J Intell Robot Syst (2015) 80 (Suppl 1):S331–S354
DOI 10.1007/s10846-014-0169-8

Intelligent Decision-Making for Smart Home Energy Management

Heider Berlink · Nelson Kagan · Anna Helena Reali Costa

Received: 7 March 2014 / Accepted: 1 December 2014 / Published online: 18 December 2014

Abstract One of the goals of Smart Grids is to encourage distributed generation of energy in houses, hence allowing the user to profit by injecting energy into the power grid. The implementation of a differentiated tariff of energy per time of use, coupled with energy storage in batteries, enables profit maximization by the user, who can choose to sell or store the energy generated whenever it is convenient. This paper proposes a solution to the sequential decision-making problem of energy sale by applying reinforcement learning. Results show a significant increase in the total long-term profit by using the policy obtained with the proposed approach, when compared with a price-unaware selling policy.

Keywords SmartHome · SmartGrid · Energy management system · Reinforcement learning

H. Berlink (✉) · N. Kagan · A. H. Reali Costa
Escola Politécnica
Universidade de São Paulo (USP)
São Paulo, SP, Brazil
e-mail: heiderberlink@usp.br

N. Kagan
e-mail: nelsonk@pea.usp.br

A. H. Reali Costa
e-mail: anna.reali@usp.br

1 Introduction

Smart Grids have the potential to lead a revolution in the energy sector, by inserting new measurement, automation and telecommunication technologies into the power grid [1–3]. The implementation of this complex infrastructure produces gains in reliability, efficiency and operational safety. Besides, it can provide new business models arising from the new role of residential and commercial consumers.

The main motivation for the emergence of the Smart Grid concept is the optimization of the power grid use. The development of more efficient ways to use energy has, as a precondition, a more active participation of consumers in the complete power grid energy management. This new scenario suggests management of power supply from the demand side, a concept that is related to the way consumers pay for the energy use and to the possibility of the consumers generate their own energy [4]. *Demand Side Management* (DSM) has two fundamental tools: *Distributed Energy Generation* (electric energy generation in houses and commercial establishments provided by alternative energy sources such as wind and solar) and *Differentiated Tariff* (the electricity price has different values for each time of the day).

The effects of these changes in the distribution grid are under wide discussion, especially considering the market response to the difference in the energy price for residential consumers [5–8]. This represents an important paradigm shift so that consumers will now

contribute directly to the insertion of energy into the grid. Besides, if we consider that energy generation can exceed energy consumption in a given period, the consumer can still sell the surplus in the energy market, profiting from it. In this sense, considering the variation in energy price, the consumer must decide when and how much to sell of this extra energy, maximizing his/her profit.

As the generation through alternative sources is not constant during the day and the consumer can store energy using storage devices, we can see that the decision-making problem involving the sale of this extra energy has a complex dynamics. Thus, it is necessary to develop an autonomous decision-making system that identifies the current energy generation and the price tendency in the energy market, and chooses to sell or to store the surplus energy, so as to maximize the user long-term profit.

A Decision-Making Support System (DMSS) [9] is defined as a computer-based system that helps users to achieve specific goals in individual or organizational decision-making processes. In general, DMSSs help the management, planning and operational levels, depending on the user degrees of freedom. DMSSs have found application in many areas, e.g., clinical decision support for medical diagnosis [10], business intelligence applications [11], agricultural production [12], among others.

DMSS has evolved to Intelligent DMSS (IDMSS), in which Artificial Intelligence concepts improve the DMSS robustness [13, 14]. As a result, a new set of applications emerged in different areas, increasing the importance of those systems to solve optimization problems that are difficult to humans.

In this paper, we model the distributed energy selling problem as a Markov Decision Process. We propose a Reinforcement Learning-based IDMSS in order to obtain the optimal power selling policy for a given sequence of prices from the energy market. Tests evaluate the algorithm performance and its ability to identify price trends. Results were compared with a *Naïve-greedy* policy that represents a price-unaware selling policy. The developed system was applied to two different cases, in which we studied the North-American and the Brazilian energy markets.

The structure of the paper is as follows. Section 2 provides the literature review about the application of IDMSS to energy managment in Smart Homes. Section 3 states the problem formally, describing each subsystem considered in the proposed solution. Section 4 covers the mathematical framework and the theoretical concepts used in the paper. Section 5 explains our proposal: the modes of operation, models and algorithms. Section 6 explains the methodology used for training and testing and shows the results achieved in two case studies. Finally, Section 7 concludes the paper by highlighting the main contributions and suggestions for future work.

2 Intelligent Decision-Making in Smart Homes

Intelligent Decision-Making Support Systems have appeared as an evolution of traditional Decision-Making Support Systems. The original concept defines it as a computer-based system that is able to make rational decisions in the same way humans do. An IDMSS should have the ability to deal with uncertainties and to analyze situations in order to identify and to diagnose problems, proposing a set of actions to achieve the user goals with greater effectiveness.

These support systems must optimize the system operation in an autonomous and flexible way, being responsible for receiving information from the environment through sensors, for choosing the best action considering a specific goal and for applying it through actuators. Our goal here is to propose an IDMSS that operates as an Energy Management System (EMS), allowing the dynamic decision of selling or storing electric energy in response to price signals for Smart Homes. This IDMSS is called Reinforcement Learning-based EMS (RLbEMS).

The concept involving the development of a Smart Home can be seen in Fig. 1, where it is possible to identify the subsystems that compose it. The EMS must optimize the house energetic balance, so as to minimize the user energy bill (or to maximize the user profit) in a given period. The system must meet the user demand with the most convenient source of energy, taking into consideration the economic signals from the energy market. When there is a surplus of energy because of low consumption, the EMS must sell it to the utility and make profit. Clearly, this problem can be viewed as a sequential decision-making problem, in which a decision-maker must, at each moment, choose the best source of energy considering the level of generation, the level of storage, the user demand and the price signal.

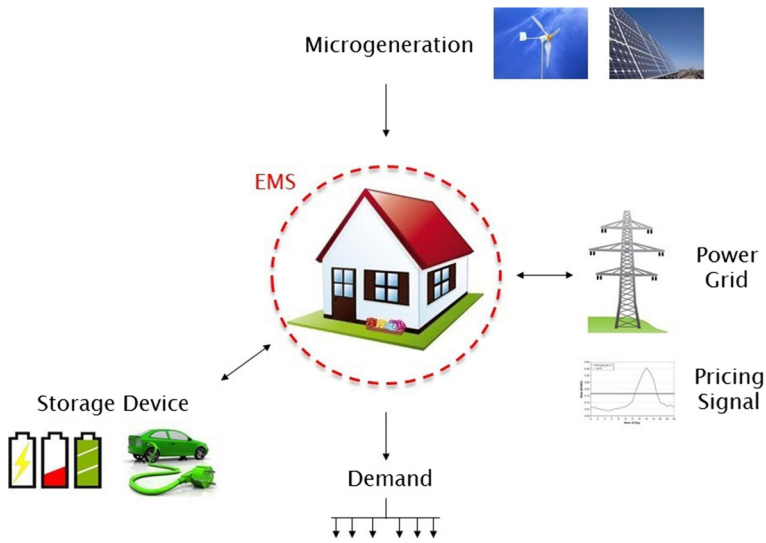

Fig. 1 Smart Home Scheme: the home receives energy from the power grid and from its own microgeneration system; this energy is used to meet the home demand or it can be or sold or stored for future use. All the decisions in a Smart Home are made by the EMS

The EMS must have enough information about the environment so that it can achieve its goals. The complete Smart Home system involves stochastic and deterministic subsystems. As stochastic subsystems we can mention the generation using alternative renewable sources that depend on the weather conditions; the real-time energy price that depends on the energy market, and the user demand that depends on the user behavior. As deterministic subsystems, we can mention the storage system and the power grid that maintain its characteristics during the operation of the system, i.e., the storage maximum capacity remains always fixed and we consider that the energy from the power grid is always available for use. Because of the high level of uncertainty involved in the process and the lack of information about the dynamics of the Smart Home subsystems, we propose a learning-based EMS, making the system adaptable to variations in the environment dynamics.

In the computing and power systems community, many researchers have developed optimization algorithms to deal with the energy management of Smart Homes. The integration of the local energy management in houses and the global energy management of the power distribution grid can be analyzed under different perspectives.

Dusparic et al. [15] propose a multi-agent approach that uses predicted energy consumption data to promote a combined demand response , reducing the consumption of a group of houses in the peak hours. In their work, each house has a reinforcement learning agent that controls the energy consumption of electrical devices in a household, taking into consideration current and predicted energy prices. The renewable sources are integrated into the model in an indirect way by decreasing or increasing the predicted prices when there is more or less availability of these sources, i.e., their work does not consider the insertion of generation data from a real system. The authors implemented a multi-policy strategy called W-learning, that consists of integrating independent Q-learning agents, one for each house. During the operation, the state of each house is observed by its agent and the action to be performed in each house is chosen. Then, the immediate reward is estimated (see the scheme in Fig. 2). An agent is chosen to actually apply its action to the system. The agent chosen is the one that observed the maximum reward. The agents are trained to consume energy during off-peak predicted price times, resulting in a global and effective demand response.

In contrast, O'Neill et al. [16] present an algorithm called CAES, based on a single reinforcement learning agent that controls the operation of the home appliances to reduce the energy costs and smooth the energy usage. CAES is an online learning application that implicitly estimates the impact of future energy prices and consumer decisions on long-term costs and schedules residential appliances usage, as shown in Fig. 3. This schedule delays the use of each appliance, so as to operate each one when the energy price is lower. Their MDP model considers the current energy

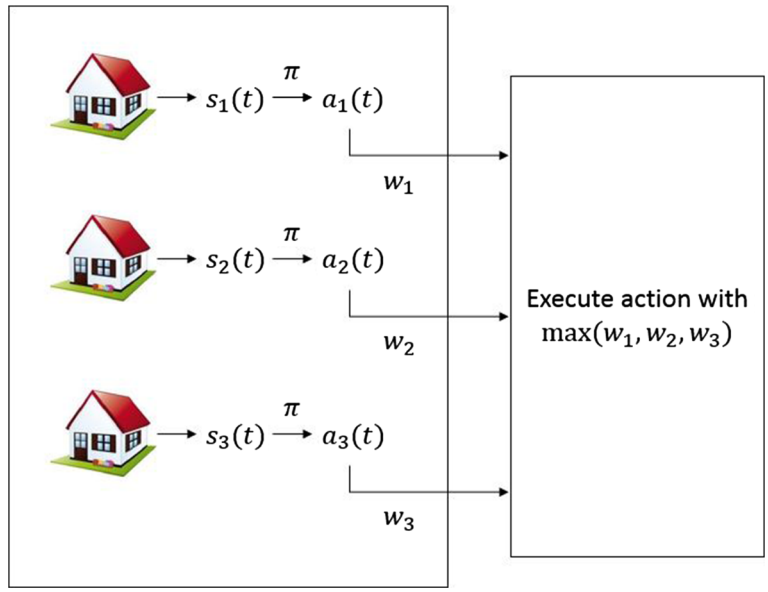

Fig. 2 W-Learning, methodology used by [15] to implement a multi-agent approach based on reinforcement learning independent agents

consumption, the delay time for each appliance and the current energy price. The system does not consider microgeneration and storage devices. The objective of the agent is to operate the appliances when the price is low and with minimum delay. Results show that CAES reduces costs up to 40% with respect to a price-unaware energy allocation.

