



HAL
open science

An Automated Energy Management Framework for Smart Homes

Houssam Kanso, Adel Nouredine, Ernesto Exposito

► **To cite this version:**

Houssam Kanso, Adel Nouredine, Ernesto Exposito. An Automated Energy Management Framework for Smart Homes. JAISE - Journal of Ambient Intelligence and Smart Environments, 2023, 16 (1), pp.23-42. 10.3233/AIS-220482 . hal-04396046

HAL Id: hal-04396046

<https://hal.science/hal-04396046v1>

Submitted on 15 Oct 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

An Automated Energy Management Framework for Smart Homes

Houssam Kanso^{a,*}, Adel Noureddine^a and Ernesto Exposito^a

^a *Université de Pau et des Pays de l'Adour, E2S UPPA, LIUPPA, Anglet, France*

E-mails: houssam.kanso@univ-pau.fr, adel.noureddine@univ-pau.fr, ernesto.exposito@univ-pau.fr

Abstract.

Over the last fifty years, societies across the world have experienced multiple periods of energy insufficiency with the most recent one being the 2022 global energy crisis. In addition, the electric power industry has been experiencing a steady increase in electricity consumption since the second industrial revolution because of the widespread usage of electrical appliances and devices. Newer devices are equipped with sensors and actuators, they can collect a large amount of data that could help in power management. However, current energy management approaches are mostly applied to limited types of devices in specific domains and are difficult to implement in other scenarios. They fail when it comes to their level of autonomy, flexibility, and genericity. To address these shortcomings, we present, in this paper, an automated energy management approach for connected environments based on generating power estimation models, representing a formal description of energy-related knowledge, and using reinforcement learning (RL) techniques to accomplish energy-efficient actions. The architecture of this approach is based on three main components: power estimation models, knowledge base, and intelligence module. Furthermore, we develop algorithms that exploit knowledge from both the power estimator and the ontology, to generate the corresponding RL agent and environment. We also present different reward functions based on user preferences and power consumption. We illustrate our proposal in the smart home domain. An implementation of the approach is developed and two validation experiments are conducted. Both case studies are deployed in the context of smart homes: (a) a living room with a variety of devices and (b) a smart home with a heating system. The obtained results show that our approach performs well given the low convergence period, the high level of user preferences satisfaction, and the significant decrease in energy consumption.

Keywords: Energy Management, Smart Home, Contextual Knowledge, Artificial Intelligence, Automated Model Generation

1. Introduction

In the past few years, the number of connected devices has significantly increased and will continue in the future, in particular, for Internet of Things (IoT) and smart home devices [1]. Cisco estimates that 500 billion devices will be connected to the internet by 2030 [2]. With this increase, the energy consumption of these devices is growing and may challenge the capacity of energy production. For instance, in 2019, over 25 000 TWh of final energy consumption was electrical [3]. The increasing energy needed for these devices results in a high carbon footprint, therefore, a negative impact on the environment. Moreover, the impact of the COVID-19 pandemic and the Russo-Ukrainian war on energy has demonstrated the need for developing adaptive strategies for energy management efficiency and sustainability. In recent years, more sustainable and renewable energy is used, such as wind, solar,

*Corresponding author. E-mail: houssam.kanso@univ-pau.fr.

1 and ocean energies. These sources lead to greener energy production, however, energy leakages and losses continue
2 to occur on the demand side.

3 On the demand side, industrial and residential sectors consume the majority of final electrical energy [3]. In
4 residential sectors, conventional electrical systems and devices are not very eco-friendly. Some are considered high
5 energy demanding (e.g., water heater, fridge, and laundry machine), and others are smaller devices and do not
6 consume as much (e.g., internet modem, led lamp, and mobile phone). However, their large number leads to high
7 energy consumption of the entire environment. Therefore, optimizing a large number of small devices can have a
8 big impact [4]. A saving of 1.0 Mega-Watt per hour worldwide by saving 0.25 W per device for 4 million users and
9 these numbers increase as we go to larger scales, as estimated in [5].

10 The introduction of IoT and smart devices has led to the emergence of the concept of a smart home. Many IoT
11 devices now offer built-in intelligence capabilities that enables them to communicate with each other and connect to
12 the internet. These intelligent devices help automate tasks, reduce cost, and save energy. A large number of devices
13 integrate sensors allowing them to produce data describing the state of a device or the behavior of a person. In
14 addition, a large number of these devices are also equipped with actuators. Such devices can accomplish tasks that
15 directly impact the environment where they belong. These elements could allow the implementation of autonomous
16 feedback control loops for power management allowing easy integration and understanding between devices to
17 achieve higher energy savings.

18
19 In smart homes, several heterogeneous devices are connected using different protocols and sharing all kinds of
20 data. They interact with each other to accomplish specific tasks, in addition to their interactions with humans which
21 adds an extra level of complexity to the system. The smart home environments are scalable as the number of devices
22 is limited, even if they can be quite heterogeneous having different types, interfaces, vendors, etc. However, these
23 environments are dynamic as changes can occur to existing devices such as software updates, hardware updates,
24 location changes, etc. Within this scenario, each device has its power consumption that needs to be estimated or
25 measured in order to be able to estimate global energy consumption and implement the appropriate power manage-
26 ment strategies. Energy is widely recognized as a significant concern in the research community, given the expo-
27 nential growth of connected devices and appliances. Numerous studies have been conducted to address this issue,
28 including the exploration of greener energy sources, changes in the design phase, task scheduling, machine learning
29 approaches, and encouraging users to adopt energy-efficient behaviors through recommendations. However, the ex-
30 isting literature is subject to certain limitations, such as its application in specific scenarios or environments, limited
31 device, or a restricted number of metrics.

32
33 The goal of this study is to reduce energy consumption by collecting data, building knowledge, and executing
34 optimal green actions using ontologies and reinforcement learning techniques while respecting user preferences.
35 In this study, a scalable, flexible, and automated energy management approach for smart homes is proposed. The
36 developed energy management system could be implemented in any smart home easily. The approach is based on
37 three main components: an automated power estimator, a formal knowledge representation of the environment, and a
38 flexible reinforcement learning energy manager. In this paper, we focus on the knowledge representation and the RL
39 energy manager. On the one hand, the knowledge representation component is used to understand the relationship
40 between metrics from different devices, power consumption, and actions that can be applied. This model correlates
41 the metrics and the potential actions to be carried out by any new device introduced to the environment. The use
42 of an ontology guarantees the inclusion of heterogeneous devices and the general purpose of the approach. On the
43 other hand, the RL energy manager has the central role of identifying the best energy-efficient action that should be
44 taken and its impact on the entire system while respecting user satisfaction. It is composed of an agent generator that
45 creates a RL agent based on the knowledge received from the knowledge component, and a decision-making model
46 that receives the metrics and the power consumption to choose the best action that can be taken to save energy.

47
48 The main contribution of this study is to guarantee the autonomic management of energy consumption in cyber-
49 physical environments. It also provides visibility on energy drains by estimating the power consumption of devices
50 in an automated manner. It lays out a way to formally describe energy-related knowledge. It also proposes a generic
51 RL agent generator that adapts to different environments.

The remainder of this paper is organized as follows: Section 2 reviews the state of art approaches used for energy management in smart environments. In section 3, the theoretical development, architecture, and different components of the approach are presented. Section 4 presents the case studies, experimental setup, and implementation of each system's component. Section 5 discusses the results and validity of the approach. In section 6, we enumerate the main limitations of our approach. Finally, Section 7 concludes and layouts future directions and perspectives.

2. Literature Review

Energy consumption is a significant concern for the research community, therefore, several approaches have been proposed to reduce energy consumption in smart environments. The following section details previous studies that use different techniques to tackle the power management issue.

As seen in [6], user behavior has a direct impact on energy consumption, therefore, some approaches considered user behavior and recommendations. These studies are based on encouraging users and recommending them to take actions that reduce energy consumption. It also includes raising awareness among users about their consumption by showing them visual information and notifying them. In [7], authors focused on the individual level by studying the initial choice of a user to use an energy inefficient application. Likewise, [8] focused on the enterprise level by displaying current consumption and the amount of available renewable energy in collective open workspace. Otherwise, [9] designed a community-scale energy feedback system based on building power meters. It increased the visibility of a district's energy consumption. In addition, others used games to show users their energy consumption and motivate them to turn off unused appliances [10–12].

Some authors have suggested scheduling tasks defined by changing the time of execution of a specific task. Scheduling is usually done to benefit from energy production when it is greener [13], better energy efficient execution time slot [14], or during the absence of users [15, 16]. In addition, scheduling could have a target of detecting unused devices to change their operating mode [17–19]. In [17], authors proposed an approach that minimizes energy consumed by home appliances during standby mode, where connected outlets cut off the current from a device consuming below a certain threshold. Likewise, [18] proposed energy monitoring and saving functions that reduce standby power automatically by cutting-off electricity for unused devices. In [19], an approach that compromises between energy efficiency and reliability of a system in both shutdown and scale-down scheduling techniques was proposed. Scheduling is implemented in different contexts such as data centers [20–22], Wireless Sensor Networks (WSN) [23, 24], smart homes [6, 25], and IoT [26].

Improving system design and using newer technologies have a significant impact on reducing energy consumption. Therefore, some studies aim to raise a device's efficiency while reducing its power consumption by making changes in the design phase of a device. It can take the form of using newer technologies or making changes in the architecture and design of the system. In [27], design decisions were made after evaluating and redesigning different mechanical principles, regulation algorithms, and energy consumed during different communication scenarios. In [28], a model-based design methodology for residential micro-grid was proposed. In addition to a simulation presenting structural and behavioral parts of a residential micro-grid system. In [29], an energy management framework used for autonomous electric vehicles in the smart grid was proposed. Energy savings in this system were done during design (architecture and communication protocol). In [30], contextual collected data and machine learning techniques were used in the context of a cloth dyeing factory. Their goal was to identify and predict inefficiency in order to change the manufacturing process.

