_2 \), and one output variable \( y \) as in [37,38] with a Mamdani fuzzy controller using a max-min deduction and centroid method defuzzification.

![Figure 4. The Mamdani fuzzy controller.](image)

In Figure 4, the membership functions for the input and the output variable are also given. Each of the rules 1 and 2 will provide true values for the membership functions of the output variable.
The membership functions are triangular; the ascending slope represents a partition of the membership function; the downward slope represents the other partition. The partitions are characterized by the following equations:

\[ y = a_1 x_1 + b_1 \]
\[ y = -a_2 x_2 + b_2 \]  

(1)

The second partition has a negative slope. The output values are combined by the OR operator (‘OR’), then the final output value is obtained using the centroid method. We note that in case of a Mamdani fuzzy controller, the knowledge about input and output is intuitive representing an advantage. Defuzzification, meant to obtain the distinct output value, involves calculating the area, which is an intensive computational operation. The definition of the fuzzy rules for the input (rules defined on each input variable) and output variables is specific.

The Takagi-Sugeno (T-S) fuzzy controllers allow the adaptive fuzzy control as well as the distributed parallel control of the output, such controllers being used in closed-loop systems (similar to the reaction of the theory of automated systems) that proved high stability. To explain the mathematical model of the fuzzy controller proposed by T-S, we should define a relation of fuzzy implication \( R \):

\[ R: \text{If } (x_1 \text{ is } A_1, \ldots, x_k \text{ is } A_k) \text{ then } y = g(x_1, \ldots, x_k) \]  

(2)

where:

\( y \) — output variable of the system whose value is to be estimated;
\( x_1, \ldots, x_k \) — input variables;
\( A_1, \ldots, A_k \) — fuzzy sets with linear membership functions. These fuzzy sets form a fuzzy subspace on which the \( R \) relation can be applied;
\( f \) — a logical function that exists between the input variables;
\( g \) — function that sets the output value when the input variables \( x_1, \ldots, x_k \) meet a given condition.

For \( R_i, i = 1, \ldots, n \) and the input variables as \( x_1 = x_0^{i_1}, \ldots, x_k = x_0^{i_k} \), the output can be calculated as in [39]:

(a) the output variables \( y^i \) given by \( R_i \) are calculated with function \( g^i \):

\[ y^i = g^i(x_0^{i_1}, \ldots, x_0^{i_k}) = p_i^0 + p_1^i x_0^{i_1} + \ldots + p_k^i x_0^{i_k} \]  

(3)

(b) the true value of \( y = y^i \):

\[ |y = y^i| = |x_0^{i_1} \text{ is } A_1^i, \ldots, x_0^{i_k} \text{ is } A_k^i| \land |R^i| = (A_1^i(x_0^{i_1}) \land \ldots \land A_k^i(x_0^{i_k})) \land |R^i| \]  

(4)

The operator \( \land \) represents the minimal operation that is similar to AND. The expression \(|R^i|\) represents the true value of the fuzzy implication \( R^i \).

For simplicity, \(|R^i| = 1\), therefore the relation of fuzzy implication is always true, so the Equation (4) becomes:

\[ |y = y^i| = A_1^i(x_0^{i_1}) \land \ldots \land A_k^i(x_0^{i_k}) \]  

(5)

(c) the output is given by the weighted average:

\[ y = \frac{\sum |y = y^i| \times y^i}{\sum |y = y^i|} \]  

(6)

The main difference between Mamdani and T-S fuzzy controllers is that in the case of Mamdani, defuzzification of the fuzzy output value is used to obtain the distinct output value, while for T-S the distinct output value is calculated as in Equation (6) [40]. The output value of Mamdani fuzzy
controllers has an expressive, intuitive value, but a slower processing time. The T-S fuzzy controllers have a faster processing time due to the lack of defuzzification and their output value is a weighted average that is more rapidly calculated. According to [9,41], the most commonly used methods for defuzzification are:

- The principle of the maximum membership function is characterized by:

\[
\mu_C(z^*) \geq \mu_C(z)
\]  

where:

\( z^* \) — defuzzification value, the value for which the value of the membership function (\( \mu \)) is greater than or equal to any other value of the membership function that has argument values in the fuzzy set.

- The centroid method given by:

\[
z^* = \frac{\int \mu_C(z) \times zdz}{\int \mu_C(z)dz}
\]  

- The average weighted method characterized by:

\[
z^* = \frac{\sum \mu_C(\bar{z}) \times \bar{z}}{\int \mu_C(z)dz}
\]

- The mean-max membership function method given by:

\[
z^* = \frac{a + b}{2}
\]

where \( a \) and \( b \) are the points of the membership function.

The answer to the question which method is better is a relative one as it depends on the context of the problem. In [9], five criteria are mentioned in the comparison process of the defuzzification methods: continuity, uniqueness, plausibility, simplicity of calculation and weighing mode.

2.3. Components of a RES Systems

Although there is a diversity of RES systems, Figure 5 highlights the main components that are involved in their operation. The energy captured via the Maximum Power Point Tracking (MPPT) controller loads the battery bank. As long as the battery bank voltage is in the operating range of the inverter, it generates an alternative voltage through the Automatic Transfer Switch (ATS) supplying the building. For example, in case of a 24 V inverter, the voltage range for which it generates is between 22–30 V. The charging controller, inverter and ATS can be integrated, forming a hybrid inverter. In terms of optimizing the RES system operation, the principle is the same whether these components are integrated or not.

The Power Forecast Module (PFM) is essential since it estimates the power generated by RES according to the environmental parameters such as solar irradiance intensity, temperature, wind speed, humidity, etc. The methods for estimating the generated power may use machine learning based on Support Vector Machine (SVM) as proposed in [42], nonlinear autoregression time series, Artificial Neural Network (ANN) [43,44], etc. The machine learning techniques for the estimation of the generated power relies on the environmental parameters and other recorded data, such as: history of generated power, currents, voltages, etc.). Other estimation methods are based on sky images provided by satellites. The third category of prediction methods is based on both environmental parameters and sky images provided by satellites [45]. However, it is better to combine these techniques for accuracy reasons. Regardless of the chosen method, if the data used in the processing stage generates large
volumes over time, it is recommended to integrate the power prediction methods with a big data system as in [42].

The Monitoring and Control Module (MCM) can be designed to override the SASM in the sense that if SASM engages specific consumers then a human operator through the MCM can further plug or unplug consumers if needed.

With power forecast and other input data from sensors of a RES system, the main goal of the SASM is to connect consumers or groups of consumers with different power levels in order to maximize the consumption from the local available renewable energy and to maintain the battery bank at a certain charging level, thus ensuring the maximum battery lifetime. According to IEC 60896 [46], the life cycle of a battery is depending on the depth of discharge as in Figure 6. Therefore, the main challenge is to have the battery bank charged over a given voltage level for as long as possible. As noted in [42], it is necessary to avoid discharging the batteries below 30–40% of its maximum capacity, and it is advisable to have the battery bank loaded over 60%.