Many proposals combine different optimization techniques to provide a fast and robust solution. Xiaodao et al. [17] combine Linear Programming (LP) and Monte Carlo Simulation (MC) in a house with solar generation, storage device and load management. In their work, the energy management is modeled as a LP problem that has as output the usage schedule of a set of appliances in order to minimize the energy cost for the user. The proposed algorithm takes into account the uncertainties in household appliance operation time and the intermittency of renewable generation, proposing a rigid scheduling for the use of appliances that is calculated by using information of the previous day. During the operation, this schedule is updated by an on-line adjustment that considers the current price and generation data. Their approach has, as its main advantage, the fast solution provided by the LP. However, the modeling is a simple approximation of the real system, which must be evaluated carefully given the system high level of uncertainty. Besides, the obtained schedule considers the generation and price data from the previous day, i.e., the solution is not feasible when the user must deal with an energy real-time pricing model. One of the disadvantages of this proposal is the rigid schedule for the appliances use, which we do not consider feasible for real applications. The scheduling restrictions are treated in a more flexible way by O'Neill et al. [16], because the operation of CAES makes a reservation of energy anytime the user wants to operate any appliance. CAES looks for the best time to operate the appliance, aiming at minimizing the energy costs. In contrast, our solution does not involve a rigid scheduling for the usage of the consumer appliances because we think it is important to keep the user freedom in decisions about energy consumption.

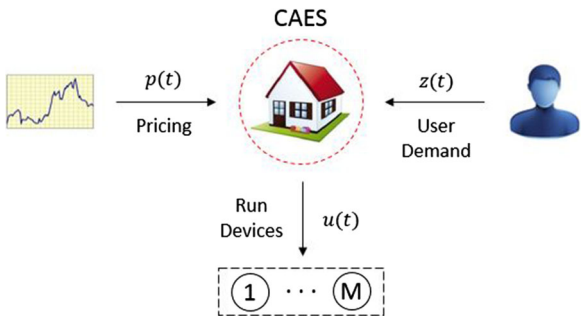

Fig. 3 System implemented by [16] to promote a reinforcement learning based demand response for a single house. This system receives price data and user demand information to schedule the appliances energy usage

Other algorithms apply usual optimization strategies. For instance, Mohsenian-Rad et al. [18] present

a distributed demand-side energy management system among users that uses the game theory (GT) and formulates an energy consumption scheduling game, where the players are the users and their strategies are the daily schedules of their household appliances and loads. Their work considers a scenario where a source of energy is shared by several users, each one equipped with an energy consumption scheduler that is installed in the smart-meters, as shown in Fig. 4. The optimization objective is to minimize the energy cost in the complete system. They show that for a usual scenario, with a single utility company serving multiple customers, the optimal global performance in terms of minimizing energy costs is achieved at the Nash equilibrium of the formulated energy consumption scheduling game.

Game theory is also used by Atzeni et al. [19], whose work considers a day-ahead optimization process regulated by an independent central unit. However, differently from Ref. [18], the solution considers distributed storage and energy generation by alternative sources in the houses. Here, the users are classified as passive or active consumers, where the active are those which can insert energy into the grid. The optimization problem is formulated as a noncooperative game and the existence of optimal strategies is analyzed.

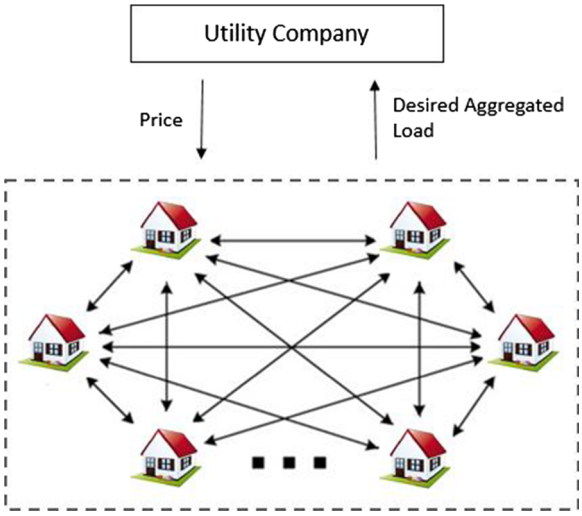

Fig. 4 Scheme of the game theoretic solution proposed by [18]. A single energy source is shared by a group of users, that respond to a differentiated energy price promoting a combined demand response

These works consider different ways of dealing with the energy management problem. A comparison between the main points of these efforts can be viewed in Table 1. We can also observe in the table the differences between our proposal and other ones. Each approach considers sets of subsystems depicted in Fig. 1 to provide demand response, and it is difficult to compare quantitatively the solutions proposed in the literature, because each one is very specific to each configuration of the problem.

We consider microgeneration and storage to provide a demand response, while we do not consider the energy consumption in our model. One of the disadvantages of proposing a rigid schedule for the appliances usage is the feasibility of this methodology for real applications. The scheduling restrictions are treated in a more flexible way by O'Neill et al. [16], because the operation of CAES reserves energy anytime the user wants to operate any appliance. CAES looks for the best time to operate the appliance, aiming at minimizing energy costs.

Considering this, our solution does not involve a rigid scheduling for the energy usage. We propose a demand response without changing the energy consumption profile, which is an advantage for the user. Our control strategy coordinates the energy generation and storage operation in response to the price signal, promoting an efficient response to demand while ensuring the comfort of the user in deciding when to use his/her appliances.

The solution using reinforcement learning requires modeling the problem as a sequential decision process. One point to be discussed is how the price information is inserted into the model, considering that the energy price varies considerably during the day. This may increase the problem state space. We propose an innovative way to enter price information into the model as trend indexes, which represents a significant gain regarding the resulting reduced space of states and the amount of information available to be considered in the model. Many works use only the absolute value of the price in their models, not considering the real variation of the energy price [16, 17]. We tested our approach in two different pricing models in order to assess its performance in scenarios with different levels of uncertainty in energy prices.

Table 1 Comparison among related work

Reference	Dif. Tariff	Microg.	Storage	Demand	Control Strategy	Solution
Dusparic et al. [15]	RTP	No.	Yes.	Yes.	Aggregated load control	RL(Multi Agent)
O'Neill et al. [16]	RTP	No.	No.	Yes.	Schedule of appliances	RL(Single Agent)
Xiaodao et al. [17]	DA-RTP	Yes.	Yes.	Yes.	Schedule of appliances	LP + MC
Mohsenian et al. [18]	RTP	No.	No.	Yes.	Aggregated load control	GT
Atzeni et al. [19]	DA-RTP	Yes.	Yes.	Yes.	Aggregated energy balance	GT
Our Proposal	RTP+TOU	Yes.	Yes.	No.	Sell or Store energy	RL(Single Agent)

RTP: Real-Time Pricing; DA-RTP: Day-Ahead RTP; TOU: Time-Of-Use Pricing;
RL: Reinforcement Learning; LP: Linear Programming; MC: Monte Carlo; GT: Game Theory

In the following sections, the problem studied in this work is stated in a more precise and detailed way. The theoretical background needed is also presented.

3 Problem Statement

Figure 1 shows a scheme of a Smart Home EMS that makes decisions based on an energy production model, an energy storage model and an energy pricing model. The objective of this section is to discuss the models involved in the Smart Home EMS and to introduce the adopted models for each of the subsystems.

3.1 Residential Demand Response

Demand Response is commonly defined as changes in energy usage by end-use customers from their normal consumption patterns in response to changes in the price of electricity over time, or to incentives through payments in order to induce lower electricity use at times of high wholesale market prices or when system reliability is jeopardized [20]. This concept can be applied in a passive or active way. Passive consumers are those who just change their consumption pattern by modifying the amount and time of energy use from the power grid; active consumers are those who, besides changing the amount and time of energy use, insert energy into the power grid following a price signal. This concept is only possible because of the Smart-Metering and Information Technology infrastructure implemented by Smart Grids [21].

The problem investigated herein is a particular case of the complete problem shown in Fig. 1. We consider a house that has its own power generation through alternative sources and that has a storage device. The generated power can be sold directly to the energy market at the current price or can be placed in the storage device to be sold at a convenient future time. All the energy generated is available to sell, i.e., the power consumption of the house (Energy demand in Fig. 1) is not taken into consideration. The solution we propose keeps the user freedom in decisions about energy consumption, i.e., we promote a demand response without changing its consumption profile. This is a possible way to treat the problem while trying to avoid the difficulty in obtaining real consumption data to be considered in the solution. This assumption does not invalidate the results obtained and, moreover, yield advantage to users, since the results do not take into account the change of his/her way of consuming energy. However, as discussed in Section 5, the power consumption by the user, once defined, can be subtracted from the total energy available. We model a decision process that does not consider any interference in the consumption profile of the user. We also analyze the relation between the generation intermittency of alternative sources and the dynamics of the energy price variation, besides its influence on the decision to sell or to store energy.

3.2 Energy Production Model

Alternative methods of generating energy through renewable sources differ in many ways from conventional methods of energy production. As conventional methods, the most commonly used are those using fossil fuels (Thermoelectric) or the kinetic energy of water (Hydroelectric). The alternative ways of producing electricity have a sustainable appeal and

have as fundamental characteristic the low environmental impact. However, they have low capacity of production compared with conventional methods.

Considering the alternative methods available today, solar and wind energy are more suitable for residential systems due to their easy implementation. These energy sources have in common a strong dependence on specific weather conditions for generation, which varies considerably during the day. For residential systems, usually we have a peak of generation in the middle of the day for solar technology and a completely variable generation profile for wind technology. One of the main purposes of an EMS is to deal with this intermittence, improving the energy use.

In this work, the generation of electricity is modeled considering a typical profile observed in a Solar Photovoltaic Generation System (Solar PV System). We consider an array with the generation data from the Solar PV System as an input for the EMS (Microgeneration in Fig. 1). The values of generation were discretized and for each step packages of generated energy corresponding to 200Wh were considered. Technical characteristics such as losses and the process efficiency are not considered in the problem. These features will be considered in future works.

The analysis presented here was performed for two different places: USA and Brazil. Considering the strong climate dependence of the generation through alternative sources, a different profile of generation must be used for each case. For the first case study in the USA, we used the typical profile shown in Fig. 5. For the second case study, we used treated and discretized data from a real solar power plant located in São Paulo, Brazil, as can be seen in Fig. 6.

3.3 Energy Storage Model

Energy can be stored in a storage device to be sold when the price is high, for example. In particular, energy generated by the Solar PV System can also be stored, increasing the user profit. The storage device is one of the most important elements of Smart Homes, and, despite of the growing application in houses nowadays, it is still the most expensive one [22, 23]. Considering that, special care must be taken in the specification of this device, otherwise profit from the sale of energy will not make up for the investment.