Other studies focus on power demand and supply. Power demand changes regularly with time, and so does the power supply, especially with renewable energy such as photovoltaic cells and wind turbines. Providers need to match their power production with the demand to be as efficient as possible [31]. Many methods are used such as predicting the power needed at a specific time based on historical records [32], balancing available power supply and demand [33, 34] or storing energy when it is available which is the case for electric vehicles [35, 36].

1 More advanced studies proposed adapting approaches based on contextual data. These approaches are based on 1
2 the ability of a system to change its initial configuration by tuning some parameters allowing it to adapt to changes 2
3 that take place during run-time. Some are based on a finite number of cases that were predefined during design 3
4 time or multiple predefined operating modes. In [37], authors presented an energy-efficient reconfiguration tool built 4
5 on Raspberry Pi in an intelligent transportation scenario. It identified energy-consuming concerns at design time, 5
6 analyzed configurations and variants, created a file containing all the possible configurations and the consumption 6
7 of each of them, selected an initial configuration, and reconfigured devices at run-time according to the context. 7
8 The authors in [38] proposed multi-operating modes for each layer of the system to increase energy efficiency. In 8
9 this approach, the optimization of each layer of the system is done by the layer itself based on predefined modes 9
10 (e.g., sleep, active, low frequency, high-frequency modes). The authors in [39] ensured energy optimization while 10
11 satisfying the user comfort by proposing a smart zoning multiple-mode feedback system in a smart building based 11
12 on the usage of each room, its orientation, and its occupancy. Each zone can be in normal utilization, pre-cooling, 12
13 or power-off mode. Other solutions propose autonomous systems that use machine learning techniques and have 13
14 knowledge bases. 14

15
16 Recently, there is a growing body of work focused on using RL for energy management in smart homes and 16
17 buildings [40, 41]. In [42], a machine learning energy manager for hybrid electrical vehicles (which have an internal 17
18 combustion engine and an electric motor) was proposed. A nested reinforcement learning approach was adopted 18
19 where an inner-loop was responsible for choosing the electric and fuel engines to minimize fuel usage, while the 19
20 outer loop was in charge of modulating the battery health degradation. Finally, another promising line of research 20
21 would be the use of both ontologies and machine learning to accomplish context understanding that leads to bet- 21
22 ter power management. Web-of-Objects (WoO) [43] defines a systematic description of smart distributed applica- 22
23 tions enabling the integration of any connected device. It considers some contextual data (e.g., weather and loca- 23
24 tion). However, power consumption per device was not considered. In [44], authors proposed an intelligent multi- 24
25 agent system, called Thinkhome. It was responsible for the execution of control strategies that manage the building 25
26 state. It introduces the semantic context to artificial intelligence by using an ontology for knowledge representation. 26
27 ThinkHome considers mainly user comfort, user behavior, contextual information (e.g., weather), building infor- 27
28 mation (e.g., spaces and walls), and energy information (e.g., environmental impact and energy sources). Lack of 28
29 heterogeneity is the main drawback of this approach due to its focus on Heating, Ventilation, and Air Conditioning 29
30 (HVAC) systems without studying other types of devices. In [45], authors modeled the energy management sys- 30
31 tem using a Markov decision process. They described states, actions, transitions, and rewards of the environment. 31
32 Their environment is based on a smart energy building connected, an external grid, a distributed renewable energy 32
33 source, an energy storage system, and a vehicle-to-grid station. In their study, actions were limited to {Buying, 33
34 Charging, Discharging, Selling} energy, thus, a high-level solution with no impact on individual devices. In [46], 34
35 authors proposed a RL algorithm that ensures user satisfaction, followed by energy savings and load shifting. It 35
36 focused on houses equipped with renewable energy resources and setting a dynamic indoor temperature setpoint. A 36
37 multi-agent RL for home-based demand response was proposed in [47]. Each of the agents was responsible for a 37
38 type of device (non-shiftable, power-shiftable, time-shiftable, or electric vehicle). In [48], authors proposed a home 38
39 energy optimization strategy based on deep Q-learning (DQN) and double deep Q-learning (DDQN) to perform the 39
40 scheduling of home appliances tasks. In [49], authors proposed a demand response algorithm of the next hour for 40
41 home energy management systems. It focused on reducing electricity cost by shifting loads based on historical data. 41
42 [50], proposed a deep RL energy management algorithm based on Deep Deterministic Policy Gradients (DDPG) 42
43 to control a HVAC system. However, an important issue in the previously mentioned existing research is that their 43
44 environments are static and it is difficult to train agents in new environments. 44

45 3. Theoretical Development 46

47 3.1. Framework Architecture 48

49
50 The approach proposed in this study aims at providing an intelligent power management solution for connected 50
51 environments based on three components: (1) an automated approach to empirically generate power estimation 51

models (PEMs) for a large set of devices; (2) an extension of the SAREF ontology as a knowledge-base contextual understanding component; and (3) an automated knowledge-based reinforcement learning environment and agent generator that generates a decision-making agent responsible of choosing the optimal actions to be executed. It also includes a data repository component formed by a collection of databases that include previously collected metrics, states, applied actions, rewards, power consumption, power estimation models, and potential external data (e.g., outdoor temperature). The proposed architecture ensures the following criteria: high autonomy maturity level, flexibility, heterogeneity of devices, general-purpose, and ability to deal with a large variety of metrics. This architecture provides high flexibility in dynamic environments, therefore, it is not domain specific and can be implemented easily in another domain. Any changes or updates that occur in the environment are considered and lead to the calibration of estimation models and reinforcement learning agents. It also guarantees a high autonomy level where high-level user-oriented energy policies control the infrastructure operations in an automated manner. Beside the high flexibility, implementations can be conducted on any type of device because of the knowledge-base component and the automated power modeling technique. Therefore, high heterogeneity of devices and metrics is guaranteed.

Fig. 1 shows the architecture of the automated intelligent energy management approach, including its distinct but complementary components. It is able to collect metrics, estimate power consumption per device, and represent knowledge present in the environment and the relationship between devices. In addition, it can generate a reinforcement learning environment used for training an agent with the policy of optimizing energy consumption while respecting user preferences.

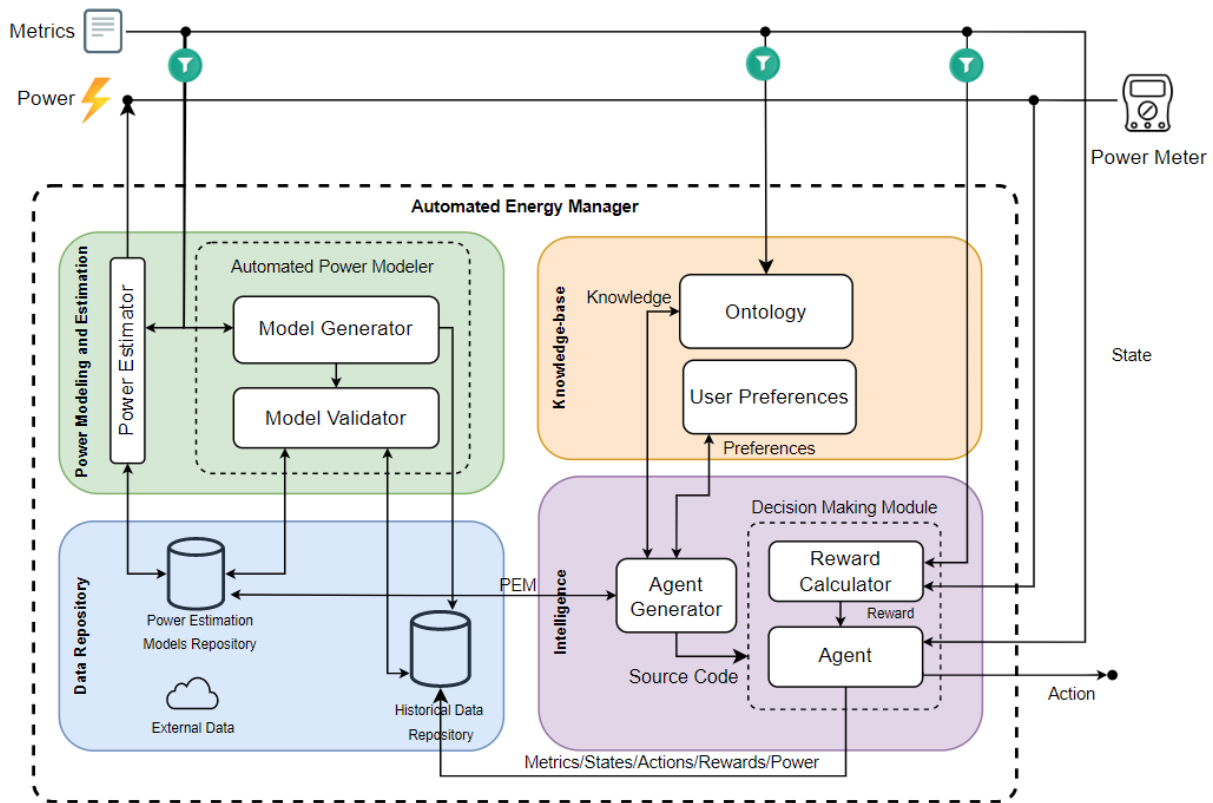


Fig. 1. Automated Energy Management Architecture

3.2. Autonomic Power Estimator

The knowledge of devices' real-time power consumption is an essential and challenging need to quantify and monitor energy leakage and saving [51]. The used automated power estimator ensures up-to-date and accurate power models with error rates as low as 0.33% and up to 7.81% for linear models, and 0.3% up to 3.83% for polynomial models [52]. It aims to empirically generate power models for, potentially, unlimited devices and configurations. The power estimation process follows a crowd-sourcing architecture where benchmarking components can run on any device, generate empirical data, and lastly, the power model generator component will generate an accurate power model for the specific device, or improve the model if a previous one already exists. Generated power estimation models can be shared between different devices in the environment. Monitoring software can connect to this component to query and retrieve the most accurate and up-to-date power model for their devices. Once the model is generated, power consumption can be estimated with negligible overhead in real-time. The overall overhead difference between using the estimator and not is quite low with an absolute average difference of 0.084 watts (corresponds to a relative difference of 1.32%) for the RPi Zero W, and only 0.109 watts (corresponds to a relative difference of 0.72%) for the RPi 3B+. With time, the accuracy of the model can be improved as more data are collected and fed to generate more accurate regression models. In this paper, we consider that an accurate real-time power estimation model was previously generated for each device in the environment.