![Figure 5. Example of a RES system.](image)

![Figure 6. Life cycle in relation with the depth of discharge for a UCG100-12 battery type.](image)
Figure 6 depicts the life cycle regarding the depth of discharge for a particular UCG100-12 battery type, but the principle is similar for any battery type.

At point A, 40% of the energy is consumed from the battery, therefore the battery is not discharged more than 60%; in this mode, the battery supports about 1700 cycles of operation. If we approximate that a discharge cycle corresponds to one day, 1700 cycles correspond to 4 years and 240 days. At point B, 70% of the energy is consumed from the battery, subsequently the battery has 30% energy left, in this case, the number of cycles is reduced to about 760, which corresponds to only 2 years and 30 days. Thus, it explains why it is always strongly recommended for the battery bank to be loaded over 60%.

The SASM module can be used in all types of RES systems. Therefore, in case of an off-grid RES system, the fuzzy logic maximizes the consumption and storage from the available renewable energy. In case of an on-grid RES system, the battery bank is not included, thus the fuzzy logic will maximize the consumption from the available renewable energy as it is more efficient to use the renewable energy than to inject it into the grid, knowing that the price of the injected electricity is smaller than the electricity taken from the grid. As for a hybrid system, the logic fuzzy will also consider the battery charging level as in the case of an off-grid RES system. In addition, the fuzzy logic will maximize the consumption from the available RES and minimize the electricity consumption from the grid, especially when the tariffs are exceedingly high. Based on these principles, the fuzzy rules will change accordingly.

2.4. Consumption Power Levels for SASM Implementation

The SASM involves grouping appliances according to their importance and consumption level. This can be subjective depending on the context. For example, let us assume that for a residential consumer, we have four groups of appliances, which correspond to four power levels:

- **Group 1/Level 1** includes the alarm system, gas sensors, humidity sensors, other necessary sensors if any, some of the lighting system (the most critical bulbs), the main circulation pump directly connected to the boiler (if any);
- **Group 2/Level 2** includes level 1 plus the refrigerator, freezer, the rest of the lighting system, TV, radio, other kitchen appliances, other recirculation pumps in the heating system (if any), vacuum cleaner;
- **Group 3/Level 3** includes level 2 plus some of the most important air conditioning equipment, hydropower, heat exchanger under 1 kW, instantaneous water heater coupled directly to the sink (if any);
- **Group 4/Level 4** includes level 3 plus boiler, hot water tanks or accumulators, pool pump, high power heaters.

According to this classification, level 1 should be fed as constantly as possible. Basically, level 1 and 2 are mandatory levels. Even though the grid connection is not available, the off-grid RES system should provide a minimum level of comfort. Additionally, level 3 adds more load that can often be reduced or switched off, whereas level 4 adds very high-power load. Both level 3 and level 4 appliances have the capacity of storing electricity in a form other than classical.

Consequently, if it is a cold and sunny winter day at 12:30, for instance, the PV energy is excessively available but if the batteries are already full, by connecting the boiler, the water once heated can be used later, when necessary, e.g., when the PV energy is no longer available, in this case, the water is already warm, and thus the consumer uses less electricity from the grid. Another example is the use of the solar energy for heating that can be a complementary source to the traditional heating system. The dwelling may have a gas boiler, thus, additionally connecting the electric heating to the renewable energy decreases the gas consumption.

Therefore, the SASM will gradually connect the four levels depending on the available renewable energy, the external conditions and other factors. The first example is for winter, with low temperature,
sunlight, and fully charged batteries. In this case the level 4 appliances can be coupled. If, on the contrary, it is cold and cloudy and the batteries are unloaded, then only level 1 is connected.

If the off-grid system RES cannot supply the level 1 appliances, the ATS will switch automatically to the grid supply. It can also be switched manually to the grid if necessary (e.g., if it is very cold, the operation of high-power appliances is required, but the off-grid system cannot supply the heating system).

Thus, a fuzzy logic method is proposed to connect the power levels. Accordingly, the input parameters such as the battery voltage, outside temperature, solar irradiance, wind speed, etc. form a set of input values. Therefore, several set of input values, which are based on a fuzzy logic classification algorithm, indicate the connection of the power levels: 1, 2, 3 or 4.

2.5. Storage System Constraints

The efficient operation of an off-grid system involves the usage of the local available RES as much as possible. However, this requirement is not always consistent with the operating characteristics of inverters. The maximum energy taken from the PV panels is in bulk charging state as in Figure 7. The float charging state, without going into details regarding the MPPT controller theory, is required to protect the battery life. According to the charging cycle, the equalization charging is performed by the MPPT on day 28 of each month. It is essential because, for most of the batteries as at this stage, there is a combination of the electrolyte, the battery voltage balancing, and some chemical reactions, representing the rebuilt of the battery.

![Figure 7. The charging status of a battery in MPPT controller (for an EPEVER Tracer3210A controller).](image-url)

In a nutshell, it is useful to keep the battery in bulk charging state, knowing that on the 28th day the MPPT carries out the equalization charging. For example, for a 24 V system, the bulk charging is between the “Low Voltage Disconnect” 22.2 V and “Boost Charging Voltage” 28.8 V. Note that if the battery bank voltage drops to a specific “V cut-off” value, the inverter stops. For a 24 V system, the “V cut-off” is around 22 V, so the “V cut-off” is very close to or equal to “V Low Voltage Disconnect”, at this value the charging controller disconnects the load. Therefore, the battery bank in our example should have a voltage between 22.2 V or “V Low Voltage Disconnect” and 28.8 V or “V Boost Charging". Also, [47] recommends to use the maximum available charging current from the PV panel at the beginning of the battery charging (bulk charging first). Now, the fuzzy logic comes into play. Therefore,
first, if the battery bank tends to drop to the lower limit (“V Low Voltage Disconnect”), we have to disconnect some appliances. Second, if it tends to discharge quickly, we need to disconnect high power appliances or if it tends towards a slower discharge, we need to disconnect the medium or low power appliances. Similarly, if the battery bank tends to charge rapidly towards “V Boost Charging”, we need to connect the high-power appliances, and if it tends to charge slowly towards “V Boost Charging”, we need to connect low power or medium power appliances.

To determine whether the charging is fast or slow, technically, we need to measure the voltage variation of the battery bank using a sensor, for example, by connecting the battery to a computer using an Arduino module to determine the voltage rate of variation. A much simpler approach is to check the power forecast or the solar irradiance intensity or another parameter, for example, the power value generated by the PV panel. If the intensity of the solar irradiance or the power generated by the PV panel is high, then the battery charge rate is also high.

### 2.6. SASM Architecture

As stated in Section 2.4, classification of the appliances is essential to increase efficiency of RES operation. In response, a SASM architecture with three variants is proposed:

- **Variant (a)** through separate circuits of the electrical network (i.e., topology);
- **Variant (b)** by switches for each major appliance;
- **Variant (c)** by means of sensors and smart appliances.

**Variant (a)** Dividing consumers into levels can be done by having separate circuits of an internal electrical network. Each circuit can be connected/disconnected as needed using relays or controllable plugs as in Figure 8. Details are provided in Figures 9–15.