In general, we consider as residential storage devices the rechargeable batteries and electric vehicles. The main difference between them is the availability, because while the battery is always available for use, electric vehicles will not be available outside

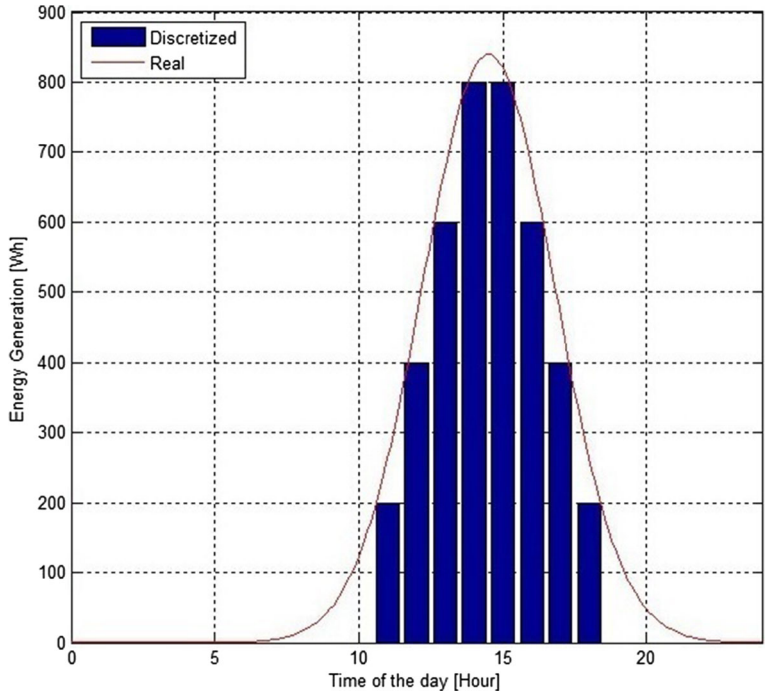

Fig. 5 Solar photovoltaic generation profile for USA[17]

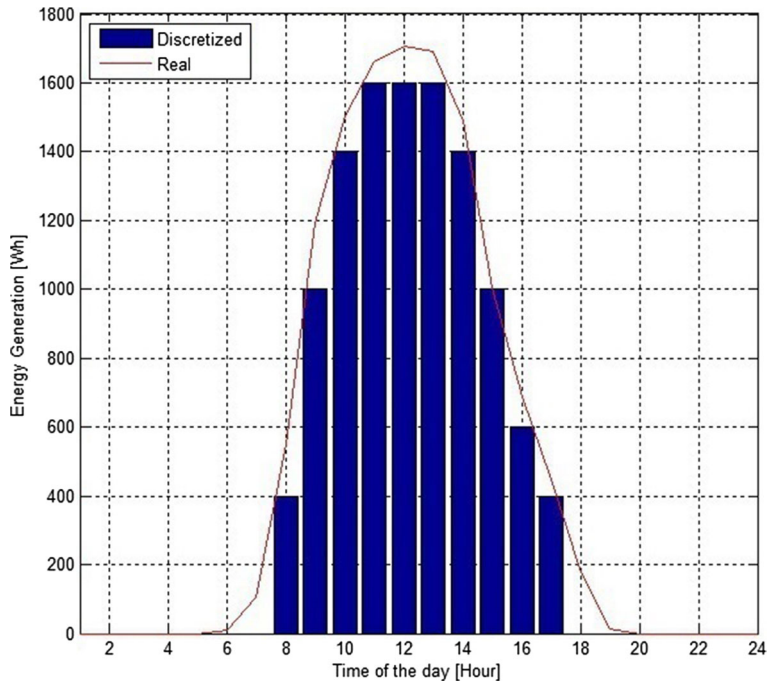

Fig. 6 Solar photovoltaic generation profile for Brazil

the house. For this paper, we consider solely a rechargeable battery as a storage device.

The main characteristics of this device are the cost to store energy, charge and discharge efficiency, maximum charging and discharging rate, rate of energy loss over time and maximum storage capacity. To simplify the problem, the model used here considers only the maximum storage capacity of the battery, because this characteristic directly affects the solution of the problem and it is sufficient for the proposed analyses. The battery is represented as a discretized variable, $B(t)$, limited by the maximum storage capacity, B_{MAX}, that increases its value when the EMS chooses not to sell the generated energy. The values of $B(t)$ belong to the set $\{0, 1, 2, ..., B_{MAX}\}$, where $B(t) = 0$ means that there is no energy in the battery and $B(t) = B_{MAX}$ means that the battery is full. The battery stores energy in packages of energy, with the same discretization used for the energy generated by the Solar PV System.

The maximum storage capacity of the battery has impact on the agent degree of freedom to choose actions. This increases when we use storage devices that have higher capacities. Note that there are always more admissible actions to choose in each state of operation when batteries with higher capacities are used. However, the increase of battery capacity makes the system more expensive. We also evaluated the influence of the Maximum Storage Capacity, B_{MAX} on the profit obtained by the EMS.

3.4 Energy Pricing Model

Here we characterize the energy market by sales prices of energy. In general, the way prices vary over time should be related to local energy demand, i.e., price is high when there is intense use of energy and, in contrast, it is low when there are few consumers using it. This financial motivation encourages consumers to change their consumption pattern, relieving the electric grid when there is great demand.

The data used in the first case study were obtained from an on-line database that contains the Energy Local Marginal Price for the District of Columbia, USA [24]. These data represent the energy price variation of a real market, i.e., the algorithm developed and the results take into account real price variation for a given location. This location uses as price model the *Real-Time Pricing* (RTP), i.e., the energy price varies hourly considering many different factors, such as current demand, availability of energy, among others [21]. A variation of the RTP model is the Day-Ahead RTP (DA-RTP), that differs from RTP because, each day, the user receives the previous day pricing profile.

A curve of price for the winter of 2009 in Columbia, USA can be seen in Fig. 7.

These data correspond to the hourly energy-selling prices for five consecutive years. Before getting into the algorithm, the data were mapped as indexes that represent the price trend over a time window, as explained in Section 5.1.1. It is important to mention that the energy sold by the user is very small compared to that of the market, i.e., the amount of energy sold at each instant does not affect the energy price.

For the second case study, we consider the differenciated tariff recently implemented in Brazil, called *White-tariff*. In this case, there are three values of tariff during the day and these values remain the same for a given period of time, what is called *Time-of-use tariff* (TOU) [25]. The values of tariff used in this case study can be viewed in Fig. 8.

The profit, $PR(t)$, made from the sale of energy is simply the money raised from the sale, computed at each moment as the actual price of energy multiplied by the amount of energy sold.

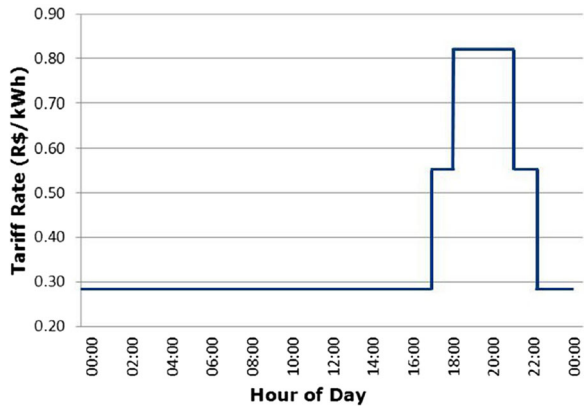

Fig. 8 Brazilian Time-of-use tariff. [25]

4 Background

We investigate the problem of getting the most profit with a policy that decides to store or to sell energy, according to the intermittency of alternative energy generation and energy price in the market. This problem is a sequential decision-making problem whose solution maximizes long-term gain of the decision maker.

A sequential decision-making problem can be modelled as a Markov Decision Process [26] and its solution can be obtained through the Reinforcement Learning Q-Learning algorithm [27]. We describe both concepts in the next subsections.

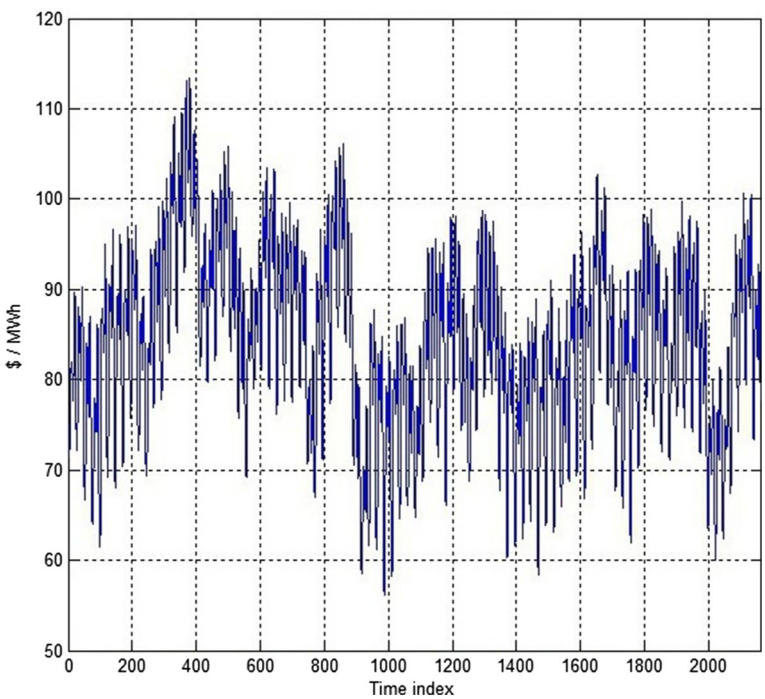

Fig. 7 Energy Price for the winter of 2009 in the USA.[25]

4.1 Markov Decision Process

A sequential decision-making problem is characterized as a Markov Decision Process (MDP) if the environment is observable and evolves probabilistically according to a finite and discrete set of states. Besides, for each state there is a finite set of possible actions to be executed. In general, at each step of the process evolution, states are observed, actions are performed and reinforcements are collected, as shown in Fig. 9.

In a formal way, we can define a MDP as a quadruple $< S, A, T, R >$, where [26]:

- S is a finite set of states;
- A is a finite set of possible actions;
- $T : S \times A \times S \rightarrow [0, 1]$ is a state transition probability function, which defines the transition probability from a state $s(t) \in S$ to a state $s(t + 1) \in S$ when an action $a(t) \in A$ is applied in $s(t)$;
- $R : S \times A \rightarrow \Re$ is the reward function. Hence, for each state transition $r(t) = R(s(t), a(t))$;

We define $A_s(s)$ as the admissible set of actions for a state $s \in S$. Considering $i \in S$ the current state, the transition from $i \in S$ to $j \in S$ in response to the application of the action $a \in A_s(i)$ will occur with probability $T(i, a, j)$ and a reward $R(i, a)$ will be received.

Thus, at each step, the decision-maker observes the current state $s(t)$ of the environment and must choose the best action to perform. After executing action $a(t)$, it receives a reward $r(t)$ and the environment evolves to the next state $s(t + 1)$, which will be observed in the next step. The probability of transition from a current state to another future state depends only on the current state and the action taken at this instant. This is known as the Markov Property [27]. The reward is a measure of the value of the applied action, considering the goals of the optimization problem.

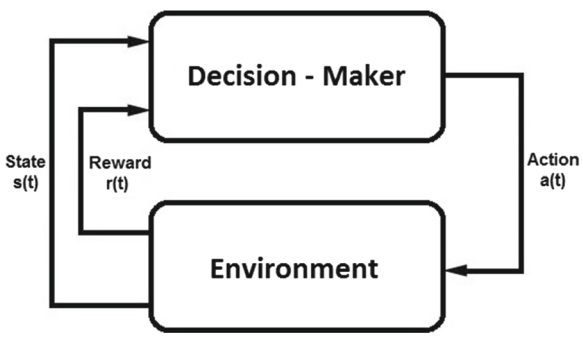

Fig. 9 The decision-maker interaction with the environment

Solving a MDP means finding a policy π that specifies which action must be executed in each state, considering the maximization of the discounted cumulative rewards during an infinite time horizon.