3.3. Knowledge-Base Representation

An ontology is a defined set of concepts and relationships used to represent the meanings and a shared understanding in a certain domain [53]. Ontologies are considered highly effective for knowledge modeling and information retrieval. They provide a common representation and definition of the concepts and the relationship between them. In addition, ontologies support inference to discover new relationships and knowledge. They are scalable and their reuse is possible. Therefore, an ontology is used to formally represent the environment's knowledge.

SAREF, or Smart Applications REference Ontology [54], is a well-known and adopted ontology used to represent devices, their properties, and capabilities in the context of smart homes. It focuses on defining the properties of devices, which can be measurable, controllable, or both. Moreover, SAREF defines the type of device, such as appliance, sensor, actuator, meter, or HVAC. Overall, SAREF provides a well-defined and organized framework that not only enables developers and researchers to better understand the properties and capabilities of different types of devices, but also assures a high level of interoperability between devices.

Moreover, existing ontologies do not consider power estimation and measurement in real time. Therefore, it is necessary to extend one of these ontologies to represent this concept, as seen in Fig. 2. The way these concepts are defined in SAREF provides a compatible and easy way to represent the knowledge base for reinforcement learning model generation.

In order to include the real-time power and different layers of a device into the knowledge base, we propose the extension of the SAREF ontology with the following classes, attributes, and relations:

- *ecps:Location* is the geographic space in which a device is implemented. It helps building a contextual understanding of the environment and identify devices that share the same location.
- *ecps:Layer* is the structural level of the system to which a property belongs. It can be application, computational, network, storage, and contextual.
- *ecps:LayerState* is the state in which a specific layer is running. It composes the state of the device.
- *ecps:Power Estimation Model* is the mathematical model that takes as input the properties of a device and returns the power consumption.
- *ecps:Real Power* is the real-time power consumption of a device. It can be measured directly by a hardware wattmeter or estimated using an estimation model.
- *ecps:hasRange* represents the accepted values for each property. It takes the form of a list of accepted values or minimum and maximum accepted values. Ranges specified with a minimum and maximum can include a step that defines a fixed value increased or decreased from a property value.
- *ecps:deviceId*, *ecps:deviceName*, *ecps:propertyId*, *ecps:propertyName* represent respectively the id and name of a device used to identify a device from another. In addition to the id and name of a property.

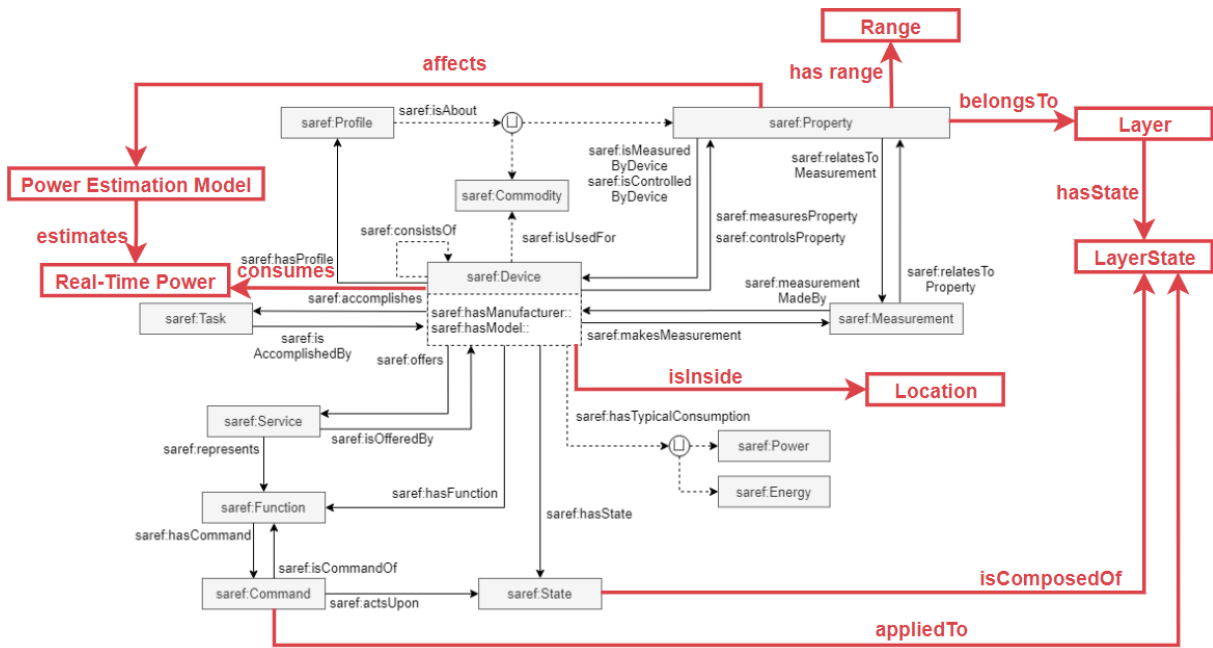


Fig. 2. SAREF Extension Overview (The extension is represented using the red color)

3.4. Automated Reinforcement Learning Energy Management

Fig. 3 presents the architecture of the reinforcement learning environment and the agent generation process. The agent generator aims to collect intelligence from the ontology and create a virtual environment and a reinforcement learning model based on the devices present in the environment and the relations between them. First, the agent generator requests necessary knowledge from the ontology and the power estimation models repository for each device. Once this information is retrieved, the agent generator creates the source code of the reinforcement learning environment or modifies the existing one. The generated source code includes all the necessary information needed to optimize energy consumption using reinforcement learning techniques such as states, actions, reward calculation, and power estimation models.

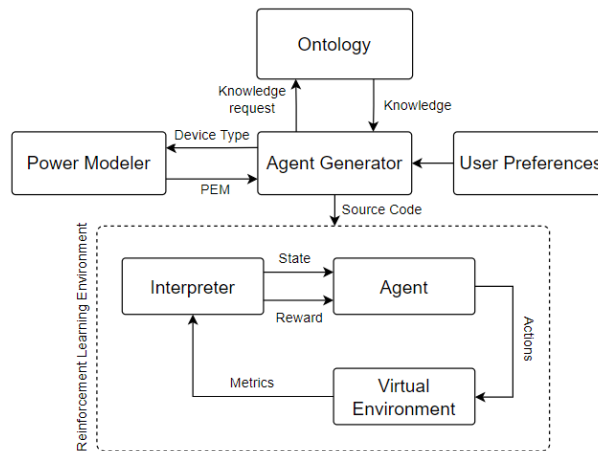


Fig. 3. Reinforcement Learning Environment Generation Architecture

4. Implementation

4.1. Ontology Implementation

The ontology is defined using the Web Ontology Language (OWL) [55], a language of knowledge representation for ontologies in the Protégé ontology editor and knowledge management system [56]. As seen in Fig. 4, a series of classes, data properties (attributes), and object properties (relations) are defined in order to make the ontology suitable with the agent generator. Newly introduced concepts' names start with the prefix *ecps*.

We also create individuals for each of the defined case studies. Fig. 4d and Fig. 4e define each device, controllable property, measurable property, and power estimation model present respectively in the living room scenario and the HVAC scenario. In addition, the relation between these concepts is created to represent the knowledge about the environment to the system.

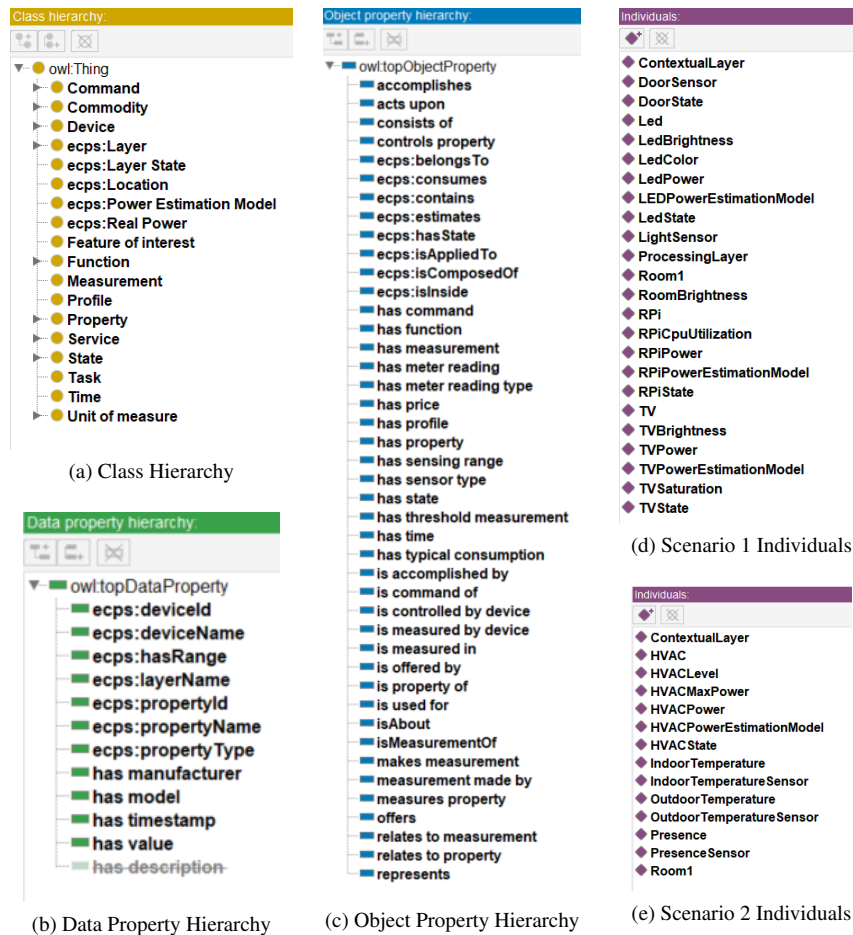


Fig. 4. SAREF Ontology Extension Implementation in Protégé

4.2. Reinforcement Learning Implementation

The agent generator is responsible for the generation of each section of the reinforcement learning agent based on TensorFlow and Keras-RL, in addition to an environment built using OpenAI Gym. Algorithm 1 provides the pseudocode of how the reinforcement learning environment is generated. First, a series of SPARQL queries are

executed to collect knowledge from the ontology using the *request_sparql* function. Graphs containing information concerning actions, metrics, states, and power estimation models of each device corresponding to a unique location are collected. Fig. 5 shows an example of the two SPARQL queries used to collect power estimation models and actions each device can accomplish.