![Figure 8](image)

**Figure 8.** Switches and separate circuits for each consumption level.

As it can be noticed in Figure 8, the SASM is mainly built on a fuzzy mechanism, therefore, using the fuzzy algorithm with input parameters (solar radiation, temperature, wind speed, battery bank voltage, invertor current), the fuzzy logic will choose one of the levels: 1 or 2 or 3 or 4 as a single output value. The logic block will match the output with the command signal of the switches [48] as in Table 1, that shows how SASM is configured to control the loads.

The data is provided in binary representation via an Arduino module and controls an automated scheme consisting of a Logic switch that switches the power levels. Thus, for the output value 1 of the fuzzy logic system, the maximum level is 1 or 001; four switching signals 0001 will be generated by the Logic switch, as illustrated in Figure 9.
For the output value 2 of the fuzzy logic system, the maximum level is 2 or 010; four switching signals 0011 will be generated by the Logic switch, as illustrated in Figure 10.

Similarly, for level 3 and level 4, the operational diagrams are given in Figures 11 and 12.

Figure 9. Switches and separate circuits operational diagram for level 1.

Figure 10. Switches and separate circuits operational diagram for level 2.

Figure 11. Switches and separate circuits operational diagram for level 3.

Connecting the levels does not essentially connect all the appliances at that level; it actually means only the presence of the voltage at the sockets, so that, if level 4 is connected, it does not mean that all high power appliances are connected at the same time, but they can be connected, if necessary. Also, the appliances with thermostat will operate until the temperature reaches the desired level.

Variant (b) In case of using switches for every appliance or major appliance, each appliance will have a switch or relay. When the SASM selects a level via the fuzzy logic algorithm, then a voltage

Table 1. Converting fuzzy output to switching signals.

| Level | Binary Representation | Logic Switching Output |
|-------|-----------------------|------------------------|
| 1     | 001                   | 0001                   |
| 2     | 010                   | 0011                   |
| 3     | 011                   | 0111                   |
| 4     | 100                   | 1111                   |
signal will be generated that will trigger all the switches or relays at that level. The implementation of SASM can be done with a PC, but, at the same time, using Raspberry Pi [31,49], that is a cheap solution. Instead, it offers less flexibility in testing with several fuzzy logic algorithms. Whereas when using a PC, it is easy to choose and test different fuzzy logic algorithms.

A more elaborate sensor-based SASM architecture for dividing appliances into levels using switches for each appliance is given in Figure 13. The transmission of signals from SASM to appliances can be done with Arduino via technologies such as WiFi, Bluetooth, infrared, or Arduino-specific protocols (XBee or ZigBee) [50,51]; the choice of a particular protocol is made according to costs and distances.

![Figure 12. Switches and separate circuits operational diagram for level 4.](image1)

![Figure 13. SASM architecture using switches at the appliance level.](image2)
The operation of this diagram is similar to the operation of the diagrams shown in case of variant (a). The difference lies only in the fact that the output from the Logic switch block is applied to several switches. To highlight it, the operational architectures for “Max level” = 1 and “Max level” = 2 are provided in Figures 14 and 15, respectively.

Figure 14. SASM architecture using switches at the appliance level for “Max level” = 1.

Figure 15. SASM architecture using switches at the appliance level for “Max level” = 2.

**Variant (c)** Controlling appliances by sensors is a relatively new idea, based on the use of a smart home API that can interact with smart home appliances. In this regard, there are countless manufacturers of smart home appliances providing APIs for a smart home. This type of controlling application, using the smart home API, implies the development of a fuzzy logic based SASM on a PC as most of the smart home APIs are intended to be used on a PC. The smart home APIs can access frequently used appliances such as smart light bulbs or smart plugs that connect/disconnect almost any appliance. Obviously, these often-used appliances must be compatible with various APIs such as Amazon Alexa or Google Assistant.

Another relatively new type of smart home API is based on REST protocol that can also be used with Arduino. Also, a smart home environment can be implemented well with Raspberry Pi [52], as
fuzzy logic algorithms can be implemented in Raspberry Pi. Therefore, Raspberry Pi can be a cheap platform for applying the principle of controlling appliances with sensors and smart home APIs.

2.7. Proposed Fuzzy Model for SASM

The SASM may have the following input parameters: intensity of solar irradiance, ambient temperature, wind speed, voltage of the battery bank, current generated by the inverter, power consumed by different appliances (especially from high power appliances), short and medium-term power forecast from the PFM, synthetic data from the big data subsystem, if used, or other data.

In our case, the SASM has the following input from sensors of a RES system and environmental factors:

- Power forecast in W, using various methods (autoregression, ANN);
- Solar irradiance in W/m²;
- Wind speed in m/s;
- External temperature in °C;
- Battery bank voltage in V;
- Current intensity of the inverter in A.

The output of SASM is the power level aiming to enhance the consumption of the RES system (with PV, PV and wind or wind only) from local available renewable energy. Typically, the off-grid or hybrid systems rely on a battery bank. The load controllers in the PV or wind system have the ability to manage the battery level. The Pulse Width Modulation (PWM) or MPPT controllers are designed to protect the batteries, knowing that they are one of the most expensive subsystems. Both PWM and MPPT controllers, when the batteries reach the upper limit of the charging level, completely disconnect the PV panels, protecting the batteries. Apart from this advantage, there is also a disadvantage because after disconnection, no significant electricity is used from the PV panels or the wind turbines. However, the SASM removes this deficiency, detecting that the battery bank is loaded and that the renewable source has potential to generate, it automatically connects the appliances with specific power depending on the input parameters.

To exemplify, the following case is investigated: if the predicted power is high and the intensity of the solar irradiance is also substantial, the SASM connects a high power appliance, which is supposed to be useful in the future. Thus, if the water in the boiler tank is at 35 °C, it connects the boiler and heats the water to a higher temperature. Or, if it is cold outside, it connects the air conditioning for heating, etc.

For choosing the adequate consumption level, we consider four power levels with the following values: 4—corresponds to 100% power level and switches levels s1, s2, s3, s4; 3—corresponds to 75% and switches levels s1, s2, s3; 2—corresponds to 50%, s1 and s2 are simultaneously connected; 1—corresponds to 25% and switches level s1. According to the customization requirements, we can set different percentages or more levels, etc. From the fuzzy thinking standpoint, the levels can be expressed as: 4—Very High (VH) level; 3—High (H) level; 2—Medium (M) level; 1—Low (L) level.

An example of the fuzzy rules can be as follows:

- If the power forecast and the solar irradiance level are very high, regardless of the other parameters, it connects level 4;
- If the power forecast and the solar irradiance level are high and the batteries are already charged, it connects level 4;
- If the power forecast and the solar irradiance level are high, but the batteries are less charged, and the wind has low or medium speed, it connects level 3, etc.