Considering a policy π and an initial state s_0, we can define *Value Function* as the expected cumulative reward obtained from the application of the policy π until the end of the time horizon. Mathematically, we have:

$$V^{\pi}(s) = E\left[\sum_{t=0}^{\infty} \gamma^t r(t) | \pi, s_0 = s\right], \forall s \epsilon S, \qquad (1)$$

where γ, $0 < \gamma \leq 1$, is a constant called discount factor, that ponders the reward received over time, and $r(t)$ is the reward received in each step t. The optimal policy π^* is the one that maximizes the value function for each state, which corresponds to the optimal value function V^*. Considering real problems, solving a MDP is related to finding the policy that is the best approximation of the optimal policy.

There are many ways of solving a MDP, depending on the problem studied and the available information about the system. Considering real problems, sometimes it is difficult to obtain a complete system model as specified above. In these cases, learning algorithms are needed, because the agent must decide the best action to perform in each state while learning about the state probability function or the reward function that sometimes are unknown. One way to solve this problem is applying a Reinforcement Learning algorithm [27], which is based on acquiring knowledge through interactions with the environment.

4.2 Reinforcement Learning

Learning techniques are characterized by the modification of the decision-making mechanism aiming at improving the performance of the control system. Those techniques are often used in problems whose the explored environments are unknown. Reinforcement learning (RL) is an experimentation-based

learning technique, i.e., the knowledge about the environment is acquired from the responses to the actions taken in each state. Thus, the main goal is to find out which actions should be applied in each state of the system to maximize the expected long-term reward (or reinforcement) while learning about the environment.

RL is one way of solving a MDP in problems when the agent has little information about the controlled system [27]. Q-Learning [28] was the first RL algorithm obtained to perform an experimentation-based learning, and ensures that an optimal policy is found if the environment can be modeled as a MDP and if the agent explores the environment in a way that never completely ignore any state or action.

The Q-Learning algorithm is based on the *Value-Action Function*, defined as:

$$Q(s(t), a(t)) = R(s(t), a(t)) + \gamma . V^*(s(t+1)), \quad (2)$$

where $Q : S \times A \rightarrow \Re$ is a real function that expresses the expected value of the future rewards, $Q(.)$, when an action $a(t)$ is applied to a state $s(t)$.

The core of the algorithm is based on a recursive equation that is responsible for the estimation of the final value of $Q(s, a)$ for each pair state-action. The value of $Q(s, a)$ is updated at each step considering the following equations:

$$Q_{New}(s(t), a(t)) \leftarrow \left[r(t) + \gamma \max_a Q(s(t+1), a) \right], \quad (3)$$

$$Q(s(t), a(t)) \leftarrow (1 - \alpha) Q + \alpha Q_{New}(s(t), a(t)), \quad (4)$$

where $s(t)$ and $a(t)$ are the observed state and the applied action on the step t, respectively; $r(t)$ is the received reward after applying $a(t)$ to $s(t)$; α is a constant value called learning rate that varies from 0 to 1 and that reflects how the agent will ponder the new and the old information of $Q(s(t), a(t))$; and γ is the discount factor that reflects the importance given to future rewards.

The main goal of the Q-Learning algorithm is to estimate the values of $Q(s, a)$ exhaustively until its values stabilize and converge to the optimal value. In the algorithm, $Q_{S \times A}$ is defined as a matrix with the number of rows and columns equal to the number of states, $|S|$, and actions, $|A|$, respectively. Q-Learning is described in Algorithm 1.

Algorithm 1 Q-Learning

Require: Initialize the table entry $Q_{S \times A}$ to zero.
Observe the current state $s(t)$
while TRUE **do**
Select an action $a(t)$ and execute it
Receive immediate reward $r(t)$
Observe the new state $s(t+1)$
Update the table entry for $Q(s(t), a(t))$ (Eq. 4)
$s(t) \leftarrow s(t+1)$
end while

After all, the optimal policy is obtained by choosing from matrix $Q_{S \times A}$ which actions maximize the final value-action for each state. This procedure can be matematically defined as:

$$\pi^*(s) = \arg\max_{a \epsilon A} Q(s, a), \forall s \epsilon S. \quad (5)$$

For each state, there is an action that, if applied, will maximize the rewards received in the considered horizon of actuation.

5 Reinforcement Learning-Based EMS for Smart Homes

In this section we describe the complete architecture of our proposed Reinforcement Learning-based EMS for Smart Homes (RLbEMS), as well as how we have modeled the energy selling problem as a MDP and how RLbEMS learns a selling policy using the Q-Learning algorithm. Finally, we describe how RLbEMS applies the policy learned when new data both from energy generation and from energy price are available.

RLbEMS consists of two modes:

– **Training mode:** This mode corresponds to a preliminary off-line step of actuation, when the system learns a policy. In the training mode, the system receives historical price data from the energy market, historical generation data from the Solar PV System and the system characteristics, as can be viewed in Fig. 10. After receiving this data set, RLbEMS learns a policy of actuation using the Q-Learning algorithm, i.e., a selling action is defined for each state of the environment. At this step, the system uses Algorithm 2. It is important to

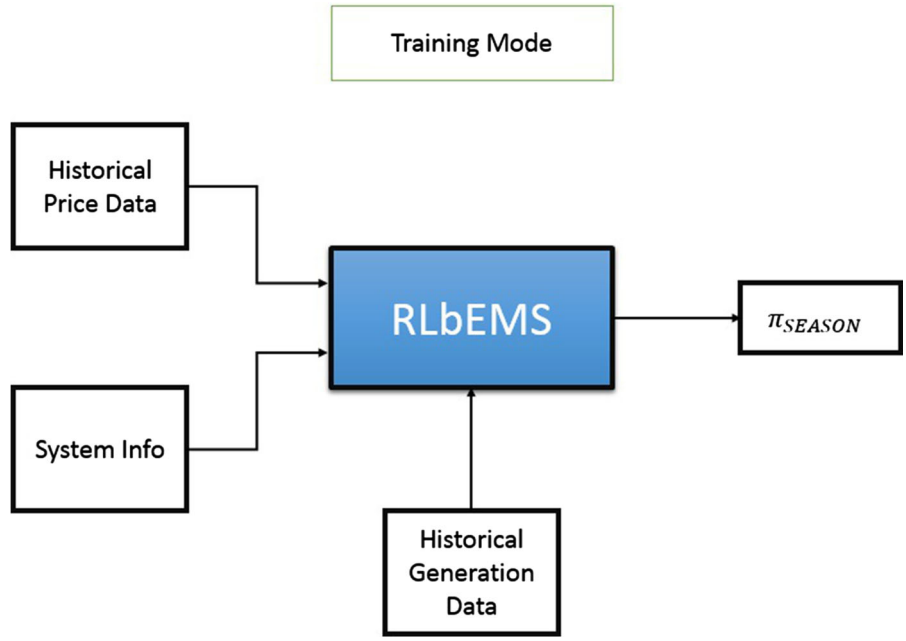

Fig. 10 RLbEMS: Training mode

mention that the system is designed to learn a policy for each season of the year; the output of the training mode is hence a set of four policies, each one related to a specific season.

– **Operation mode:** After running the training mode, the system is able to operate at runtime. This mode corresponds to the actual operation of the system, when information about current prices and power generation data are observed. The system performs the best action learned to the situation observed, i.e., here the system only applies the policy learned in the training mode. In the operation mode, the system also receives information about the current season, in order to apply the correct policy learned. Here, the system uses Algorithm 3, that also calculates the profit made by using the energy-selling policy. The scheme of this mode of operation is shown in Fig. 11.

The right policies are learned for each season, based on historical price and power generation data. We model the process as a MDP and use the Q-Learning algorithm to learn policies for each season, in the training mode of the system. The proposed model and learning approach are described in the following sections. Finally, the operation mode of the system is described in greater detail.

5.1 EMS for Smart Homes as a MDP

We consider a single residence that has its own energy generated by a Solar PV System and that receives the pricing signal in real-time. Besides, this residence has batteries to store energy whenever it is convenient, with B_{MAX} as storage capacity.

The system has, as priority, selling the generated energy at each instant; it is not possible to sell the stored energy before selling the generated energy. In addition, at each instant, the EMS must not sell more than the total available amount of energy (stored and generated energy) and must respect the battery maximum capacity of storage.

In our MDP model, the state transition probability function is unknown. That is the main reason for applying a RL technique to find an optimal policy for each season, in the training mode of RLbEMS. Next, we provide the details of the MDP modeling.

5.1.1 States

States should reflect the main features of the system, which influence the solution of the problem. As stated earlier, a MDP is characterized by a fully observable environment; it must be possible to obtain the desired information about the system at any instant.

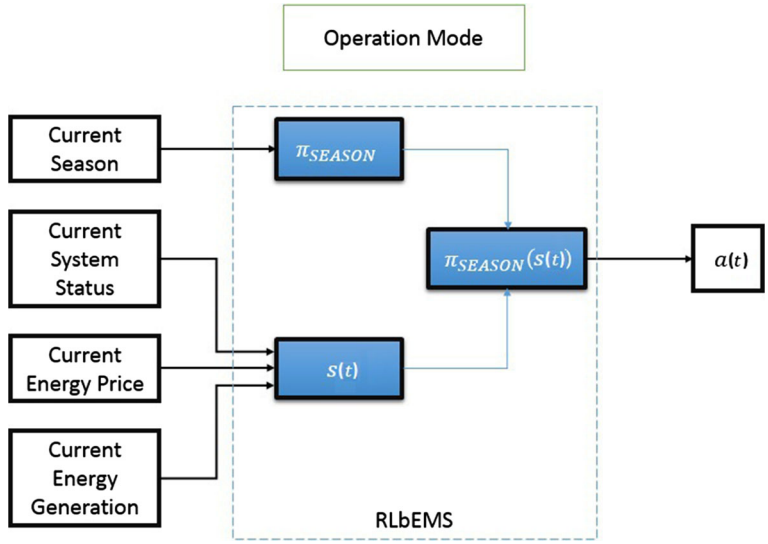

Fig. 11 RLbEMS: Operation mode

We choose to define the states with completely observable and fundamental information about the problem solution: the amount of energy stored in the battery, the amount of energy generated by the Solar PV System and information about the price trend. The information about the battery storage level and the generated energy are just the absolute value of each variable at each instant; on the other hand, the information about the price trend is composed of two indexes that represent the price trend and the price average level in a given time window.