<pre> 6 SELECT DISTINCT ?Power ?DataType ?PEM_value ?State ?Device 7 WHERE { 8 ?Power rdf:type saref:ecps:RealPower. 9 ?Device saref:ecps:consumes ?Power. 10 ?PEM saref:ecps:estimates ?Power. 11 ?PEM saref:hasValue ?PEM_value. 12 ?device saref:ecps:isInside ?location, saref:Room1. 13 ?Power rdf:type ?Class. 14 ?Class rdfs:subClassOf ?SuperClass. 15 ?SuperClass owl:onDataRange ?DataType. 16 ?Device saref:hasState ?State. 17 } </pre> <p>(a) Power Estimation Model per Device</p>	<pre> 6 SELECT ?Device ?Property ?DataType ?Range 7 WHERE { 8 ?Device saref:controlsProperty ?Property; 9 saref:ecps:isInside saref:Room1. 10 ?Property rdf:type ?Class. 11 ?Class rdfs:subClassOf ?SuperClass. 12 ?SuperClass owl:onDataRange ?DataType. 13 ?Property saref:ecps:hasRange ?Range. 14 } </pre> <p>(b) Environment Actions per Device</p>
---	--

Fig. 5. SPARQL Queries Examples

The collected knowledge is used for the creation of the source code of all the functions that compose the reinforcement learning environment. As seen in algorithm 1, a function called *generate_code* generates the source code of each of the reinforcement learning environment sections based on the collected ontology knowledge. It generates the source code of six main components: (a) actions, (b) states, (c) rewards, (d) power estimation, (e) initialization, and (f) reset. A template of the source code of the reinforcement learning environment is created. It includes the necessary libraries and fixed chunks of code. However, the automated source code generator writes the majority of the code by using the *write_code* function. It identifies the blank variable sections of the source code that change from one environment to another and writes the correct code in each section. It can generate the source code of any number of properties and devices due to its extensible way and ability to request knowledge and use it.

Algorithm 1: Reinforcement Learning Environment Generation Algorithm

```

32 1: procedure GENERATERLE(ontology)
33 2:   Initialize sparql queries
34 3:   actions, metrics, states, power ← request_sparql(ontology, sparql_query)
35 4:   actions_code = generate_code(actions, states) // Generate source code
36 5:   states_code = generate_code(metrics, states)
37 6:   rewards_code = generate_code(power, metrics)
38 7:   power_estimation_code = generate_code(power, metrics)
39 8:   init_code = generate_code(actions, metrics, states, power)
40 9:   reset_code = generate_code(actions, metrics, states)
41 10:  write_code(init_code, reset_code, actions_code, states_code, rewards_code, power_estimation_code)
42 11: end procedure

```

Once the source code is generated, the reinforcement learning environment is ready to be launched. Its structure is presented in algorithm 2. The following explains the role of each of its functions:

- *__init__* function defines the observation and the action spaces, i.e., the minimum and maximum acceptable values for each observed metric or action. In addition to a few other attributes, i.e., accepted values for each property and the initial state of each device. This function also initializes states, properties, and power consumption per device. It also encodes and maps properties and states from their initial format to a numerical or Boolean format to be accepted by the reinforcement learning algorithm.

- *step* function executes a step in the environment by applying an action and returns the new reward, observation, power consumption per device, and other info. It includes the list of possible actions and the reward calculation based on user preferences and total power consumption. In this function, randomness and external changes impacting the environment can be applied (e.g., outside temperature changes).
- *calculate_power* function contains the power estimation models for each device in the environment. It returns the value of the power consumption of a specific device at a particular time.
- *total_power* function returns the total power consumption of all devices in the environment at a particular time.
- *reset* function re-initiates the environment to its initial state without changing the observation and the action spaces.
- *log* function records events occurring in the environment into a CSV file, in particular, the value of each property, states of devices, power consumption for each device, total power consumption, executed action, and reward value.

The reward value is a rational number bounded between -1 and +1. It is calculated based on three equations (1), (2), and (3). First, it considers user preferences and makes sure all user preferences are respected. Doing so guarantees the respect of the user preferences before starting to manage power consumption. If any preference is not respected, the reward will be negative between 0 and -1. It is calculated by counting the number of unrespected preferences and dividing it by the total number of user preferences, as seen in Eq. (1), when the user preferences are not measurable.

$$R_t = -1 \times \frac{\text{count}(\text{unsatisfied user preferences})}{\text{count}(\text{user preferences})} \in [-1, 0] \quad (1)$$

However, if the user preferences are measurable (i.e., temperature), the reward is calculated based on the difference between the property value V_t^i at a specific time t and the desired value V_{user}^i , as seen in Eq. (2).

$$R_t = -1 \times \frac{|V_{user}^i - V_t^i|}{1 + |V_{user}^i - V_t^i|} \in [-1, 0] \quad (2)$$

Once all user basic preferences are respected, Eq. (3) is adopted to calculate the reward and ensure a higher reward for each power-aware action.

This equation is bounded between -1 and 1. Its value is -1 when the current power consumption is equal to the maximum power consumption and tends to 1 when the power consumption goes down to 0. The equation is modeled in such a manner to guarantee that the highest reward value is given to our agent if the energy consumption in the environment is the lowest possible while respecting user preferences. It also guarantees that the lowest reward is obtained when the power consumption is at the highest level possible. Once user preferences are not respected anymore, we stop calculating the reward using Eq. (3) and use again Eq. (1) and Eq. (2). Reaching 0 as power consumption for the entire environment is difficult due to user preferences but remains possible in some extreme cases (occupant is not present in a room with only lamps that should be all turned off).

$$R_t = 1 - 2 \times \frac{P_t^{total}}{P^{max}} \in [-1, 1] \quad (3)$$

where: R_t is the reward calculated at time t , P_t^{total} is the total power consumption estimated or measured at time t , and P^{max} is the maximum total power consumption of the environment.

Algorithm 3 shows the model and agent generation pseudocode. First, states and actions are defined based on the previously generated reinforcement learning environment. Then, the model is built using the *build_model* function. The model is based on a Deep-Q network that takes as an input layer the set of states and returns possible actions in output nodes. It has two hidden dense Keras neural network layers along with a ReLU activation. The *build_agent* function specifies the policy and builds the agent based on the previously built model. The agent and model are compiled using the *compile* function, they are later used for training and testing. The *fit* function trains the agent given the environment and the number of training steps.

Algorithm 2: Reinforcement Learning Environment Algorithm

```

1
2 Procedure __init__(self) :
3   Set controlled properties upper bounds
4   Set controlled properties lower bounds
5   Set measured properties observation space
6   Initialize a set of states
7   Initialize a set of properties
8   Initialize a set of power consumption
9   Map states to integer values
10  Initialize total energy
11
12 Procedure reset (self) :
13   Reinitialize states
14   Reinitialize properties
15   Reinitialize power consumption
16   Reinitialize total energy
17
18 Function calculate_power (device_type, device_state, properties[]) :
19   if device_type ∈ devices then
20     power = estimate_power(device_type, device_state, properties[])
21   else
22     power = default_manufacturing_power(device_type, device_state)
23   return power
24
25 Function total_power (self) :
26   total_power = 0
27   foreach device  $d_i$  ∈ devices do
28      $d_i$ .power = calculate_power( $d_i$ .type,  $d_i$ .state, properties[])
29     total_power = total_power +  $d_i$ .power
30   return total_power
31
32 Function step (action) :
33   Execute(action)
34   if properties ≠ userPreferences then
35     if userPreferences.isCountable then
36       reward =  $-1 \times \frac{\text{count}(\text{userPreferences.unsatisfied})}{\text{count}(\text{userPreferences})}$ 
37     else
38       reward =  $-1 \times \frac{|V_{user}^i - V_t^i|}{1 + |V_{user}^i - V_t^i|}$ 
39     else
40       reward =  $1 - 2 \times \frac{\text{self.total\_power}}{\text{MAX\_POWER}}$ 
41   cycle_length = cycle_length - 1
42   if cycle_length == 0 then
43     day_ended = True
44     daily_energy = daily_energy + total_power(self)
45   else
46     day_ended = False
47     daily_energy = 0
48   return reward, properties[], power[], states[], total_power
49
50 Function log (self) :
51   foreach device  $d_i$  ∈ devices do
52     foreach property  $p_i$  ∈  $d_i$ .properties do
53       data_to_save+ = property
54     data_to_save+ =  $d_i$ .power
55     data_to_save+ =  $d_i$ .state
56   data_to_save+ = env.total_power_consumption
57   data_to_save+ = env.action
58   write(data_to_save)
59   return total_power
60

```

Algorithm 3: Deep Reinforcement Learning Model and Agent

```

1: procedure CREATEERLAGENT(generated_environmentenv)
2:   states = env.observation_space.shape[0]
3:   actions = env.action_space.n
4:   model = build_model(states, actions)
5:   agent = build_agent(model, actions)
6:   agent.compile(model, actions)
7:   agent.fit(env, nb_steps)
8: end procedure

```

4.3. Case Studies

Smart home environments are scalable and contain devices that are quite heterogeneous having different types, interfaces, vendors, etc. In addition, these environments are dynamic as changes can occur to existing devices such as software updates, hardware updates that lead to a change in power consumption or location changes that impact the knowledge of the environment. In the following section, our aim is to ensure that all four components of our approach work cohesively to facilitate energy management while simultaneously considering user satisfaction. Our approach was evaluated in two separate case studies, both in the context of a smart home. The first case study (I) was conducted in a living room equipped with a variety of devices, while the second case study (II) involved a smart home with a heating system. These two studies provided us with an opportunity to analyze device types, data types, and customizable environments. The main objective of case study I is to reduce energy consumption with data provided from different devices, while the objective of case study II is to guarantee that it is possible to increase user satisfaction and reduce energy consumption.