The fuzzy rules are summarized in Table 2.
Table 2. Example of fuzzy logic rules (VH—very high, H—high, M—medium, L—low).

| Predicted Power | Solar Irradiance | Wind Speeds | Temperature | Batteries Voltage | Inverter Current Level | Level |
|-----------------|------------------|-------------|-------------|-------------------|-----------------------|-------|
| VH              | VH               | any         | any         | any               | any                   | 4     |
| H               | H                | any         | any         | H                 | any                   | 4     |
| VH              | any              | any         | any         | any               | any                   | 4     |
| H               | H                | L           | any         | M                 | any                   | 4     |
| H               | M                | any         | L           | H                 | any                   | 3     |
| M               | M                | any         | H           | H                 | any                   | 3     |
| M               | M                | any         | M           | L                 | any                   | 1     |
| M               | M                | any         | M           | M                 | any                   | 1     |
| L               | L                | any         | any         | H                 | L                     | 2     |
| L               | L                | VH          | any         | M                 | any                   | 2     |
| L               | L                | any         | any         | any               | M                     | 1     |
| L               | L                | any         | any         | any               | L                     | 1     |

Thus, the fuzzy logic algorithms implemented in R language allow, based on the existence of a large number of input and output values, the recreation of the fuzzy logic rules which are subsequently used to indicate the power level. The concept of SASM is shown in Figure 16.

![Figure 16. Simplified block diagram for SASM.](image-url)

In this diagram, the Forecasted power parameter is separated from the other input parameters to delimit it as a derived parameter obtained from PFM. Historical data records containing the Input parameters, Forecasted power and Power levels are stored as Data store records. Storing can be performed on a disk file, a database or a HDFS (Hadoop File System). The stored data is mixed and divided into two groups: training data and test data to be fetched by the R Machine Learning & fuzzy logic.
Once trained and tested, the fuzzy logic algorithm is used for prediction. The real-time dataset called *Data to predict* is also used by fuzzy logic as an input to predict the power level that will be sent to Logic switch, according to which appliances are connected/disconnected.

3. Case Study

3.1. RES System Depiction

The tests have been carried out on a system located at 70 km northwest of Bucharest (Romania), in Hulubești (Dâmbovița) containing the following equipment:

1. **PV panels:**
   - 4 × 24 V 245 W polycrystalline, Photowatt PW2450F
   - 4 × 24 V 250 W polycrystalline, Photowatt PW2350F
   - 4 × 24 V 260 W polycrystalline, Kingdom Solar KD-P260W
   - 4 × 24 V 300 W monocrystalline, Q.ANTUM 300W

2. 4 × MPPT solar controller, EPEVER Tracer3210

3. 2 × Wind Turbines 200 W 24 V VAWT and FW12/24 wind controller

4. One Power Jack Inverter LFPSW-5000-24-240 (5000 W off grid)

5. 4 × battery Sorgetti Deep Cycle 200 Ah, 12 V

At each MPPT Tracer3210, 4 panels are connected, grouped in 2 series, connected in parallel to the controller. MPPT receives as maximum input voltage 2 × “Open circuit voltage” PV panel, i.e., approximately 2 × 37.2 V = 74.4 V under the maximum allowable voltage of the controller (100 V).

The PV panels have a total power output of 4220 W, with average solar irradiation of 1000 W/m² (under real conditions, the power may be less or sometimes even higher depending on various factors such as solar light, cloudy sky, season, etc.).

For the solar irradiance measurement, a Tenmars TM-207 Solar Power Meter is utilized. It is relatively a cheap sensor, but usually it is advisable to have performant sensors. Calibrating and checking the sensor parameters before acquisition are also significant. The specifications of this meter are:

- Display: 3-1/2 digits, 2000 counts;
- Range: 2000 W/m², 634 BTU/(ft² × h);
- Resolution: 0.1 W/m², 0.1 BTU/(ft² × h);
- Accuracy: Typically, within ±10 W/m² [±3 BTU/ (ft² × h)] or ±5% whichever is greater in the sunlight;
- Temperature included error ± 0.38 W/m²/°C [±0.12 BTU/(ft² × h)]/°C] deviation from 25 °C;
- Angular Accuracy: Cosine corrected;
- Drift: less than ±2% per year;
- Overload: Display ‘OL’;
- Sampling Rate: 0.25/s.

It results that the accuracy is ±10 W/m² for a PV panel of 300 W with the dimensions around 1.0 m × 1.6 m, giving an accuracy of 10 × 1.6 per panel = 16 W(panel). Then, for such a PV panel, the tolerance can be of 16/300 = 5.3% for the solar irradiance. Since the PV system has a relatively small number of panels (16), spread on a small area, using a single solar irradiance measurement sensor is enough. For larger PV systems, it is necessary to install more sensors, especially if the system is spread on a large area, it requires multiple solar irradiation measurements.

Another important challenge is to acquire weather parameters. One way to acquire them is to use a weather station installed on site. However, a more accessible and cheaper solution is utilized;
building a Java program that accesses Accuweather API. The servers of Accuweather are accessed via an API with a frequency of several minutes, using http requests. The server responds based on parameters (such as wind speed, outside temperature and cloud cover) collected from a weather station that is near the PV system location. The Accuweather server responds in the .json format, that is converted in a Java program and written in a .csv file. By using APIs from the Accuweather servers, there can be differences between the server parameters (wind speed, outside temperature, cloud cover) and the real parameters at the location. However, these differences are not that relevant as the Accuweather weather station from which the data is collected is close to the location of the PV system (usually at a distance of a few kilometres). The sensors for current and voltage are regular sensors, that do not raise issues regarding the tolerance. Even if there is some tolerance for sensors of the input parameters, as we will notice in simulations, the model is validated by the results.

3.2. Simulations

In order to highlight the practical operation of SASM, the tendency is to perform simulations for different times of the year. Specifically, depending on the season and the location of the system, within a day, the power generated by PV is variable. In mid-February, for example, PV generates between 09:00 a.m. and 16:00 a.m., while in mid-April, PV generates between 08:30 a.m. and 17:00 a.m. Considering that the primary purpose of SASM is to switch appliances adaptively according to the input parameters (such as weather conditions and battery charging status), testing on different days is not very relevant, but it is particularly useful to test SASM for a single day with fluctuations of the power forecast, weather conditions and even consumption. To comply with these requirements, we chose 14 February 2019.

Thus, the input file contains the parameters for SASM, with the LoadL column, that will be populated after running the fuzzy logic algorithm. The input data are collected at 15-min intervals between 09:00 and 16:00 local time, when the PV panels can generate significant energy as in Figure 17.