Pricing signals are inserted on the EMS as a pricing sequence $p(t) \in \Re_+$, and represents the amount of money received by the consumer for selling a unit of energy at a given discrete time t. Considering a time window of three steps, we define *Price interval*, $\Delta p(t)$, as a n array composed of the price at instant t and the prices of two instants earlier. Therefore:

$$\Delta p(t) = [p(t-2),\, p(t-1),\, p(t)]. \tag{6}$$

For each $\Delta p(t)$, the price average value, $\overline{\Delta p(t)}$, is defined as the average price at the interval. The price trend, $\overrightarrow{p(t)}$, is defined considering the price incremental at the interval, as can be seen in Fig. 12. Working with price trend and average value at the interval

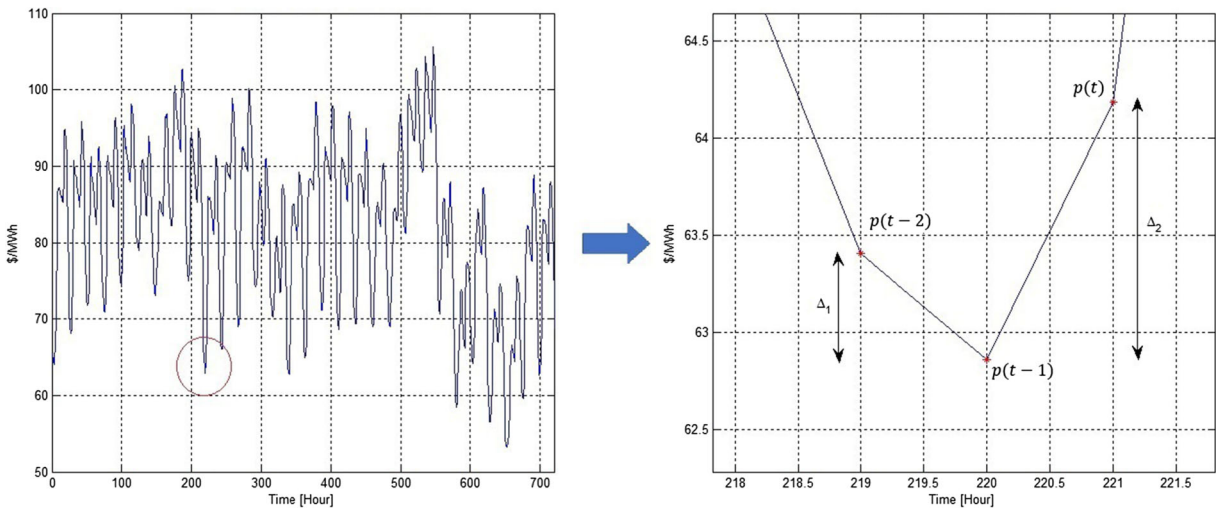

Fig. 12 Determination of the price intervals to calculate the price trend

Table 2 Price trend index

Price Trend Index	$\overrightarrow{p(t)}$	$\overrightarrow{p(t)}_{IND}$
$\Delta_1 \geq 0, \Delta_2 > 0$	$\lvert\Delta_2\rvert \geq \lvert\Delta_1\rvert$	8
$\Delta_1 \geq 0, \Delta_2 > 0$	$\lvert\Delta_2\rvert < \lvert\Delta_1\rvert$	7
$\Delta_1 < 0, \Delta_2 \geq 0$	$\lvert\Delta_2\rvert > \lvert\Delta_1\rvert$	6
$\Delta_1 < 0, \Delta_2 \geq 0$	$\lvert\Delta_2\rvert < \lvert\Delta_1\rvert$	5
$\Delta_1 \geq 0, \Delta_2 \leq 0$	$\lvert\Delta_2\rvert \leq \lvert\Delta_1\rvert$	4
$\Delta_1 \geq 0, \Delta_2 \leq 0$	$\lvert\Delta_2\rvert > \lvert\Delta_1\rvert$	3
$\Delta_1 \leq 0, \Delta_2 < 0$	$\lvert\Delta_2\rvert < \lvert\Delta_1\rvert$	2
$\Delta_1 \leq 0, \Delta_2 < 0$	$\lvert\Delta_2\rvert \geq \lvert\Delta_1\rvert$	1

seems to be a good practical approximation considering the Markov Property described in Section 4.1. Besides, it provides a smaller state space in comparison to the one obtained by using the absolute price value. We calculate the price variation as:

$$\Delta_2 = p(t) - p(t-1), \tag{7}$$

$$\Delta_1 = p(t-1) - p(t-2). \tag{8}$$

These variables were mapped to the indexes $\overrightarrow{p(t)}_{IND}$ and $\overline{\Delta p(t)}_{IND}$ as can be viewed in Tables 2 and 3.

The system state is established as an array composed of four variables: the power stored in the battery in the current moment, $B(t)$, the power generated by the Solar PV System at the current instant, $G(t)$, and the indexes of price trend, $\overrightarrow{p(t)}_{IND} \in \{1, 2, ..., 5\}$, and price average level, $\overline{\Delta p(t)}_{IND} \in \{1, 2, ..., 8\}$. So:

$$s(t) = \left[B(t); G(t); \overrightarrow{p(t)}_{IND}; \overline{\Delta p(t)}_{IND}\right], s(t) \in S. \tag{9}$$

5.1.2 Actions

The actions related to the proposed decision-making problem indicate when and how much energy to sell considering the price trend and the current level of generation. In order to simplify the problem, the set of actions is composed of discrete integer values that represent the amount of energy to be sold at a given instant. A null value implies that no energy should be sold at a given time; on other hand, if there is no generation, an action equal to the maximum storage value, B_{MAX}, means that all stored energy must be sold, considering a full battery.

Table 3 Average price index

$\overline{\Delta p(t)}$ intervals	$\overline{\Delta p(t)}_{IND}$
$\overline{\Delta p(t)} > 120$	5
$100 < \overline{\Delta p(t)} \leq 120$	4
$80 < \overline{\Delta p(t)} \leq 100$	3
$60 < \overline{\Delta p(t)} \leq 80$	2
$\overline{\Delta p(t)} \leq 60$	1

It is worth mentioning that, for each state, there is a set of admissible actions. At a given moment, one must not choose to sell more energy than the amount available, and if the battery is full, one should not choose to store energy at the risk of exceeding the storage limit.

Hence, the set of actions is defined as:

$$A = \{0, 1, 2, ..., B_{MAX} + G_{MAX}\}, \tag{10}$$

where G_{MAX} is the maximum amount of energy generated. Given that, we define the amount of available energy $C(t)$ as the sum of the stored energy $B(t)$ and the generated energy $G(t)$ at a given instant:

$$C(t) = B(t) + G(t). \tag{11}$$

The admissible set of actions is a subset of A and it is related to the current state. Given that $0 \leq G(t) \leq G_{MAX}$ and $0 \leq B(t) \leq B_{MAX}$, and defining:

$$\Delta B(t) = B_{MAX} - B(t) \tag{12}$$

as the storage capacity available at the instant t, then we have:

$$A_s = \{G(t) - \Delta B(t), ..., C(t)\}, \text{when } G(t) > \Delta B(t) \tag{13}$$

and:

$$A_s = \{0, ..., C(t)\}, \text{when } G(t) \leq \Delta B(t). \tag{14}$$

5.1.3 Reward function

The reward function maps the current state and the chosen action on a reward, which is a real number. We use a reward function that corresponds to an index that considers the information of selling energy for a price that is above or below the average energy price for a given sequence. Thus,

$$R(s(t), a(t)) = a(t) \times (p(t) - \overline{p}), \tag{15}$$

where $\overline{p}$ corresponds to the price data average from the initial state to the current state, $p(t)$ is the price at

the decision time and $a \in A_s$ is the amount of energy to be sold in state $s \in S$.

5.2 Q-Learning and RLbEMS

The Q-Learning algorithm is the core of the RLbEMS training phase. Following the basic RL principle, the proposed system acquires knowledge about the environment through trial and error, i.e., it tries actions for a given state and learns about the effect of that action based on the evaluation of the obtained reward. However, in RLbEMS this learning is conducted in an offline training phase, using historical data for experimentation. The RLbEMS learning algorithm is given in Algorithm 2. The implemented algorithm has the same structure as Algorithm 1, and has as main parts: *Definition of training parameters*, *Action selection strategy* and *Update of system status*. Each one of these blocks and their importance to the implementation are discussed in the following subsections.

5.2.1 Definition of Training Parameters

The first block of the algorithm is where the system data is loaded. In this block, the system receives the price and generation historical data and the maximum storage capacity, B_{MAX}. Also, parameters α and γ are defined as 0.1 and 0.95, respectively.

After this, the algorithm generates all possible states and actions, considering the given system characteristics. Besides, also in this block, matrix $Q_{S\times A}$ is initialized as a null matrix and the initial state is defined. The initial state is chosen randomly, considering the set of possible states defined before.

5.2.2 Action Selection Strategy

As stated before, RLbEMS must learn about the environment and, also, must maximize the expected long-term reward. This is generally cited on the literature as the *Exploit* $\times$ *Explore* dilemma, when the agent has to choose at each step whether to explore the environment and to acquire more knowledge, or to choose the action that leads to maximize the expected long-term reward.

A way to do this is, at times, to choose the action that maximizes the expected long-term reward, and at other times to choose a random action in order to obtain information about the system. The frequency at which the system prioritizes to exploit or to explore is known as the rate of exploration, ϵ, a numerical index that indicates when each option must be followed. This index represents a probability of occurrence and can be defined as fixed or variable, depending on the studied problem. In cases which ϵ varies over time, the agent prefers to explore more frequently in the beginning of the learning process, choosing and applying random actions, and, as times goes by, the agent starts choosing more frequently to maximize the expected reward.

We here chose to keep ϵ fixed and equal to 30% during the training mode. At each step, the algorithm chooses to explore with a probability of 30% and to exploit with a probability of 70%. In general, choosing to explore means to randomly apply an admissible action to a given state; on the other hand, choosing to exploit means choosing an action that corresponds to the maximum value of the value-action function for a given $s(t) \in S$. However, if exploiting is chosen for a non-visited state, the algorithm considers a *Naïve-greedy* policy in which the system chooses to sell all the generated energy, $G(t)$.

5.2.3 Update of System Status

After choosing to exploit or to explore, the next step is to apply the chosen action, receiving the immediate reward and updating the system status. The immediate reward is calculated as presented in Eq. 15 and is used to update matrix $Q_{S\times A}$ as presented in Eq.4.

Also, in this block the system evolves to the next state, i.e., the values of $B(t)$, $G(t)$, $\overrightarrow{p(t)}_{IND}$ and $\overline{\Delta p(t)}_{IND}$ are updated. Besides, the new value of $Q(s(t), a(t))$ is calculated as indicated in Eq. 4. Finally, the current state is updated and the learning step is incremented.

The RLbEMS training mode runs until all the available price data are visited. After this, the policy is obtained considering Eq. 5.

5.3 The RLbEMS Operation Mode

After training, the system enters operation mode, in which the current information determines the state of the system. Having the current state and the energy-selling policy learned in the training mode for the current season, the action is defined and the incremental user profit is calculated, as illustrates Algorithm 3.

It is worth noting that the selling action may be changed depending on the user consumption. In our model, the user consumption would change the control action to be taken, indicating that we might sell or buy energy, as follows. Be $u(t)$ the user consumption at time t, then:

$$C_u(t) = B(t) + G(t) - u(t). \quad (16)$$

In the model that does not consider user consumption, the set of actions applicable to a given state is given by actions with null or positive values, indicanting that the EMS must sell nothing or at most $C(t)$ power at that instant. Now, the action may be negative, indicating that the EMS should buy energy. Thus, we have three cases:

1. If $C_u(t) < 0$, then the EMS should buy $C_u(t)$ energy;
2. If $C_u(t) \geq 0$ and $C_u(t) \leq B_{MAX}$, then $A_s = \{0, ..., C_u(t)\}$;
3. If $C_u(t) \geq 0$ and $C_u(t) > B_{MAX}$, then $A_s = \{C_u(t) - B_{MAX}, ..., C_u(t)\}$.

Once it is difficult to obtain real data of the user consumption, the tests described in the next section were made without considering it. This, however, does not invalidate the analysis, once the user consumption is not inserted in the training mode.