4.3.1. Living Room Case Study

As seen in Fig. 6, the first simulated case study is defined as a living room of a smart home with several devices. A led bulb controls its brightness (LedBrightness) and color (LedColor). A TV controls its brightness (TVBrightness). A Raspberry Pi that measures its CPU utilization and has a light sensor mounted on it that measures the room brightness. The LED bulb, TV, and raspberry pi are considered power consumers and have each a power estimation model that gives their power consumption in real-time. A door sensor checks if the door is opened or closed without being able to change its state. The door sensor is not considered as a power consumer because its power consumption is minimal and unchangeable. LedBrightness has a minimum value of 0 and a maximum value of 100 and can be incremented or decremented by 1% per minute to ensure a smooth light transition that is not annoying for the users. LightColor can have one of the following values: red, blue, green, and white. Devices states are usually on, off, or sleep. However, they are not limited to these states and can take many other forms. For the living room scenario, user preferences are generated based on [57]. They are manually adjusted to add randomness that may occur from a day to another, between weeks, and during weekends.

4.3.2. HVAC Case Study

Another simulated case study is conducted with different devices in a different context. It is based on a HVAC system in a smart home. The room where the HVAC is mounted is equipped with an indoor temperature sensor and a presence sensor. In addition, a temperature sensor is positioned outside the room to capture the outdoor temperature. We model the environment using the previously proposed extension of the SAREF ontology as done in the living room case study. Then, we launch the agent generator to build the environment and the reinforcement learning agent. To simulate indoor temperature dynamics in the environment, the exponential decay model in Eq. (4) is adopted [58].

$$T_{t+1}^{in} = \varepsilon \times T_t^{in} + (1 - \varepsilon) \times (T_t^{out} \pm \frac{\eta_{hvac}}{A} \times e_t) \quad (4)$$

(+ : heating, - : cooling)

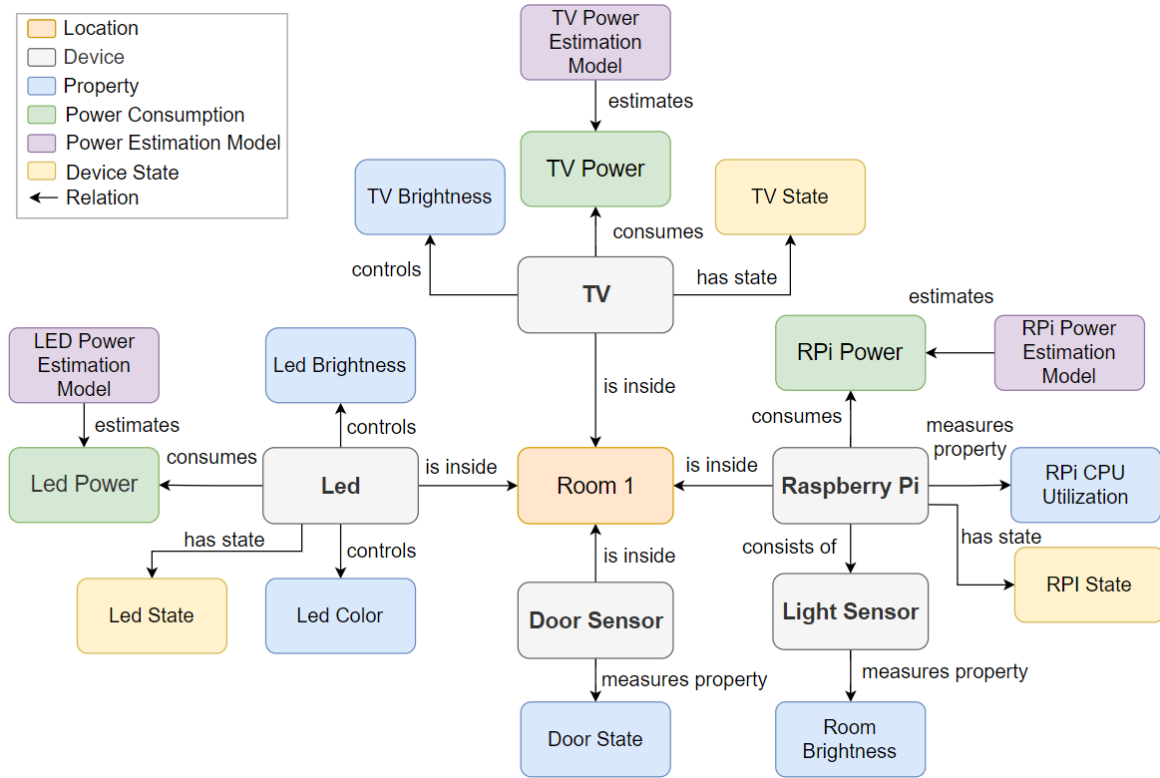


Fig. 6. Devices, Properties, and Relations Representation of the Living Room Case Study

where T_t^{in} denotes room internal temperature during time t , T_t^{out} is the real outside temperature during time t , $\varepsilon \in [0, 1]$ is a thermal time constant, η_{hvac} is the thermal conversion efficiency (heating) or the coefficient of performance (cooling), A is the thermal conductivity that varies based on the insulation of a space, and $e_t \in [0, e_{max}]$ is the electric HVAC system power input at time t (It ranges between 0 when the HVAC system is turned off and e_{max} the maximum power consumption of the HVAC system). We consider $\varepsilon = 0.7$, $\eta_{hvac} = 2.5$, $A = 252 \text{ w}/^\circ\text{C}$ based on the study conducted in [59], and $e_{max} = 2000 \text{ w}$.

Eq. (4) was widely adopted in previous research such as in [59–63]. This experiment used the outdoor temperature dataset proposed by USCRN/USRCRN [64] due to its accuracy, low error rates, and high quality of collected data. The dataset is cleaned and filtered mainly to remove missing data indicated by the lowest possible integer, i.e., -9999,0. The used dataset was collected in Santa Barbara, California, the USA in 2021. User preferences temperatures are extracted from the study conducted in [65]. User temperature preferences range between 20°C and 24°C during the day. They range between 18°C and 22°C during the night when the home occupant is sleeping.

Fig. 7 shows the outside temperature of a sample day in January 2021. In addition, it shows the calculated inside temperature with the HVAC turned OFF (Fig. 7a) and ON (Fig. 7b). In Fig. 7b, two conditions are added to guarantee that the inside temperature respects user preferences and illustrates a real-life use case.

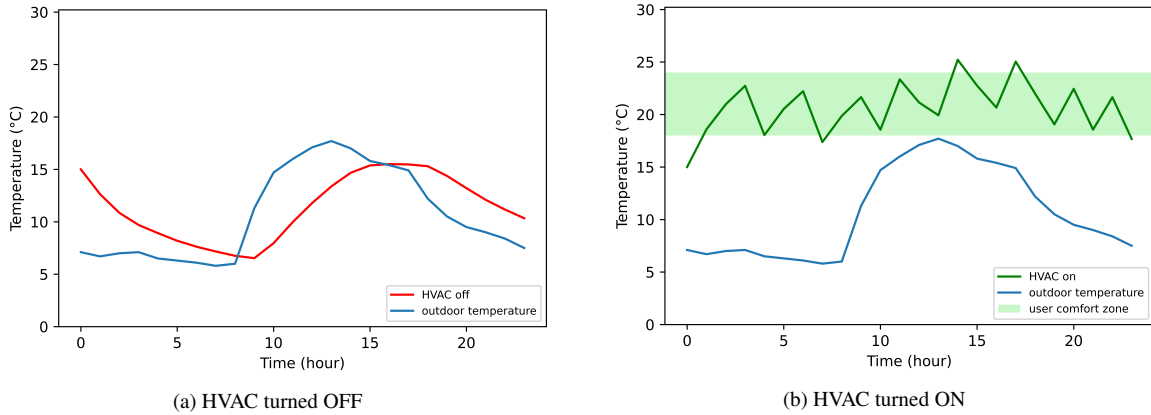


Fig. 7. Outdoor and Indoor Temperature Samples

5. Results and discussion

A prototype of the proposed power management approach has been developed to reduce power consumption in smart connected environments. In particular, the following two application case studies have been developed, deployed, and experimented in a smart home for validation purposes.

5.1. Living Room Case Study

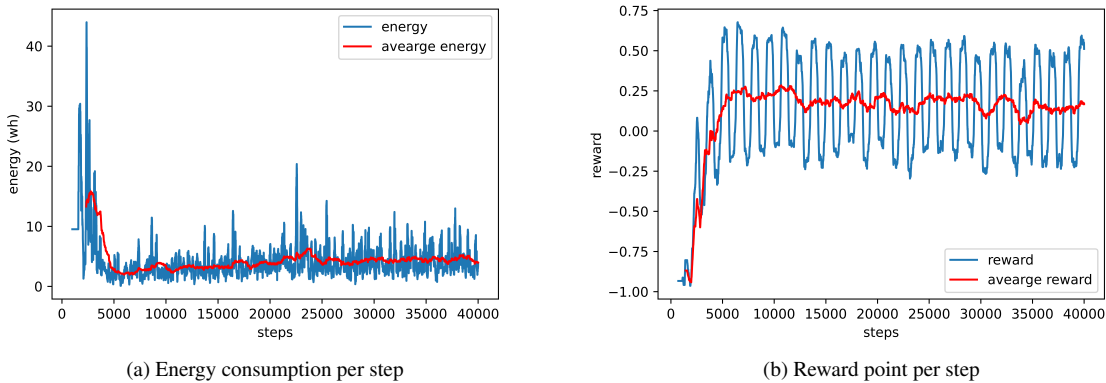


Fig. 8. Living Room Case Study Experimental Results

A first experiment has been conducted for 70 episodes (100k steps). Each episode lasts for 24 hours and one step is executed per minute (1660 steps per episode). Once per minute, metrics and states of devices are collected. Then, the agent chooses an action to be executed per device. However, the agent has the possibility to execute a neutral action that keeps all devices in their current state. User preferences for the 28 days are defined (i.e., led state, led brightness, led color, TV brightness, and TV state). Fig. 8 shows the results of the living room case study. Fig. 8a shows the total energy consumption of the environment per step and their average per day. Fig. 8b shows the total reward points and their average per day calculated by the agent. In such scenario, energy has converged during a brief period of time (4 days), and the reward has passed from a negative value to a positive one meaning that the user

preferences are respected and that the power consumption is at its lower accepted values. This study case shows that the proposed framework can deal with different types of devices, properties, and actions related to these devices. It validates that this approach can be used to represent knowledge, generate reinforcement learning environments, and train agents. In addition, it confirms the ability to improve energy efficiency and savings in the long term.