| Forecasted Solar | Wind | Outside | BatteryV | Inverter | LoadL | LocalObservationDateTime | CloudCover |
|------------------|------|---------|----------|----------|-------|---------------------------|------------|
| 426              | 108  | 7.3     | -0.8     | 31.2     | 4     | 2019-02-14T09:00:00+02:00 | 76         |
| 1457             | 175  | 6.4     | -0.6     | 31.3     | 3.6   | 2019-02-14T09:15:00+02:00 | 21         |
| 1486             | 177  | 6.4     | -0.6     | 24.9     | 4     | 2019-02-14T09:30:00+02:00 | 26         |
| 1902             | 502  | 7.4     | 0.4      | 21.5     | 7     | 2019-02-14T09:45:00+02:00 | 7          |
| 2093             | 530  | 7.4     | 0.4      | 24.5     | 10    | 2019-02-14T10:00:00+02:00 | 7          |
| 2014             | 510  | 3.7     | 6.1      | 26.0     | 16.8  | 2019-02-14T10:15:00+02:00 | Will       |
| 2124             | 535  | 3.7     | 6.1      | 31.8     | 11.4  | 2019-02-14T10:30:00+02:00 | 15         |
| 2476             | 627  | 3.7     | 6.1      | 27.2     | 16    | 2019-02-14T10:45:00+02:00 | 9          |
| 2844             | 720  | 3.7     | 6.1      | 28.3     | 17    | 2019-02-14T11:00:00+02:00 | 4          |
| 2806             | 709  | 3.7     | 7.8      | 29.4     | 10    | 2019-02-14T11:15:00+02:00 | be         |
| 3005             | 761  | 7.4     | 7.4      | 28.0     | 9.6   | 2019-02-14T11:30:00+02:00 | 6          |
| 3058             | 623  | 7.4     | 7.4      | 29.1     | 4.7   | 2019-02-14T11:30:00+02:00 | 2          |
| 3100             | 600  | 7.4     | 7.4      | 25.9     | 16.3  | 2019-02-14T11:45:00+02:00 | 8          |
| 3092             | 783  | 7.4     | 8.8      | 28.9     | 9.5   | 2019-02-14T12:00:00+02:00 | Predicted  |
| 3135             | 789  | 7.4     | 8.8      | 28.7     | 8.7   | 2019-02-14T12:15:00+02:00 | 13         |
| 2512             | 656  | 11.1    | 7.9      | 27.9     | 8.9   | 2019-02-14T12:30:00+02:00 | 43         |
| 501              | 200  | 11.1    | 7.9      | 28.5     | 1.5   | 2019-02-14T13:00:00+02:00 | 76         |
| 796              | 194  | 7.4     | 7        | 25.2     | 3.4   | 2019-02-14T13:15:00+02:00 | 78         |
| 501              | 127  | 7.4     | 7        | 25.2     | 2     | 2019-02-14T13:30:00+02:00 | 83         |
| 367              | 93   | 7.4     | 7        | 24.8     | 2.8   | 2019-02-14T13:45:00+02:00 | 87         |
| 406              | 103  | 7.4     | 8.2      | 24.1     | 4.5   | 2019-02-14T14:00:00+02:00 | 65         |
| 170              | 44   | 11.1    | 6.7      | 23.9     | 2.1   | 2019-02-14T14:15:00+02:00 | 93         |
| 47               | 12   | 11.1    | 6.7      | 23.8     | 3.8   | 2019-02-14T14:30:00+02:00 | 98         |
| 191              | 48   | 11.1    | 6.7      | 23.9     | 1.9   | 2019-02-14T14:45:00+02:00 | 91         |
| 483              | 122  | 4.7     | 7.1      | 23.9     | 2.8   | 2019-02-15T00:00:00+02:00 | 76         |
| 455              | 152  | 7.4     | 7.1      | 23.6     | 2.3   | 2019-02-15T00:15:00+02:00 | 76         |
| 331              | 84   | 7.4     | 7.1      | 23.5     | 2     | 2019-02-15T00:30:00+02:00 | 80         |
| 211              | 58   | 7.4     | 7.1      | 23.1     | 2.1   | 2019-02-15T00:45:00+02:00 | 85         |
| 165              | 42   | 7.4     | 7.6      | 23       | 2     | 2019-02-15T01:00:00+02:00 | 86         |

Figure 17. Data sample used in simulation.

As it can be seen in this dataset, the cloud coverage of sky has significant variations from 76% at 09:00 to 7% at 10:00. Around 12:00, it grows from 8% to 33% then, in the afternoon, it goes over 75%.
This sequence has a significant variation of the input weather parameters corresponding to the test requirements. Note that a day with a full clear sky or completely sunny is not relevant or does not test enough the SASM because such a day has less variable weather parameters.

To predict consumption levels for February 14, we have taken the data from the input file (datFzy14Feb.xlsx) using the following R code:

```r
dfzy1 <- read.xlsx("datFzy14Feb.xlsx", sheetIndex = 1)
and we considered it as a test dataset redefining the tst.dfzymix test dataset as follows:
tst.dfzymix <- dfzy1[1:nrow(dfzy1),1:6]
```

For simulations, FH.GBML is implemented. It is an Ishibuchi fuzzy logic method based on the hybrid cooperative Genetic Compilation-Competitive Learning (GCCL) algorithm and the Pittsburgh algorithm (FH.GBML). Some of the control parameters used to call the method are genetic algorithm-specific parameters: maximum population size, `popu.size`; the maximum number of generations for the genetic algorithm, `max.gen`; the probability of the GCCL process, `p.gccl`; the maximum number of generations for the genetic algorithm, `max.gen`; the probability of a mutation, `persen_mutant`; that is also a number between 0 and 1; the maximum number of generations for the genetic algorithm, `max.gen`; the probability of the GCCL process, `p.gccl`.

The following control parameters are related to the fuzzy method: the maximum number of classes, `num.class`; the maximum number of fuzzy logic rules that will be generated `max.num.rule`.

The R code used in one of the simulations are given in Figure 18.

```r
# load records
dfzy <- read.xlsx("datFzy.xlsx", sheetIndex = 1)
# max records
dfzymix <- dfzy[sample(nrow(dfzy)),]
dfzymix[,7] <- unclass(dfzymix[,7])
# creating training set
trn.dfzymix <- dfzymix[1:146,]
# creating test set
tst.dfzymix <- dfzymix[147:nrow(dfzymix),1:6]
tru.dfzymix <- matrix(dfzymix[147:nrow(dfzymix),7], ncol = 1)
# calculate range
range.dfzymix <- matrix(apply(dfzymix[, -ncol(dfzymix)], 2, range), nrow = 2)
# Ishibuchi\'s method based on hybridization of genetic cooperative-competitive
# learning (GCCL) and Pittsburgh (FH.GBML).
method.type <- "FH.GBML"
classification <- list(popu.size = 50, max.num.rule = 10, num.class = 3, persen_cross = 0.9,
max.gen = 10, persen_mutant = 0.5, p.doare = 0.5, p.gccl = 1, name = "sin-0")
# apply fuzzy logic FH.GBML model
frbs.obj <- frbs.learn(trn.dfzymix, range.dfzymix, method.type, control)
# predict (classify remaining records)
predict.dfzymix <- predict(frbs.obj, tst.dfzymix)
# display results
print(predict.dfzymix)
print(trn.dfzymix)
print(tst.dfzymix)
```

Figure 18. R code for simulations.

Results of the simulation should be analysed considering the predicted LoadL (`predict.dfzymix`) and the input values for which the prediction is performed (`tst.dfzymix`), as shown in Figure 19.

SASM predicts the load level (LoadL) based on a fuzzy logic algorithm, the error `Err` being calculated with following equation:

\[
Err = \frac{\text{Number of problematic predictions}}{\text{Total number of predictions}} \times 100\%
\] (11)

In this equation, “number of problematic predictions” is used (not the “number of wrong predictions”) as the process is fuzzy. LoadL = 1 prediction that should be LoadL = 2 is not an error,
or less correct. LoadL = 1 prediction is still useful as it protects the battery banks from discharging beyond prespecified level. However, it can also be disadvantageous because it does not optimally use the local available renewable energy.