Algorithm 2 Q-Learning for RLbEMS

Require: Define parameters: α, γ;
Require: Load data: Price data, energy generation, B_{MAX};
Require: Define $finalstep$ as the size of the price vector;
Require: Define the set of states and actions: S, A;
Require: Initialize the table entry $Q_{S\times A}$ to zero.
Define a random inital state $s(t = 0)$;
while $step \leq finalstep$ **do**
Calculate the energy available to sell: $C(t) \leftarrow B(t) + G(t)$;
Decide for Exploit or Explore:
if Choose Exploit **then**
$action \leftarrow argmax_a(Q(s(t), a))$;
if Q(s(t),.)=0 **then**
$action \leftarrow G(t)$; *// Naïve–greedy policy*
end if
$a(t) \leftarrow action$;
else
Choose Explore
Define an admissible set of actions: A_s
Choose a random action from A_s;
$a(t) \leftarrow action$;
end if
Apply a(t);
Calculate the immediate reward: $R \leftarrow a(t) \times (p(t) - \overline{p})$;
Evolve the system state:
$B(t+1) \leftarrow (B(t) + G(t) - a(t))$;
$\overrightarrow{s(t+1)} \leftarrow [B(t+1); G(t+1); \overrightarrow{p(t+1)}_{IND}; \overrightarrow{\Delta p(t+1)}_{IND}]$;
Update the table entry for $Q(s(t), a(t))$ as in Eq.4;
$s(t) \leftarrow s(t+1)$;
$step \leftarrow step + 1$;
end while

6 Experimental Results

In this section, two case studies were performed to evaluate the RLbEMS performance. Considering the influence of the climate characteristics and difference in the energy price model, we conducted these case studies in two different places, USA and Brazil. Those tests were important to validate the proposed system in different conditions of operation. As will be analyzed in this section, there is a strong relation between the pricing signal, the generation profile and the increase of the accumulated profit in a given period. All the tests compare RLbEMS-generated policies with a *Naïve-greedy* policy, showing that RLbEMS is very effective.

Before showing the results for each case study, we describe in the next subsection the methodologies used

Algorithm 3 The RLbEMS operation mode

Require: Season of the year;
Require: Learned policy π_{SEASON};
while TRUE **do**
Observe the current state $s(t)$;
$a(t) \leftarrow \pi_{SEASON}(s(t))$
Apply $a(t)$;
Calculate the current profit: $PR(t) = a(t) \times p(t)$
end while

to run the training mode and to test the operation mode of the RLbEMS system.

6.1 Training and Testing Methodologies

Training is the way in which the RLbEMS system learns an energy selling policy, i.e., it learns the policy that defines when and how much energy RLbEMS must sell considering the state of the problem. For a good performance, the data used for training should follow a pattern similar to the data received during the operation of the system. In other words, the data available for training should reflect the dynamics of the real system. Thus, the on-line use of the policy learned in training will enable the RLbEMS system to maximize long-term profit.

Considering the state definition for the present problem, the energy microgeneration and the energy price are both external variables that vary their profiles in a long period. In particular, the price of energy in the USA varies considerably and shows a seasonal pattern, i.e., there is a specific pattern of energy price for each season, as can be seen in Fig. 13.

In this paper, the annual USA energy price data used for training were previously divided into four sets of data, each one corresponding to one season of the year. After the division, the seasonal price data for each year were combined, resulting in a larger set of data for each season that was a combination of the seasonal data for different years.

After that, each set of data for each season was divided again into four data slots. These slots were grouped in different ways and used as separate training sets. Each training using a training set learned a policy for that season. In the end, four policies $\pi_i, i \in [1, 2, 3, 4]$ were learned for each season.

The training mode runs Algorithm 2 for each seasonal data set, considering a specific number of episodes. An *episode* begins with choosing a random initial state and ends when all the price data are visited. At each step of the episode the $Q_{S\times A}$ matrix is updated. The total number of episodes corresponds to the minimum number of iterations in which we observed that the learning process converges. The procedure of finding the necessary number of episodes for each season involves to run the training mode until we see that the accumulated reward remains fixed and does not change even if we increase the number of episodes. This procedure, together with the final policy for the winter, can be viewed in Fig. 14.

After the training process is completed, there is a set of four policies for each season, which must be combined. The union of these four policies results in the final seasonal policy that will be applied to the RLbEMS operation mode:

$$\pi_{SEASON} = \bigcup (\pi_1, \pi_2, \pi_3, \pi_4)\,, \tag{17}$$

where the union is made by taking, for each state, the action that gives the maximum value-action function value. As stated previously, we chose a *Naïve-greedy* policy when a state has not been visited in any of the partial policies π_i. This training methodology makes learning better by exchanging the sequence of data used in the training mode, for each season.

In a real runtime system, the policy learned for the current season π_{SEASON} would be used with data observed at the moment of decision, following Algorithm 3.

To perform our tests, we simulated the RLbEMS operation mode using the data available for energy price and energy generation for both locations, USA and Brazil.

The four policies π_{SEASON} generated during the training were tested and compared with a *Naïve-greedy* policy. The *Naïve-greedy* policy considers a

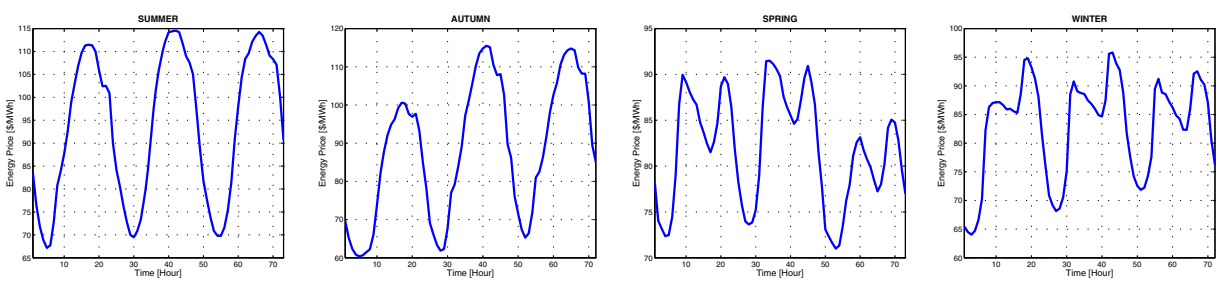

Fig. 13 Price pattern for three consecutive days of each season in the USA. As can be seen, the price has similar patterns in each case

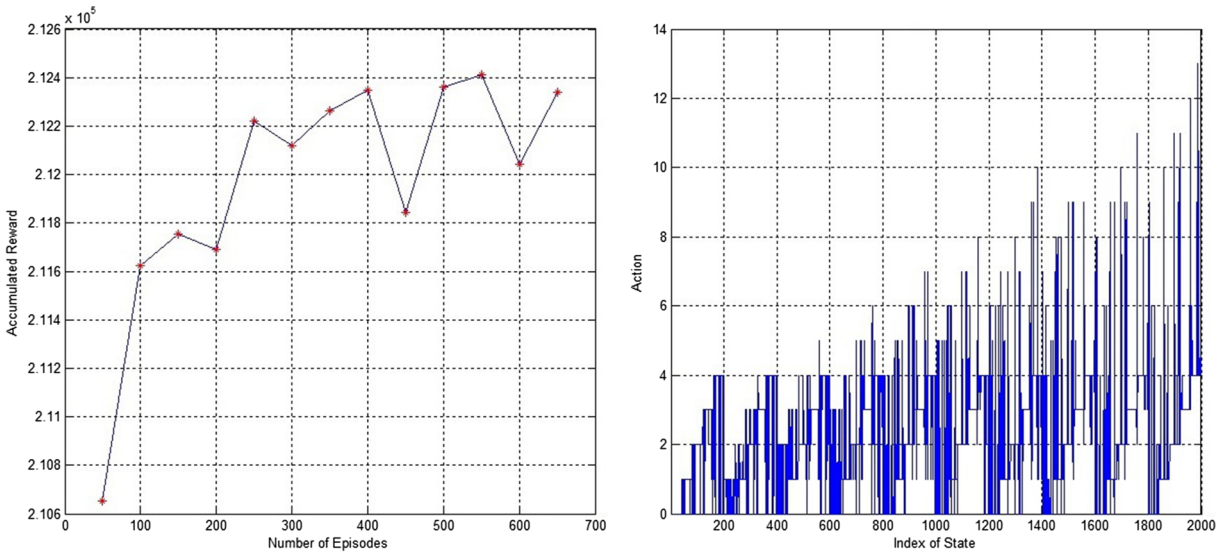

Fig. 14 Output of the training process: Graph that shows the convergence of the learning process for the winter (*Left*). The final policy learned for the Winter (*Right*)

system without a storage device, i.e., at each instant, all the energy generated is sold at the current price.

As a comparative index, we calculated the percentage increase of the profit accumulated over each season, given by the ratio of the policy obtained by using RLbEMS and the *Naïve-greedy* policy.

6.2 Case Study 1: The North-American Smart Home

This case study was performed considering the real pricing signal for the District of Columbia, USA. These data correspond to the Local Marginal Price (LMP), which reflects the value of energy at a specific location at the time it is delivered [24]. This price is also used to calculate the amount of money paid to the consumer when he/she sells energy to the power grid.

For this test, we used the hourly pricing data of consecutive years (2008-2013). We applied the data from 2008 to 2012 to the system training mode and, after having a policy for each season, we simulated the operation mode using the data from 2013.

For the simulation, we considered the Solar PV System suggested by [17]. This system is composed of the KD200-54 P series Photovoltaic modules from the Kyocera Solar Incorporation [29] that has 220Wh as the peak energy generation per module. In our tests, we consider a system composed of four modules. This generation profile was discretized as packages of energy, each package corresponding to 200Wh of generated energy. The real generation profile and the corresponding discretized profile can be seen in Fig. 5.

The storage device is considered as a set of batteries that works at 48V with a capacity of 200Ah. The storage device can supply energy for a load of 1.8kW for three hours, approximately. Thus, given the same discretization used for the generation system, B_{MAX} is restricted to 9, and $0 \leq B(t) \leq 9$.

First of all, the system characteristics and the historical data of energy price and energy generation from 2008 to 2012 were inserted into RLbEMS, that was put on training mode. After calculating the selling policy for each season, the system was put on operation mode, when the data of energy price and energy generation for 2013 were sequentially inserted. By the end, the simulation is stopped and the system calculates the accumulated profit for the RLbEMS policy and the one that would be achieved if the system were using the *Naïve-greedy* policy. The results per season can be viewed in Table 4.

Table 4 Result of the test for the Smart Home in the USA: The values represent the percentage increase of the accumulated profit in comparison to a *Naïve-greedy* policy

	Summer	Autumn	Spring	Winter
Profit growth	1.3 %	1.7 %	1.5 %	2.0 %

The application of the RLbEMS policy increases the accumulated profit by the end of each season. However, this increase is not so expressive, considering that the implementation of a system with a *Naïve-greedy* policy is simple and does not require the RLbEMS infrastructure to be defined. This result can be explained by the characteristics of the generation and price profile observed at that locality.

As can be viewed in Fig. 15, there is some similarity in the patterns of energy generation and energy price for all the seasons. Thus, as the peak of generation is near to the peak of price, using the *Naïve-greedy* policy naturally gives a good result, because there is always a good level of generation when the price is high, and also there is always a poor level of generation when the price is low. Hence, most of the time the RLbEMS chooses actions equivalent to the *Naïve-greedy* policy and the small increase observed when we apply the RLbEMS policy occurs because of the little difference between both curves. Besides, the increase is higher when there is a major difference, as in winter, for example.