5.2. HVAC Case Study

A second experiment was conducted in the HVAC environment for 596 episodes (100k steps). Each episode has lasted one week with a frequency of one step executed each hour (168 steps per episode). User preferences of the minimum and maximum accepted temperatures are defined for an entire week. Fig. 9 shows the results of the living room case study. Results show an increase in the reward per episode, therefore, the user satisfaction and the power consumption, as seen in Fig. 9b. User preferences vary between the day and the night. However, for the purpose of simplicity, Fig. 9b graphs the user comfort zone between the lower and higher accepted values. Fig. 9a shows the room temperature per step and the average room temperature per episode. Results show that in the first part of the training (before the 20 000th step), the indoor temperature is under the user preferences and converges with time to respect user preferences. Therefore, validating the ability of the proposed reward function to lead to a better user experience. Initial room temperature is randomized for each new episode ranging between 12°C and 18°C, and the initial HVAC level is a random value between 30% and 70%.

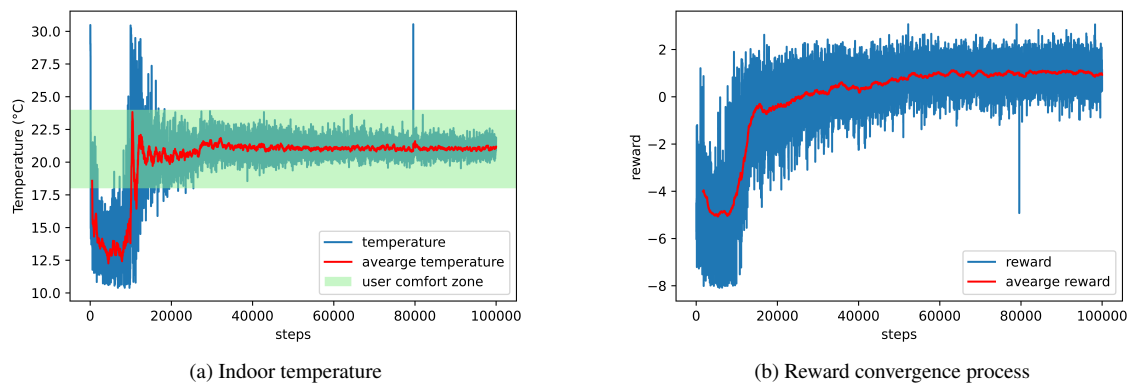


Fig. 9. HVAC Case Study Experimental Results

To compare the energy consumption of the proposed approach to other control values, we identify three different baselines:

- *Baseline 1*: Represents the results of the proposed automated energy management approach, proposed in this paper, including the knowledge, environment and model generation, and training of the agent. It also considers user preferences and presence.
- *Baseline 2*: Represents an environment where user preference indoor temperature is set to 21°C. This value is obtained by calculating the average temperature between the minimum and maximum accepted values. It was previously represented in Figure 7b. The HVAC is turned ON whenever the room temperature goes below 21°C and turned OFF when it outpaces this value. The obtained values are based on the captured indoor and outdoor temperature readings.
- *Baseline 3*: Represents an environment with a traditional electric heater that stays ON all the time without capturing and considering any of the indoor or outdoor temperatures.

Fig. 10 shows the mean value of energy for each of the previously defined baselines. The results show that the proposed method has the lowest energy consumption compared to the two other traditional methods. Baseline 1 energy consumption equals 932,82 wh, while baselines 2 and 3, respectively, are 1083,3 wh and 2000 wh. We

consider that the model has already converged before step number 40 000 and energy consumption is almost stable. Therefore, in the 1st case, the average energy consumption is calculated with data corresponding to steps higher than 40 000 steps.

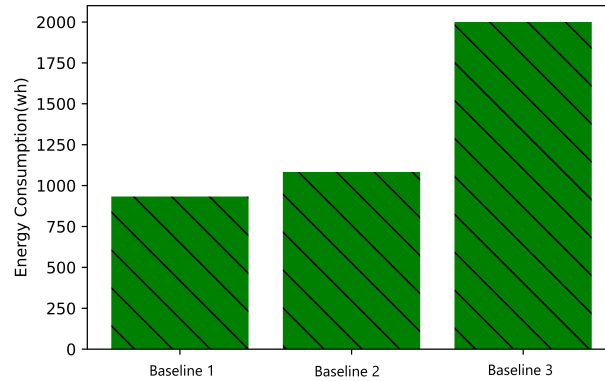


Fig. 10. Mean Value of Energy in the HVAC Case Study

The presented results highlight a significant decrease in power consumption while respecting user preferences in both case studies. In addition, our proposal performs well given the low convergence period (4 days for case study I), the high user satisfaction (the temperature is maintained in the comfort zone for case study II), and the energy consumption decreases. From these results, it is clear that the reward functions are well adapted as feedback to the agent that has the two main goals of facilitating energy optimization and user comfort.

The implementation in two use cases, with completely different devices, confirms the general purpose of our approach because the code generation is done in an automated way allowing the adaptation of the scenario to the framework. It proves the possibility of generating customizable environments with various devices. The approach also showed the capability of dealing with different metrics that correspond to different scenarios and layers of CPS, therefore, we conclude that the approach is extensible on the level of a smart home.

6. Limitations

Although our automated energy management framework is established as an efficient way to manage energy consumption, it is important to acknowledge its limitations.

One limitation of our framework is that it utilizes a Markov decision process, where the current states of devices exclusively determine the direct future action, without requiring any knowledge of the past. This approach enables the system to react efficiently to changes in device states and provides real-time decision-making capabilities.

A major source of limitation of our energy management framework is that it relies on a single agent to simultaneously optimize two distinct objectives: reducing power consumption and increasing user satisfaction. While our current approach has demonstrated promising results, this design choice may not be ideal for all scenarios. However, emerging multi-agent reinforcement learning algorithms offer solutions to handle complex environments with multiple optimization problems. In these approaches, multiple agents learn to coexist in a shared environment, with each agent motivated by its own reward function and executing actions that align with its objectives, such as energy efficiency, user comfort, or cost reduction. By utilizing multi-agent RL, we could potentially achieve better outcomes and overcome the limitations of our current approach.

7. Conclusion and Future Directions

In this paper, we proposed an automated management framework designed to optimize energy usage in smart homes using knowledge representation and reinforcement learning techniques. Our approach allows the representation of knowledge and generation of virtual learning environments with a spotlight on energy to provide the best action on how to reduce energy consumption while respecting user preferences. The framework has been validated through two proof-of-concept case studies in a smart home environment, where comprehensive experiments were conducted to evaluate its effectiveness.

The results of these experiments demonstrate the significant potential of this approach in reducing energy consumption while still meeting the user's preferences. Moreover, the framework demonstrated high user satisfaction and was able to achieve its objectives, defined by our three rewards equations, within a significantly short convergence period. The use of knowledge representation and reinforcement learning enables the framework to learn from its past experiences and improve over time, making it a sustainable and effective solution for managing energy in smart homes. This proposed energy management framework represents a significant step forward in smart home technology, enabling more efficient energy consumption while still respecting user preferences. The results of the experiments demonstrate the effectiveness of the framework and its potential for widespread adoption.

Although the proposed method has shown promising results in the current study, much research still needs to be done to fully evaluate the potential of this proposal in a real-world environment with physical sensors and actuators in real homes, where the behavior of the system can be closely observed and analyzed. In addition, we plan to expand the implementation of the method to cover additional home devices on a larger scale. The extension could include many rooms in a home or even multiple homes in a building that will provide valuable insights into the scalability of the method and how it performs in more complex environments. Furthermore, it is important to investigate the impact of such an approach over a longer period of time. Doing so will help us understand how the behavior of the system evolves as users' needs and habits change over time. It is essential that future research looks into extreme cases (e.g., during the COVID-19 pandemic) that will help us understand how the system can adapt to sudden changes in user behavior and how it can continue to provide good results even in unpredictable and rapidly changing circumstances. Finally, it will be critical for future research to investigate multi-agent RL algorithms that can be used as solutions for complex environments with more than one optimization problem.

Acknowledgement

The project leading to this publication has received funding from Excellence Initiative of Université de Pau et des Pays de l'Adour - I-Site E2S UPPA, a French "Investissements d'Avenir" programme.

Conflict of Interest

The authors declare that they have no conflict of interest.

References

- [1] D. Mercer, Global Connected and IoT Device Forecast Update, Technical Report, May, Strategy Analytics - Research, Experts, and Analytics, 2019. <https://www.strategyanalytics.com/access-services/devices/connected-home/consumer-electronics/reports/report-detail/global-connected-and-iot-device-forecast-update>.
- [2] Dave Evans, The Internet of Things: How the Next Evolution of the Internet Is Changing Everything, Technical Report, Vol. 1, 2011, pp. 1–11. https://www.cisco.com/c/dam/en_us/about/ac79/docs/innov/IoT_IBSG_0411FINAL.pdf.
- [3] IEA, Electricity Information 2019 – Analysis - IEA [Online], 2019. <https://www.iea.org/reports/electricity-information-2019>.
- [4] The small devices that use lots of power and the big buildings that don't - Drax Global [Online]. <https://www.drax.com/electrification/small-devices-use-lots-power-big-buildings-dont/>.