By analysing the LoadL column and comparing it with the mapping rules mentioned above, only in case (I), row 18, the prediction is problematic because the level 1 (LoadL = 1) is connected although the battery has 26 V, ForcastedP = 821 W/m² and the wind speed is WindS = 11.1 m/s. Thus, according to Equation (11), the error is 1/29 = 3.4% which is a fair result. The error rate could be even lower as we use more records in the test set.

In case (II), at the beginning of the day when the battery bank is still unloaded, LoadL = 1 despite the fact that the power forecast starts to increase.

In case (III), the battery bank is loaded (23.5 V, 24.5 V, 26 V), the power forecast that is at medium to high level cumulated with an average current of the inverter allow coupling the power level 2 or even 3 (case (IV)) over a period of time.

In case (V), the SASM detects the high and very high power, the battery bank full or almost full, so, practically, regardless of other parameters (whatever the inverter current or consumption at that moment) it engages power level 4.

In case (VI), rows 19, 20, 21, 22, register a decrease of the power forecast, one of the reasons being the increase of the cloud cover. Thus, the SASM switches to power levels 1 and 2. In this case, the current of inverter at rows 19 and 22 varies between 3.4 A and 4.5 A, thus level 1 or 2 is chosen.

(VII) is the case in which the system is stabilized at a low power level, LoadL = 2, the consumption being small (the inverter current is approximately 2 A) and still having power predictions between 200 W and 300 W, that makes power level 2 acceptable. In maintaining the power levels, the wind turbine VAWT contributes to a small extent, especially for rows 17, 23, 24, 25, where the wind speed is 11.1 km/h.

These discussions demonstrate the following important features of the proposed SASM:

- Sustainability through the efficient use of available RES generation, switching power levels, and additionally protecting the battery bank;
- Adaptability by switching the load according to the available power.

|ForecastedP| Solar| WindS| Outside| BatteryV| Inverter| LoadL| localObservation|Date|Time|CloudCover|
|---|---|---|---|---|---|---|---|---|---|---|
|2420|105.7|3|0.8|23.2|4|1|200-2-14T09:00:00+00:00|200|| |
|1497|375.6|4|3.8|23.3|4|1|200-2-14T09:15:00+00:00|230|| |
|1429|377.6|4|3.8|24|4|1|200-2-14T09:30:00+00:00|200|| |
|1905|505.7|4|3.8|23.3|7|3|200-2-14T09:45:00+00:00|200|| |
|2903|530.7|4|3.8|24.5|10|2|200-2-14T10:00:00+00:00|200|| |
|2014|510.7|4|3.8|25|10|2|200-2-14T10:15:00+00:00|200|| |
|2115|520.7|4|3.8|26.5|10|2|200-2-14T10:30:00+00:00|200|| |
|2976|550.7|4|3.8|27.2|16|4|200-2-14T10:45:00+00:00|200|| |
|2844|720.7|4|3.8|28.3|17|4|200-2-14T11:00:00+00:00|200|| |
|2800|709.7|4|3.8|29.4|10|4|200-2-14T11:15:00+00:00|200|| |
|3005|761.7|4|3.8|28.7|9|4|200-2-14T11:30:00+00:00|200|| |
|3250|823.7|4|3.8|29.1|4|4|200-2-14T11:45:00+00:00|200|| |
|3160|800.7|4|3.8|29.5|16|4|200-2-14T12:00:00+00:00|200|| |
|3592|783.7|4|3.8|28.9|9.5|4|200-2-14T12:15:00+00:00|200|| |
|3195|805.7|4|3.8|28|8|2|200-2-14T12:30:00+00:00|200|| |
|2512|639.11|7.9|27.9|8.9|4|200-2-14T12:45:00+00:00|200| | |
|821|208.11|7.9|26|5|1|200-2-14T13:00:00+00:00|200| | |
|706|194.7|7|25.2|3.4|1|200-2-14T13:15:00+00:00|200| | |
|501|137.7|7|25.2|2.2|2|200-2-14T13:30:00+00:00|200| | |
|367|93.7|7|24.8|2.8|2|200-2-14T13:45:00+00:00|200| | |
|406|103.7|7|24.1|4.5|2|200-2-14T14:00:00+00:00|200| | |
|175|44.11|6.7|23.9|2|2|200-2-14T14:15:00+00:00|200| | |
|47|12.11|6.7|23.8|1.8|2|200-2-14T14:30:00+00:00|200| | |
|191|48.6|11.1|6.7|23.9|1.9|2|200-2-14T14:45:00+00:00|200| | |
|463|122.7|4.7|23.9|2.8|2|200-2-14T15:00:00+00:00|200| | |
|455|115.2|7.1|23.6|2.3|2|200-2-14T15:15:00+00:00|200| | |
|331|84.7|7|23.5|2.2|2|200-2-14T15:30:00+00:00|200| | |
|231|58.5|7.1|23.1|2.1|2|200-2-14T15:45:00+00:00|200| | |
|185|42.7|7.6|23|2|2|200-2-14T16:00:00+00:00|200| | |

Figure 19. The results of the simulation.
Regarding the adaptive feature, SASM reacts to weather and system parameters and continuously adapts the power level. So, in case the sky is less cloudy, the system can generate more power, charges the batteries and thus the SASM switches to a higher power level. However, if the sun is covered by clouds, based on the power forecast, the SASM switches to a lower power level.

For testing and tuning the SASM, the simulation is extended to another data set containing 156 records of which 146 are used for training and 10 for testing (as in Figure 20) and the model is re-run.

The predictions are tolerable except for record 155. Thus, for the selected record 51 that has the power forecast \( \text{ForecastedP} = 2610 \, \text{W} \), the solar irradiance intensity \( \text{SolarI} = 627 \, \text{W/m} \) (a bright sun that will generate abundant PV energy); wind speed \( \text{WindS} = 22.8 \, \text{m/s} \) (wind energy will also be generated); external temperature \( \text{OutsideT} = -17.3 \, ^\circ\text{C} \), battery voltage \( \text{BatteryV} = 26.5 \, \text{V} \) (which means that there is already an average to full charging level), inverter current \( \text{InverterC} = 19.1 \, \text{A} \); under these conditions, the power level is 4 as in Figure 21.

![Figure 20. Additional records used in simulations.](image)

![Figure 21. The results for simulation using FH.GBML.](image)
A particular case is record 155 as mentioned above: for low solar irradiance, low wind speed, low power forecast, it wrongly indicates power level 4. It is the only unacceptable result, so we can say that the error rate is 10%. This error can probably be explained by the percentage of substantial mutations persen\_mutant = 0.9 (allowed values are between 0–1). Another cause may be the small number of records used in the simulation.

3.3. Fuzzy Model and Rules Used in Simulations

Figure 22 partially reproduces the autogenerated fuzzy model based on which the prediction is performed.

Figure 22. Extract from the autogenerated fuzzy model.