Observing Fig. 15, we can see that the obtained results are justified by the dynamics of the system. The increase of profit will be greater when we have a peak of generation in a different time of a peak of price. This justifies the storage of energy. In this case, the system would choose to store energy while the price is low, selling the energy in the future at a better price. This behavior can be observed in the second case study, in which the peak of generation occurs in a different time of the day when there is a high energy price.

6.3 Case Study 2: The Brazilian Smart Home

Our second case study considers a Smart Home located in São Paulo, Brazil. The differentiated tarif

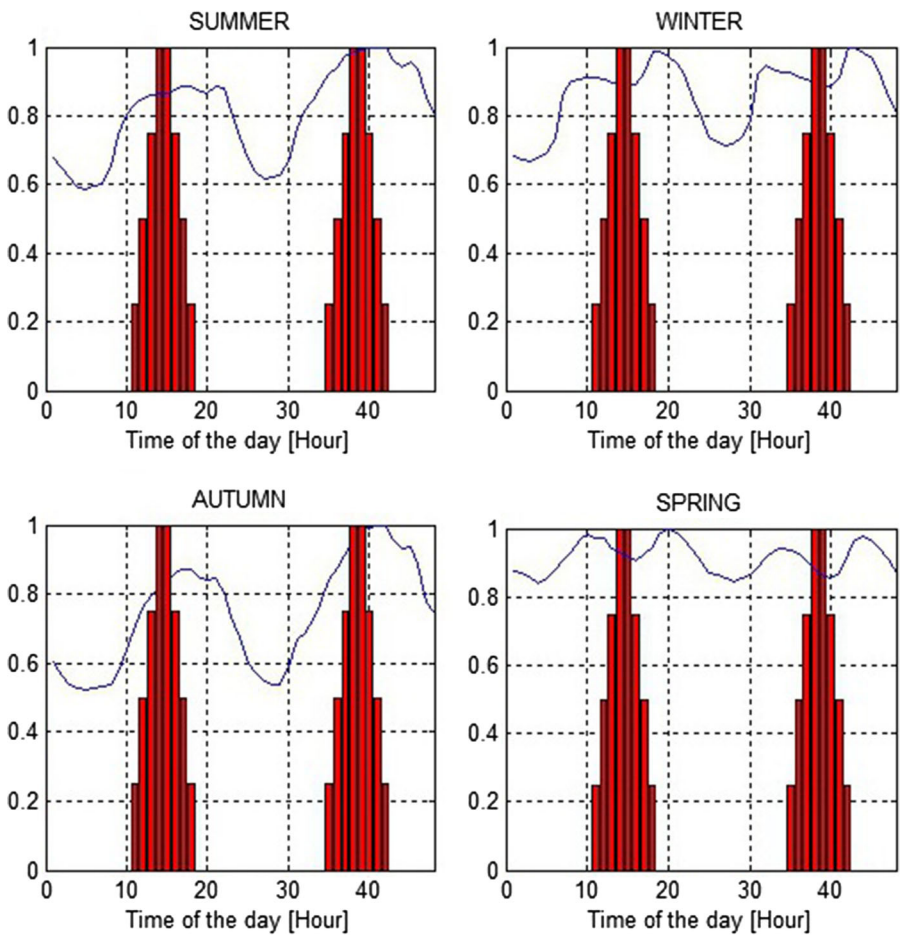

Fig. 15 Generation and price pattern comparison for two consecutive days in the USA: Pricing signal is similar to the energy generation profile, which contributes to a good result using the *Naïve-greedy* policy

f implemented in Brazil is a Time-of-use tariff, i.e., there is a fixed energy price for each time of the day, as can be viewed in Fig. 8. This pricing signal remains fixed all days of the year and concentrates major values of energy price at times when there is a high demand for energy. Given this, the pricing data used both for training and operation consists of the same pricing signal shown in Fig. 8, repeated considering the number of the days of the simulation.

The energy generation data were acquired from the Smart Grid and Power Quality Laboratory ENERQ-CT at the University of São Paulo, Brazil. The solar power plant is composed of ten modules from LG, connected as a serial array. Each module generates 255Wh at the peak of generation, resulting in a total peak of 2.55kWh. Besides, there is a complete meteorological station, where it is possible to measure solar radiation, wind speed and temperature. We obtained hourly energy generation data from August 2013 to January 2014. These data were treated and discretized in the same way as described in Section 6.2. The storage device has the same characteristics as the one described in Section 6.2. Given that the Solar PV System used in this case study generates more energy, we also tested batteries with higher capacities in order to verify its influence on the accumulated profit. For the better case, we tried a battery that has 5.4kW maximum storage capacity. Thus, considering the same discretization used for the generation system, B_{MAX} is restricted to 27, and $0 \leq B(t) \leq 27$.

The procedure used for training and testing is similar to that proposed in Section 6.2. The available data for generation were divided into two sets for training and testing, and a set of data for the energy price was generated considering the Time-of-use tariff. Hence, considering that the amount of avaiable data is smaller, we chose to implement a solution with a single policy. This decision does not affect the final result, even because in this case study the princing signal is fixed and the energy generation does not vary too much during this period of the year.

As expected, the increase of the accumulated profit of 44.96% in this case study is much more expressive as compared to the first case study. A comparison between the generation profile and the pricing signal, can be viewed in Fig. 16.

As can be viewed, in this test the generation has its peak during the middle of the day, while the peak price occurs on the beggining of the night. Given this, the RLbEMS choses to store energy during the day until

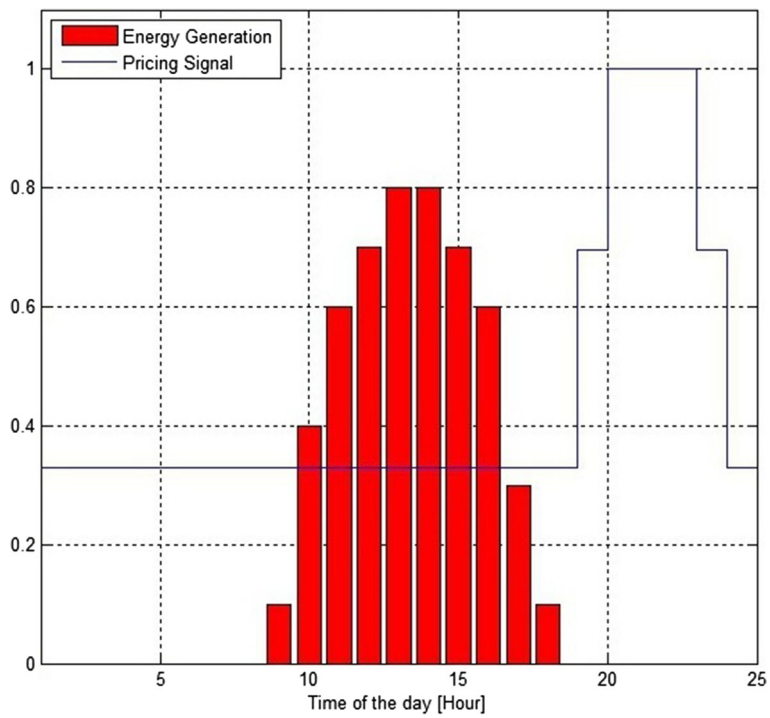

Fig. 16 Generation and price pattern comparison for Brazil. Pricing signal has a different peak hour in comparison to the energy generation profile, which contributes to a good result using the RLbEMS policy

the maximum storage capacity, selling it in the future with a higher price. This behavior can be viewed in Fig. 17.

This result shows that the application of RLbEMS is feasible, and that the accumulated profit is greater when the energy price profile differs from the energy generation profile. Considering the TOU energy price model, we achieved the expected energy-selling policy, demonstrating that the RLbEMS is working properly. In this case, considering that the energy-selling policy is simple, it can be implemented by a simpler system. However, the application of the RLbEMS is important to show that the system is always achieving the best possible policy.

Another test analyzes the relation between the accumulated profit and the maximum storage capacity. The same tests were performed with different batteries. Results in Fig. 18 show, as expected, that the accumulated reward increases when a battery with a higher maximum storage capacity is applied.

This result is very important and must be analyzed during the system design, because batteries are expensive devices and their costs rise with the increase of maximum storage capacity. Thus, the increase of the system total cost must be justified by the increase of the profit, for the solution to remain feasible.

7 Concluding Remarks and Future Work

This paper proposed a Reinforcement Learning-based Energy Management System called RLbEMS that maximizes the profit achieved by residential consumers in their energy sales, considering a Smart Home that generates its own energy and that has a storage device. Besides, this Smart Home receives a pricing signal that varies during the time of the day, which justifies the use of a RLbEMS solution that, for each state of the system, must choose between selling energy at the current price or store energy to sell at a better time in the future. The problem was modeled as a MDP and a RL algorithm was used to solve the problem. The system has two modes of operation: the training mode uses historical data to obtain a selling policy for each season of the year and the operation mode applies the learned policies considering the sequential real-time data.

In order to validate the developed system, we analyzed two case studies. The first considers a Smart Home in the USA, where we obtained a maximum gain of 2.0% for RLbEMS in comparison to a *Naïve-greedy* policy. This small increase is justified by the great similarity between the energy generation pattern and the pricing signal for this locality. The second considers a Smart Home in Brazil, where we obtained

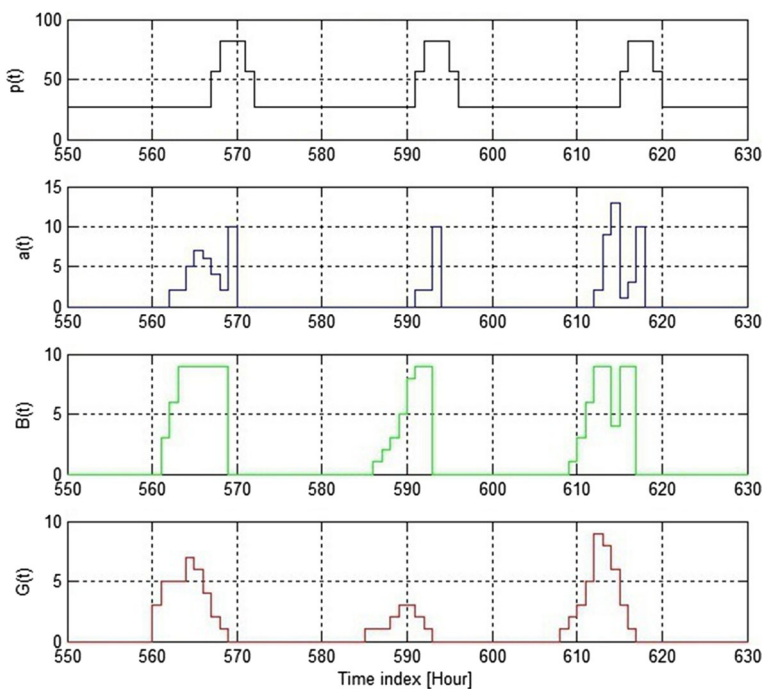

Fig. 17 RLbEMS operation for three consecutive days in Brazil: From top to bottom: price curve, action performed, stored energy, and power generated. The system choses to store energy during the day and sell the same energy in a moment where there is a higher price in the market