- [5] A. Hindle, Green mining: a methodology of relating software change and configuration to power consumption, *Empirical Software Engineering* **20**(2) (2015), 374–409. doi:10.1007/s10664-013-9276-6.
- [6] B. Aksanli and T.S. Rosing, Human Behavior Aware Energy Management in Residential Cyber-Physical Systems, *IEEE Transactions on Emerging Topics in Computing* **8**(1) (2016), 45–57. doi:10.1109/TETC.2017.2680322.
- [7] C. Zhang, A. Hindle and D.M. German, The Impact of User Choice on Energy Consumption, *IEEE Software* **31**(3) (2014), 69–75. doi:10.1109/MS.2014.27. <https://ieeexplore.ieee.org/document/6756706/>.
- [8] M. Daniel, G. Rivière and N. Couture, CairnFORM, in: *Proceedings of the Thirteenth International Conference on Tangible, Embedded, and Embodied Interaction TEI 2019*, ACM, New York, NY, USA, 2019, pp. 275–286. ISBN 9781450361965. doi:10.1145/3294109.3295634.
- [9] A. Francisco and J.E. Taylor, Understanding citizen perspectives on open urban energy data through the development and testing of a community energy feedback system, *Applied Energy* **256**(September) (2019). doi:10.1016/j.apenergy.2019.113804.
- [10] I.C. Konstantakopoulos, A.R. Barkan, S. He, T. Veeravalli, H. Liu and C. Spanos, A deep learning and gamification approach to improving human-building interaction and energy efficiency in smart infrastructure, *Applied Energy* **237**(September 2018) (2019), 810–821. doi:10.1016/j.apenergy.2018.12.065.
- [11] C.H. Lu, IoT-enabled adaptive context-aware and playful cyber-physical system for everyday energy savings, *IEEE Transactions on Human-Machine Systems* **48**(4) (2018), 380–391. doi:10.1109/THMS.2018.2844119.
- [12] B. Orland, N. Ram, D. Lang, K. Houser, N. Kling and M. Coccia, Saving energy in an office environment: A serious game intervention, *Energy and Buildings* **74**(May) (2014), 43–52. doi:10.1016/j.enbuild.2014.01.036.
- [13] B. Zhou, W. Li, K.W. Chan, Y. Cao, Y. Kuang, X. Liu and X. Wang, Smart home energy management systems: Concept, configurations, and scheduling strategies, *Renewable and Sustainable Energy Reviews* **61** (2016), 30–40. doi:10.1016/j.rser.2016.03.047.
- [14] L. Parolini, N. Tolia, B. Sinopoli and B.H. Krogh, A cyber-physical systems approach to energy management in data centers, in: *Proceedings of the 1st ACM/IEEE International Conference on Cyber-Physical Systems - ICCPS '10*, ACM Press, New York, New York, USA, 2010, p. 168. ISBN 9781450300667. doi:10.1145/1795194.1795218. <http://portal.acm.org/citation.cfm?doi=1795194.1795218>.
- [15] W. Wang, T. Hong, N. Li, R.Q. Wang and J. Chen, Linking energy-cyber-physical systems with occupancy prediction and interpretation through WiFi probe-based ensemble classification, *Applied Energy* **236**(February) (2019), 55–69. doi:10.1016/j.apenergy.2018.11.079.
- [16] L. Sánchez, I. EliceGUI, J. Cuesta and L. Muñoz, On the energy savings achieved through an internet of things enabled smart city trial, in: *Proceedings of the 2014 IEEE International Conference on Communications (ICC)*, IEEE, Sydney, NSW, 2014, pp. 3836–3841. ISBN 9781479920037. doi:10.1109/ICC.2014.6883919.
- [17] Jinsoo Han, Chang-Sic Choi and Ilwoo Lee, More efficient home energy management system based on ZigBee communication and infrared remote controls, *IEEE Transactions on Consumer Electronics* **57**(1) (2011), 85–89. ISBN 9781424487127. doi:10.1109/TCE.2011.5735485. <http://ieeexplore.ieee.org/document/5735485/>.
- [18] K.-s. Choi, Y.-k. Ahn, Y.-c. Park, W.-c. Park, H.-M. Seo, K.-m. Jung and K.-h. Seo, Architectural Design of Home Energy Saving System Based on Realtime Energy-Awareness, in: *Proceedings of the 4th International Conference on Ubiquitous Information Technologies & Applications*, IEEE, Fukuoka, Japan, 2009, pp. 1–5. ISBN 978-1-4244-5131-9. doi:10.1109/ICUT.2009.5405699. <http://ieeexplore.ieee.org/document/5405699/>.
- [19] M. Lin, Y. Pan, L.T. Yang, M. Guo and N. Zheng, Scheduling Co-Design for Reliability and Energy in Cyber-Physical Systems, *IEEE Transactions on Emerging Topics in Computing* **1**(2) (2013), 353–365. doi:10.1109/TETC.2013.2274042. <http://ieeexplore.ieee.org/document/6563161/>.
- [20] Y. Agarwal, S. Savage and R. Gupta, SleepServer: A software-only approach for reducing the energy consumption of PCs within enterprise environments, in: *Proceedings of the 2010 USENIX Annual Technical Conference*, Boston, MA, USA, 2010, pp. 285–299. ISBN 9781931971751.
- [21] L. Parolini, B. Sinopoli, B.H. Krogh and Z. Wang, A cyber-physical systems approach to data center modeling and control for energy efficiency, *Proceedings of the IEEE* **100**(1) (2012), 254–268. doi:10.1109/JPROC.2011.2161244.
- [22] L. Qi, Y. Chen, Y. Yuan, S. Fu, X. Zhang and X. Xu, A QoS-aware virtual machine scheduling method for energy conservation in cloud-based cyber-physical systems, *World Wide Web* **23**(2) (2020), 1275–1297. doi:10.1007/s11280-019-00684-y.
- [23] E.M. Shakshuki, H. Malik and T. Sheltami, WSN in cyber physical systems: Enhanced energy management routing approach using software agents, *Future Generation Computer Systems* **31**(1) (2014), 93–104. doi:10.1016/j.future.2013.03.001. <https://linkinghub.elsevier.com/retrieve/pii/S0167739X13000502>.
- [24] C. Deng, R. Guo, C. Liu, R.Y. Zhong and X. Xu, Data cleansing for energy-saving: a case of Cyber-Physical Machine Tools health monitoring system, *International Journal of Production Research* **56**(1–2) (2018), 1000–1015. doi:10.1080/00207543.2017.1394596.
- [25] J. Leitão, P. Gil, B. Ribeiro and A. Cardoso, Improving household’s efficiency via scheduling of water and energy appliances, in: *Proceedings of the 13th APCA International Conference on Control and Soft Computing, CONTROL*, IEEE, Ponta Delgada, Portugal, 2018, pp. 253–258. ISBN 9781538653463. doi:10.1109/CONTROL.2018.8514291.
- [26] Shyam Sundar Prasad and C. Kumar, An energy efficient and reliable internet of things, in: *Proceedings of the 2012 International Conference on Communication, Information & Computing Technology (ICCICT)*, IEEE, Mumbai, 2012, pp. 1–4. ISBN 978-1-4577-2078-9. doi:10.1109/ICCICT.2012.6398115. <http://ieeexplore.ieee.org/document/6398115/>.
- [27] J.A.E. Isasa, P.G. Larsen and F.O. Hansen, Energy-Aware Model-Driven Development of a Wearable Healthcare Device, in: *Software Engineering in Health Care*, M. Huhn and L. Williams, eds, Springer International Publishing, Cham, 2017, pp. 44–63. ISBN 978-3-319-63194-3.
- [28] M.A.A. Faruque and F. Ahourai, A model-based design of Cyber-Physical Energy Systems, in: *2014 19th Asia and South Pacific Design Automation Conference (ASP-DAC)*, IEEE, Singapore, 2014, pp. 97–104. ISBN 978-1-4799-2816-3. doi:10.1109/ASPDAC.2014.6742873. <http://ieeexplore.ieee.org/document/6742873/>.