A rule (as marked in yellow in Figure 22) consists in two lines. Thus, the rule [1,] is:

“IF” “ForcastedP” “is” “v.1\_a.9” “and” “SolarI” “is” “v.2\_a.7” “and” “WindS” “is” “v.3\_a.4” “and” “OutsideT” “is” “v.4\_a.13” “and” “BatteryV” “is” “v.5\_a.13” “and” “InverterC” “is” “v.6\_a.14” “THEN” “LoadL” “is” “4”

Similarly, the rule [2,] is:

“IF” “ForcastedP” “is” “v.1\_a.8” “and” “SolarI” “is” “v.2\_a.12” “and” “WindS” “is” “v.3\_a.4” “and” “OutsideT” “is” “v.4\_a.2” “and” “BatteryV” “is” “v.5\_a.7” “and” “InverterC” “is” “dont\_care” “THEN” “LoadL” “is” “3”

Also, the rule [3,] is:

“IF” “ForcastedP” “is” “v.1\_a.8” “and” “SolarI” “is” “v.2\_a.2” “and” “WindS” “is” “v.3\_a.7” “and” “OutsideT” “is” “v.4\_a.1” “and” “BatteryV” “is” “v.5\_a.10” “and” “InverterC” “is” “v.6\_a.6” “THEN” “LoadL” “is” “4” and so on.

Therefore, the autogenerated fuzzy rules are rigorously mathematically defined as logical expressions that fit into the practical rules outlined in Figure 22.

It is useful to note that the model cannot be modified directly in the sense that the fuzzy rules are self-generated based on the training data set. However, the model can be significantly influenced by the control parameters of the learning function, frbs.learn. For some control parameters of the fuzzy logic...
algorithms implemented in R language, the processing time and the computational power requirements can be considerable. Cheaper computer equipment has lately evolved; thus, this limitation would not be an issue. However, if we need to obtain the solution on simplified platforms like Raspberry PI, this will be harder to achieve.

Finding the convenient parameters for the fuzzy logic algorithms implemented in R may require a considerable number of tests. For instance, for a single model, over 75 tests have been performed in this paper. Given that R has several such algorithms, if we have to select the most adequate algorithm, more than 500 tests can be required, which means a long time. Regarding the previous aspects, this potential limitation can be technically bypassed by creating a supplementary code to automatically test various scenarios. The code creation effort is justified because it speeds up the testing process and permits a faster identification of the model and better control parameters.

4. Conclusions

In this paper, a fuzzy logic based on SASM architecture is proposed. It controls the consumption in RES systems switching the appliances, according to their power level, to maximize the usage of the RES potential. Thus, the SASM efficiently allows the usage of the local available RES by automatically connecting the power levels in an adaptive way according to the input parameters. It protects the battery bank by keeping the charging level above 60%. Also, SASM decreases the costs of a RES system since it allows a smaller capacity of the battery bank. By maximizing the consumption from the available RES, we can empirically say that SASM allows a faster ROI. Through the benefits it brings by self-learning and affordable deployment in solving a complex problem of improving RES integration, the SASM architecture proposed in this paper proved to be useful in practice.

Author Contributions: All authors have equal contribution. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Acknowledgments: This paper presents the scientific results of the project “Intelligent system for trading on wholesale electricity market” (SMARTTRADE), co-financed by the European Regional Development Fund (ERDF), through the Competitiveness Operational Programme (COP) 2014–2020, priority axis 1—Research, technological development and innovation (RD&I) to support economic competitiveness and business development, Action 1.1.4-Attracting high-level personnel from abroad in order to enhance the RD capacity, contract ID P_37_418, no. 62/05.09.2016, beneficiary: The Bucharest University of Economic Studies.

Conflicts of Interest: The authors declare no conflict of interest.

References
1. Teimourzadeh, S.; Tor, O.B.; Cebeci, M.E.; Bara, A.; Oprea, S.V. A three-stage approach for resilience-constrained scheduling of networked microgrids. J. Mod. Power Syst. Clean Energy 2019. [CrossRef]
2. Cagnano, A.; De Tuglie, E.; Mancarella, P. Microgrids: Overview and guidelines for practical implementations and operation. Appl. Energy 2020, 258, 114039. [CrossRef]
3. Hossain, M.A.; Pota, H.R.; Hossain, M.J.; Blaabjerg, F. Evolution of microgrids with converter-interfaced generations: Challenges and opportunities. Int. J. Electr. Power Energy Syst. 2019, 109, 160–186. [CrossRef]
4. Glinkowski, M.; Guglielmo, A.; Oudalov, A.; Rackliffe, G.; Rose, B.; Scholtz, E.; Verna, L.; Yang, F. Microgrids. Smart Grids: Clouds, Communications, Open Source, and Automation; CRC Press: Boca Raton, FL, USA, 2017; ISBN 9781482206128.
5. Borlase, S.; Ganji, M.; Shahidehpour, M.; Tian, W.; Burgess, P.M. Smart Grids: Advanced Technologies and Solutions, Second Edition; CRC Press: Boca Raton, FL, USA, 2017; ISBN 9781498799560.
6. Kals, J. ISO 50001 Energy Management Systems; Business Expert Press: New York, NY, USA, 2015; ISBN 9781631570100.
7. Lobaccaro, G.; Carlucci, S.; Löfström, E. A review of systems and technologies for smart homes and smart grids. Energies 2016, 9, 348. [CrossRef]
8. Pavithra, D.; Balakrishnan, R. IoT based monitoring and control system for home automation. In Proceedings of the Global Conference on Communication Technologies, GCCT 2015, Thuckalay, India, 23–24 April 2015.

9. Khan, M.; Silva, B.N.; Han, K. Internet of Things Based Energy Aware Smart Home Control System. IEEE Access 2016. [CrossRef]

10. Schieweck, A.; Uhde, E.; Salthammer, T.; Salthammer, L.C.; Morawska, L.; Mazaheri, M.; Kumar, P. Smart homes and the control of indoor air quality. Renew. Sustain. Energy Rev. 2018, 94, 705–718. [CrossRef]

11. Zhao, H.; Wu, Q.; Wang, C.; Cheng, L.; Rasmussen, C.N. Fuzzy logic based coordinated control of battery energy storage system and dispatchable distributed generation for microgrid. J. Mod. Power Syst. Clean Energy 2015. [CrossRef]

12. Yap, K.Y.; Sarimuthu, C.R.; Lim, J.M.Y. Grid Integration of Solar Photovoltaic System Using Machine Learning-Based Virtual Inertia Synthetization in Synchronverter. IEEE Access 2020. [CrossRef]

13. Ashraf, M.M.; Waqas, A.B.; Malik, T.N. Grid connected wind energy conversion system for peak load sharing using fuzzy logic controller. Int. J. Renew. Energy Res. 2017, 7, 1767–1778.

14. Sushma Devi, P.; Venugopala Rao, M. Design of energy management system for residential grid-connected microgrid with HRES. Int. J. Control, Theory Appl. 2016, 5, 131–139.

15. Goldade, J.; Salehfar, H.; Laframboise, K.; LaBlanc, J.; Mann, M. The use of a fuzzy logic algorithm to integrate renewable energy sources. In Proceedings of the NAPS 2011—43rd North American Power Symposium, Boston, MA, USA, 4–6 August 2011.