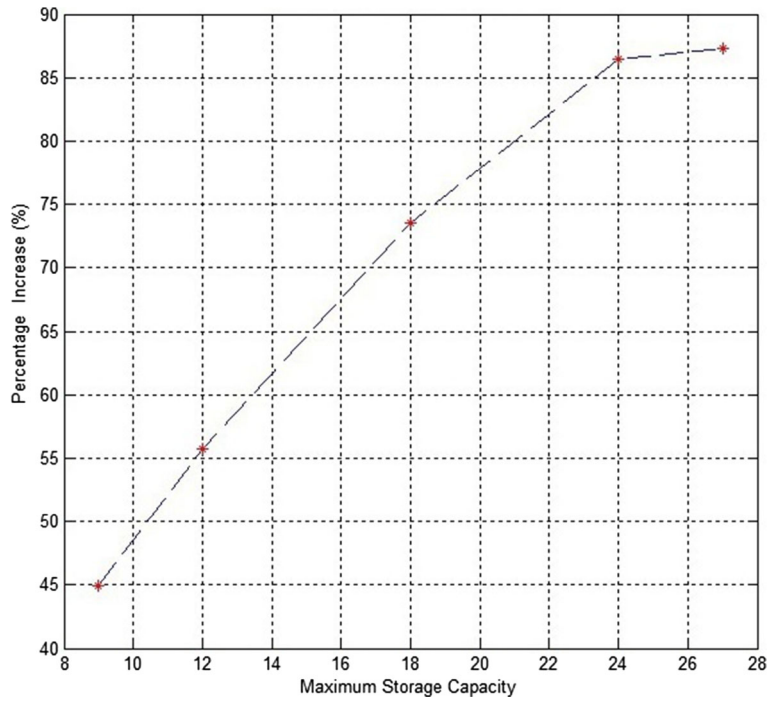

Fig. 18 Tests perfomed for different types of battery, with different maximum storage capacities. As can be viewed, the accumulated profit increases while we increase the maximum storage capacity

a maximum gain of 44.9% for the application of RLbEMS. In this second case, the energy generation profile differs considerably from the pricing signal, justifying the use of storage devices considering that the system chooses to store energy when the price is low and to sell the stored energy when the price is high. Tests were also run to analyze the influence of the maximum capacity of the battery on the total accumulated profit. The results of these tests showed that these parameters are highly related. The accumulated profit is greater in cases in which there is a higher battery capacity. This is expected considering that a higher maximum storage capacity allows the system to store more energy when the price is low and, consequently, to sell a major amount of energy when the price is high. This result is important but must be analyzed carefully, because batteries are expensive devices and their costs grow with the increase of the maximum storage capacity. Hence, the increase in profit must justify the implementation of a more expensive system, for the solution to remain feasible.

The major contribution of the article is to propose a new approach for modeling and solving the energy managment problem in Smart Homes. We propose a system that was tested with real energy generation and energy price data for two different places and, considering that these data depend on the location, we have showed that the proposed system can be used in situations characterized by different levels of uncertainty. The results demonstrate that the modeling using MDP and the solution using reinforcement learning are feasible, despite the limitations such as the small amount of data available for training. The solutions proposed in the literature describe the system in a different way and, in the most cases, the tests realized do not consider real generation and price data. Moreover, the proposed solution guarantees the degree of freedom of the user, since it optimizes the usage of energy sources considering only the price informed. Thus, the user does not have restrictions regarding the use of appliances in his/her home. We believe this degree of freedom is essential so that a solution as proposed in the article can be incorporated into the routine of the user, leading to the popularization of systems like this.

Our future work considers the insertion of information about the tendency of energy generation into the model, as we did for the pricing signal. This is important for the application of RLbEMS in places where the energy generation varies considerably during the year. Besides, with this improvement, the system will be able to operate with other alternative sources of energy, such as wind, which consists a power generation much more intermittent than

the Solar PV System. Another system improvement aims at inserting the energy demand of the house into the energy balance and a more complete model for the storage device. In this case, besides energy generation and price, the system will have to acquire knowledge about consumers behavior. The developed system allows to enter the energy demand data into the MDP model without significant changes in the proposed solution. Tests in the future should also consider a profit method based on the energy price profile per day, and the implementation of different learning algorithms to find the one that best fits the studied problem.

Acknowledgements We would like to thank CNPq (131239/2013-9 and 311058/2011-6) and FAPESP (Project CogBot, process 2011/19280-8) for supporting this research. We also thank the Smart Grid and Power Quality Laboratory ENERQ-CT at University of São Paulo for providing the solar photovoltaic generation data used in this work. We thank the reviewers for their valuable comments; one of the reviewers gave us valuable suggestions concerning future work.

References

1. Hammoudeh, M.A., Mancilla-David, F., Selman, J.D., Papantoni-Kazakos, P.: Comunication Architectures for Distribution Networks within the Smart Grid Initiative. In: Green Technologies Conference, IEEE, p. 65,70 (2013). doi:10.1109/GreenTech.2013.18
2. Hashmi, M., Hanninen, S., Maki, K.: Survey of smart grid concepts, architectures, and technological demonstrations worldwide. In: IEEE PES Conference on Innovative Smart Grid Technologies (ISGT Latin America), p. 1,7 (2011). doi:10.1109/ISGT-LA.2011.6083192
3. Uluski, R.W.: The role of Advanced Distribution Automation in the Smart Grid. In: Power and Energy Society General Meeting, IEEE, p. 1,5 (2010). doi:10.1109/PES.2010.5590075
4. Palensky, P., Dietrich, D.: Demand Side Management: Demand Response, Intelligent Energy Systems, and Smart Loads . IEEE Trans. Ind. Inf. **7**(3), 381,388 (2011). doi:10.1109/TII.2011.2158841
5. Chen, C., Kishore, S., Zhifang, W., Alizadeh, M., Scaglione, A.: How will demand response aggregators affect electricity markets? A Cournot game analysis. In: 5th International Symposium on Communications Control and Signal Processing (ISCCSP), p. 1,6 (2012). doi:10.1109/ISCCSP.2012.6217839
6. Parvania, M., Fotuhi-Firuzabad, M., Shahidehpour, M.: Demand response participation in wholesale energy markets. In: Power and Energy Society General Meeting, IEEE, p. 1,4 (2012). doi:10.1109/PESGM.2012.6344591
7. Shishebori, A., Kian, A.R.: Risk analysis for distribution company energy procurement with pool market, DGs and demand response. In: 18th Iranian Conference on Electrical Engineering (ICEE), p. 949,954 (2010). doi:10.1109/IRANIANCEE.2010.5506940
8. Chua-Liang, S., Kirschen, D.: Quantifying the Effect of Demand Response on Electricity Markets. IEEE Trans. Power Syst. **24**(3), 1199,1207 (2009). doi:10.1109/TPWRS.2009.2023259
9. Keen, P.: Decision support systems : a research perspective. Center for Information Systems Research, Cambridge (1980)
10. Wright, A., Sittig, D.: A framework and model for evaluating clinical decision support architectures q. J. Biomed. Inform. **41** (2008). doi:10.1016/j.jbi.2008.03.009
11. Power, D.J.: Decision support systems: concepts and resources for managers. Westport, Conn., Quorum Books (2002)
12. Stephens, W., Middleton, T.: Why has the uptake of Decision Support Systems been so poor? In: Crop-soil simulation models in developing countries. Wallingford:CABI (2002)
13. Sol, H.G., Takkenberg, C.A.Th., de Vries Robb, P.F.: Expert systems and artificial intelligence in decision support systems . Second Mini Euroconference, Lunteren (1987). ISBN: 90-277-2437-7
14. Efraim, T., Jay, E.: Decision Support Systems and Intelligent Systems, p. 574. Prentice Hall (2008). ISBN-13: 978-0137409372
15. Dusparic, I., Harris, C., Marinescu, A., Cahill, V., Clarke, S.: Multi-agent residential demand response based on load forecasting. In: 1st IEEE Conference on Technologies for Sustainability (SusTech), p. 90,96 (2013). doi:10.1109/SusTech.2013.6617303
16. O'Neill, D., Levorato, M., Goldsmith, A., Mitra, U.: Residential Demand Response Using Reinforcement Learning. In: First IEEE International Conference on Smart Grid Communications (SmartGridComm), p. 409,414 (2010). doi:10.1109/SMARTGRID.2010.5622078
17. Chen, X., Wei, T., Hu, S.: Uncertainty-Aware Household Appliance Scheduling Considering Dynamic Electricity Pricing in Smart Home. IEEE Trans. Smart Grid **4**(2), 932,941 (2013). doi:10.1109/TSG.2012.2226065
18. Mohsenian-Rad, A.H., Wong, V.W.S., Jatskevich, J., Schober, R., Leon-Garcia, A.: Autonomous Demand-Side Management Based on Game-Theoretic Energy Consumption Scheduling for the Future Smart Grid. IEEE Trans. Smart Grid **1**(3), 320,331 (2010). doi:10.1109/TSG.2010.2089069
19. Atzeni, I., Ordonez, L.G., Scutari, G., Palomar, D.P., Fonollosa, J.R.: Demand-side management via distributed energy generation and storage optimization. IEEE Trans. Smart Grid **4**(2), 866,876 (2013). doi:10.1109/TSG.2012.2206060
20. Balijepalli, V.S.K.M., Pradhan, V., Khaparde, S.A., Shereef, R.M.: Review of demand response under smart grid paradigm. In: Innovative Smart Grid Technologies - India (ISGT India), IEEE PES, p. 236,243 (2011)
21. Albadi, M.H., El-Saadany, E.F.: Demand Response in Electricity Markets: An Overview. In: Power Engineering

Society General Meeting, IEEE, p. 1,5 (2007). doi:10.1109/PES.2007.385728

22. Carpinelli, G., Celli, G., Mocci, S., Mottola, F., Pilo, F., Proto, D.: Optimal Integration of Distributed Energy Storage Devices in Smart Grids. IEEE Trans. Smart Grid **4**(2), 985,995 (2013). doi:10.1109/TSG.2012.2231100
23. Zhimin, W., Chenghong, G., Furong, L., Bale, P., Hongbin, S.: Active Demand Response Using Shared Energy Storage for Household Energy Management. IEEE Trans. Smart Grid **4**(4), 1888,1897 (2013). doi:10.1109/TSG.2013.2258046
24. PJM monthly locational marginal pricing [Online]. Available: http://www.pjm.com/markets-and-operations/energy/real-time/monthlylmp.aspx
25. Bueno, E.A.B., Utubey, W., Hostt, R.R.: Evaluating the effect of the white tariff on a distribution expansion project in Brazil. In: IEEE PES Conference On Innovative Smart Grid Technologies Latin America (ISGT LA), p. 1,8 (2013). doi:10.1109/ISGT-LA.2013.6554479
26. Russell, S.J., Norvig, P. Artificial Intelligence: A Modern Approach, 2nd edn. Pearson Education, Inc., Upper Saddle River (2003)
27. Sutton, R.S., Barto, A.G., Hu, S.: Reinforcement Learning: An Introduction. The MIT Press, Cambrige
28. Watkins, C.J.C.H.: Learning from Delayed Rewards, PhD thesis. Cambridge University, Cambridge (1989)
29. Kyocera Solar, Data Sheet of KD200-54 P Series PV Modules [On- line]. Available: http://www.kyocerasolar.com/assets/001/5124.pdf