- [29] J. Wan, H. Yan, D. Li, K. Zhou and L. Zeng, Cyber-physical systems for optimal energy management scheme of autonomous electric vehicle, *Computer Journal* **56**(8) (2013), 947–956. doi:10.1093/comjnl/bxt043.
- [30] K.T. Park, Y.T. Kang, S.G. Yang, W.B. Zhao, Y.-S. Kang, S.J. Im, D.H. Kim, S.Y. Choi and S. Do Noh, Cyber Physical Energy System for Saving Energy of the Dyeing Process with Industrial Internet of Things and Manufacturing Big Data, *International Journal of Precision Engineering and Manufacturing-Green Technology* **7**(1) (2020), 219–238. ISBN 0123456789. doi:10.1007/s40684-019-00084-7.
- [31] H. Shareef, M.S. Ahmed, A. Mohamed and E. Al Hassan, Review on Home Energy Management System Considering Demand Responses, Smart Technologies, and Intelligent Controllers, *IEEE Access* **6** (2018), 24498–24509. doi:10.1109/ACCESS.2018.2831917. <https://ieeexplore.ieee.org/document/8352822/>.
- [32] C.-K. Tham and T. Luo, Sensing-Driven Energy Purchasing in Smart Grid Cyber-Physical System, *IEEE Transactions on Systems, Man, and Cybernetics: Systems* **43**(4) (2013), 773–784. doi:10.1109/TSMCA.2012.2224337. <http://ieeexplore.ieee.org/document/6457509/>.
- [33] P. Zhao, M.G. Simões and S. Suryanarayanan, A conceptual scheme for cyber-physical systems based energy management in building structures, *2010 9th IEEE/IAS International Conference on Industry Applications, INDUSCON 2010* (2010). ISBN 9781424480104. doi:10.1109/INDUSCON.2010.5739891.
- [34] Wei Wu, M.K. Aziz, Hantao Huang, Hao Yu and H.B. Gooi, A real-time cyber-physical energy management system for smart houses, in: *2011 IEEE PES Innovative Smart Grid Technologies*, IEEE, Perth, WA, Australia, 2011, pp. 1–8. ISBN 978-1-4577-0875-6. doi:10.1109/ISGT-Asia.2011.6167084. <http://ieeexplore.ieee.org/document/6167084/>.
- [35] A.Y. Saber and G.K. Venayagamoorthy, Efficient utilization of renewable energy sources by gridable vehicles in cyber-physical energy systems, *IEEE Systems Journal* **4**(3) (2010), 285–294. doi:10.1109/JSYST.2010.2059212.
- [36] Z. Zhou, B. Wang, M. Dong and K. Ota, Secure and Efficient Vehicle-to-Grid Energy Trading in Cyber Physical Systems: Integration of Blockchain and Edge Computing, *IEEE Transactions on Systems, Man, and Cybernetics: Systems* **50**(1) (2020), 43–57. doi:10.1109/TSMC.2019.2896323. <https://ieeexplore.ieee.org/document/8715389/>.
- [37] J.-M. Horcas, M. Pinto and L. Fuentes, Context-Aware Energy-Efficient Applications for Cyber-Physical Systems, *Ad Hoc Networks* **82**(8) (2019), 15–30. doi:10.1016/j.adhoc.2018.08.004. <https://linkinghub.elsevier.com/retrieve/pii/S1570870518305572>.
- [38] H. Koc and P.P. Madupu, Optimizing energy consumption in cyber physical systems using multiple operating modes, in: *Proceedings of the 2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC)*, IEEE, Las Vegas, NV, USA, 2018, pp. 520–525. ISBN 978-1-5386-4649-6. doi:10.1109/CCWC.2018.8301740. <http://ieeexplore.ieee.org/document/8301740/>.
- [39] S. Baldi, C.D. Korkas, M. Lv and E.B. Kosmatopoulos, Automating occupant-building interaction via smart zoning of thermostatic loads: A switched self-tuning approach, *Applied Energy, Elsevier* **231**(August) (2018), 1246–1258. doi:10.1016/j.apenergy.2018.09.188.
- [40] K. Mason and S. Grijalva, A review of reinforcement learning for autonomous building energy management, *Computers and Electrical Engineering* **78** (2019), 300–312. doi:10.1016/j.compeleceng.2019.07.019.
- [41] L. Yu, S. Qin, M. Zhang, C. Shen, T. Jiang and X. Guan, A Review of Deep Reinforcement Learning for Smart Building Energy Management, *IEEE Internet of Things Journal* **8**(15) (2021), 12046–12063. doi:10.1109/JIOT.2021.3078462.
- [42] X. Lin, P. Bogdan, N. Chang and M. Pedram, Machine learning-based energy management in a hybrid electric vehicle to minimize total operating cost, in: *2015 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, IEEE, Austin TX USA, 2015, pp. 627–634. ISBN 978-1-4673-8388-2. doi:10.1109/ICCAD.2015.7372628. <http://ieeexplore.ieee.org/document/7372628/>.
- [43] M.G. Kibria, M.A. Jarwar, S. Ali, S. Kumar and I. Chong, Web objects based energy efficiency for smart home IoT service provisioning, *International Conference on Ubiquitous and Future Networks, ICUFN* (2017), 55–60. ISBN 9781509047499. doi:10.1109/ICUFN.2017.7993747.
- [44] C. Reinisch, M.J. Kofler and W. Kastner, ThinkHome: A smart home as digital ecosystem, *4th IEEE International Conference on Digital Ecosystems and Technologies - Conference Proceedings of IEEE-DEST 2010, DEST 2010* (2010), 256–261. ISBN 9781424455539. doi:10.1109/DEST.2010.5610636.
- [45] S. Kim and H. Lim, Reinforcement learning based energy management algorithm for smart energy buildings, *Energies* **11**(8) (2018). doi:10.3390/en11082010.
- [46] P. Lissa, C. Deane, M. Schukat, F. Seri, M. Keane and E. Barrett, Deep reinforcement learning for home energy management system control, *Energy and AI* **3** (2021), 100043. doi:10.1016/j.egyai.2020.100043. <https://linkinghub.elsevier.com/retrieve/pii/S2666546820300434>.
- [47] X. Xu, Y. Jia, Y. Xu, Z. Xu, S. Chai and C.S. Lai, A Multi-Agent Reinforcement Learning-Based Data-Driven Method for Home Energy Management, *IEEE Transactions on Smart Grid* **11**(4) (2020), 3201–3211. doi:10.1109/TSG.2020.2971427. <https://ieeexplore.ieee.org/document/8981876/>.
- [48] Y. Liu, D. Zhang and H.B. Gooi, Optimization strategy based on deep reinforcement learning for home energy management, *CSEE Journal of Power and Energy Systems* **6**(3) (2020), 572–582. doi:10.17775/CSEEJPES.2019.02890.
- [49] R. Lu, S.H. Hong and M. Yu, Demand Response for Home Energy Management Using Reinforcement Learning and Artificial Neural Network, *IEEE Transactions on Smart Grid* **10**(6) (2019), 6629–6639. doi:10.1109/TSG.2019.2909266.
- [50] L. Yu, W. Xie, D. Xie, Y. Zou, D. Zhang, Z. Sun, L. Zhang, Y. Zhang and T. Jiang, Deep Reinforcement Learning for Smart Home Energy Management, *IEEE INTERNET OF THINGS JOURNAL* **7**(4) (2019), 2751–2762. doi:10.1109/JIOT.2019.2957289.
- [51] K. Georgiou, S. Xavier-De-Souza and K. Eder, The IoT Energy Challenge: A Software Perspective, *IEEE Embedded Systems Letters* **10**(3) (2018), 53–56. doi:10.1109/LES.2017.2741419.
- [52] H. Kanso, A. Noureddine and E. Exposito, Automated Power Modeling of Computing Devices: Implementation and Use Case for Raspberry Pis, *Sustainable Computing: Informatics and Systems* **37** (2023), 100837. doi:<https://doi.org/10.1016/j.suscom.2022.100837>.
- [53] N. Noy and D. Mcguinness, Ontology Development 101: A Guide to Creating Your First Ontology, *Knowledge Systems Laboratory* **32** (2001).

- [54] M. Compton, P. Barnaghi, L. Bermudez, R. García-Castro, O. Corcho, S. Cox, J. Graybeal, M. Hauswirth, C. Henson, A. Herzog, V. Huang, K. Janowicz, W.D. Kelsey, D. Le Phuoc, L. Lefort, M. Leggieri, H. Neuhaus, A. Nikolov, K. Page, A. Passant, A. Sheth and K. Taylor, The SSN ontology of the W3C semantic sensor network incubator group, *Journal of Web Semantics* **17** (2012), 25–32. doi:10.1016/j.websem.2012.05.003. <https://linkinghub.elsevier.com/retrieve/pii/S1570826812000571>.
- [55] S. Bechhofer, *OWL: Web Ontology Language*, in: *Encyclopedia of Database Systems*, L. LIU and M.T. OZSU, eds, Springer US, Boston, MA, 2009, pp. 2008–2009. ISBN 978-0-387-39940-9. doi:10.1007/978-0-387-39940-9_1073.
- [56] M.A. Musen, The Protégé Project: A Look Back and a Look Forward, *AI Matters* **1**(4) (2015), 4–12-. doi:10.1145/2757001.2757003.
- [57] J.M.G. López, E. Pouresmaeil, C.A. Cañizares, K. Bhattacharya, A. Mosaddegh and B.V. Solanki, Smart Residential Load Simulator for Energy Management in Smart Grids, *IEEE Transactions on Industrial Electronics* **66**(2) (2019), 1443–1452. doi:10.1109/TIE.2018.2818666.
- [58] P. Constantopoulos, F.C. Schweppe and R.C. Larson, Estia: A real-time consumer control scheme for space conditioning usage under spot electricity pricing, *Computers & Operations Research* **18**(8) (1991), 751–765. doi:10.1016/0305-0548(91)90013-H. <https://linkinghub.elsevier.com/retrieve/pii/030505489190013H>.
- [59] L. Yu, W. Xie, D. Xie, Y. Zou, S. Member, D. Zhang, Z. Sun, L. Zhang, Y. Zhang and T. Jiang, Deep Reinforcement Learning for Smart Home Energy Management, *IEEE INTERNET OF THINGS JOURNAL* **7** (2019), 1.
- [60] D. Zhang, S. Li, M. Sun and Z. O’Neill, An Optimal and Learning-Based Demand Response and Home Energy Management System, *IEEE Transactions on Smart Grid* **7**(4) (2016), 1790–1801. doi:10.1109/TSG.2016.2552169. <http://www.ieee.org/publications{ }standards/publications/rights/index.htmlhttp://ieeexplore.ieee.org/document/7450180/>.
- [61] A.A. Thatte and L. Xie, Towards a Unified Operational Value Index of Energy Storage in Smart Grid Environment, *IEEE Transactions on Smart Grid* **3**(3) (2012), 1418–1426. doi:10.1109/TSG.2012.2190114. <http://ieeexplore.ieee.org/document/6191344/>.
- [62] N. Forouzandehmehr, S.M. Perlaza, Z. Han and H.V. Poor, A satisfaction game for heating, ventilation and air conditioning control of smart buildings, *GLOBECOM - IEEE Global Telecommunications Conference* (2013), 3164–3169. ISBN 9781479913534. doi:10.1109/GLOCOM.2013.6831558.
- [63] R. Deng, Z. Zhang, J. Ren and H. Liang, Indoor temperature control of cost-effective smart buildings via real-time smart grid communications, *2016 IEEE Global Communications Conference, GLOBECOM 2016 - Proceedings* (2016), 0–5. ISBN 9781509013289. doi:10.1109/GLOCOM.2016.7841899.
- [64] H.J. Diamond, T.R. Karl, M.A. Palecki, C.B. Baker, J.E. Bell, R.D. Leeper, D.R. Easterling, J.H. Lawrimore, T.P. Meyers, M.R. Helfert, G. Goode and P.W. Thorne, U.S. Climate Reference Network after One Decade of Operations: Status and Assessment, *Bulletin of the American Meteorological Society* **94**(4) (2013), 485–498. doi:10.1175/BAMS-D-12-00170.1.
- [65] M.G. Just, L.M. Nichols and R.R. Dunn, Human indoor climate preferences approximate specific geographies, *Royal Society Open Science* **6**(3) (2019), 180695. doi:10.1098/rsos.180695.