16. Garrab, A.; Bouallegue, A.; Bouallegue, R. An agent based fuzzy control for smart home energy management in smart grid environment. Int. J. Renew. Energy Res. 2017, 7, 599–612.

17. El Bourakadi, D.; Yahyaoui, A.; Bounhidi, J. Multi-Agent System Based on the Extreme Learning Machine and Fuzzy Control for Intelligent Energy Management in Microgrid. J. Intell. Syst. 2020. [CrossRef]

18. Chekired, F.; Mahran, A.; Samara, Z.; Chikh, M.; Guenounou, A.; Meftah, A. Fuzzy logic energy management for a photovoltaic solar home. Energy Procedia 2017, 134, 723–730. [CrossRef]

19. Ross, T.J. Fuzzy Logic. with Engineering Applications, 4th ed.; Wiley: New York, NY, USA, 2016; ISBN 9781119235866.

20. Palacios, J. Unity 2018 Artificial Intelligence Cookbook: Over 90 Recipes to Build and Customize AI Entities for Your Games with Unity, 2nd ed.; Packt Publishing: Birmingham, UK, 2018; ISBN 978-1788626170.

21. Al-Sakkaf, S.; Kassas, M.; Khalid, M.; Abido, M.A. An Energy Management System for Residential Autonomous DC Microgrid Using Optimized Fuzzy Logic Controller Considering Economic Dispatch. Energies 2019. [CrossRef]

22. Aghajani, G.R.; Shayanfar, H.A.; Shayeghi, H. Demand side management in a smart micro-grid in the presence of renewable generation and demand response. Energy 2017. [CrossRef]

23. Apajalahti, E.L.; Lovio, R.; Heiskanen, E. From demand side management (DSM) to energy efficiency services: A Finnish case study. Energy Policy 2015. [CrossRef]

24. Oprea, S.V.; Bara, A. Machine Learning Algorithms for Short-Term Load Forecast in Residential Buildings Using Smart Meters, Sensors and Big Data Solutions. IEEE Access 2019. [CrossRef]

25. Wikipedia Intelligent Hybrid Inverter. Available online: https://en.wikipedia.org/wiki/Intelligent_hybrid_inverter#cite_note-3 (accessed on 21 July 2020).

26.IMEON Energy SOLAR HYBRID INVERTER Installation Guide. Available online: https://imeon-energy.com/wp-content/uploads/Installation_GuideIMEON_HV.pdf (accessed on 21 July 2020).

27. Nazmul Siddique, H.A. Computational Intelligence: Synergies of Fuzzy Logic, Neural Networks and Evolutionary Computing; Wiley: New York, NY, USA, 2013; ISBN 9781118337844.

28. Ianuc, I. Fuzzy Logic—Controls, Concepts, Theories and Applications; A Mamdani Type Fuzzy Logic Controller Chapter; IntechOpen: London, UK, 2012.

29. Norris, D.J. Beginning Artificial Intelligence with the Raspberry Pi; Apress Barrington: Barrington, NH, USA, 2017.
32. Popescu, D.; Gharbi, A.; Stefanoiu, D.; Borne, P. Process. Control. Design for Industrial Applications; ISTE Ltd.: London, UK, 2017.
33. Ross, T.J. Fuzzy Logic. with Engineering Applications, 3rd ed.; Wiley: New York, NY, USA, 2010; ISBN 9780470743768.
34. Control Theory. Available online: https://en.wikipedia.org/wiki/Control_theory (accessed on 26 May 2020).
35. Spatial Decision Support Systems/Fuzzy Controller/Fuzzy Sets. Available online: https://en.wikiversity.org/wiki/Spatial_Decision_Support_Systems/Fuzzy_Controller/Fuzzy_Sets (accessed on 26 May 2020).
36. Lilly, J.H. Fuzzy Control. and Identification; John Wiley & Sons: New York, NY, USA, 2010; ISBN 9780470542774.
37. Jang, J.R.; Sun, C.; Mizutani, E. Neuro-Fuzzy and Soft Computing Chapter 4 Fuzzy Inference Systems; Prentice Hall: Upper Sadle River, NJ, USA, 1997.
38. Fuzzy Control System. Available online: https://en.wikipedia.org/wiki/Fuzzy_control_system (accessed on 25 May 2020).
39. Takagi, T.; Sugeno, M. Fuzzy Identification of Systems and Its Applications to Modeling and Control. IEEE Trans. Syst. Man Cybern. 1985. [CrossRef]
40. Kaur, A.; Kaur, A. Comparison of Mamdani-Type and Sugeno-Type Fuzzy Inference Systems for Air Conditioning System. Int. J. Soft Comput. Eng. 2012, 2, 323–325.
41. Roy, S.; Chakraborty, U. Soft Computing; Pearson India: New Delhi, India, 2013; ISBN 9789332514201.
42. Preda, S.; Oprea, S.V.; Băra, A.; Belciu, A. PV forecasting using support vector machine learning in a big data analytics context. Symmetry 2018. [CrossRef]
43. Kleissl, J. Solar Energy Forecasting and Resource Assessment; Academic Press: New York, NY, USA, 2013; ISBN 9780123971777.
44. Oprea, S.V.; Băra, A. Ultra-short-term forecasting for photovoltaic power plants and real-time key performance indicators analysis with big data solutions. Two case studies—PV Agigea and PV Giurgiu located in Romania. Comput. Ind. 2020. [CrossRef]
45. Pelland, S.; Remund, J.; Kleissl, J.; Oozeki, T.; Brabandere, K. De Photovoltaic and Solar Forecasting: State of the Art. Int. Energy Agency Photovolt. Power Syst. Program. Rep. IEA PVPS T14 2013, 14, 1–36.
46. IEC International Electrotechnical Commission Standards. Available online: https://webstore.iec.ch/publication3849 (accessed on 18 November 2019).
47. Fthenakis, V.M.; Lynn, P.A. Electricity from Sunlight: Photovoltaic-Systems Integration and Sustainability; Wiley: New York, NY, USA, 2018; ISBN 9781118963784.
48. Ferdjallah, M. Introduction to Digital Systems Modeling, Synthesis, and Simulation Using VHDL; John Wiley & Sons: New York, NY, USA, 2013; ISBN 9780470900550.
49. Paper, C. A Fuzzy based Low-Cost Monitoring Module built with Raspberry Pi-Python-Java Architecture. In Proceedings of the 2015 International Conference in Smart Sensors and Application (ICSSA), Kuala Lumpur, Malaysia, 26–28 May 2015.
50. Cameron, N. Arduino Applied: Comprehensive Projects for Everyday Electronics; Apress: New York, NY, USA, 2018; ISBN 9781484239599.
51. Seneviratne, P. Internet of Things with Arduino Blueprints; Community experience distilled; Packt Publishing: Birmingham, UK, 2015; ISBN 9781785285486.
52. Schwartz, M. Building Smart Homes with Raspberry Pi Zero; Packt Publishing: Birmingham, UK, 2016; ISBN 9781786469861.
© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).