Self-learning agent for battery energy management in a residential microgrid

Brida V. Mbuwir*†, Fred Spiessens*‡, Geert Deconinck*†
*Algorithms, Models & Optimization, EnergyVille, Thor Park 8130, 3600 Genk, Belgium
†ESAT/Electa, KU Leuven, Kasteelpark Arenberg 10, 3001 Leuven, Belgium
‡Algorithms, Models & Optimization, Flemish Institute for Technological Research (VITO), Boeretang 200, 2400 Mol, Belgium

***Abstract*—This paper presents a data-driven control approach for battery energy management in a residential microgrid. The objective is to develop a battery operation mode scheduling strategy that maximizes self-consumption of local PhotoVoltaic (PV) production. A model-free reinforcement learning technique, policy iteration, is used to solve the underlying sequential decision-making problem. The proposed algorithm learns the stochastic load of the consumer, the PV production, and the dynamics of the microgrid to construct a control policy. A case study of a residential microgrid is used to evaluate the performance of the policy iteration algorithm. Simulation results using data from Belgian residential consumers show a 12% increase in the PV captured by the battery, leading to a 30% decrease in electricity costs compared to the fitted Q-iteration algorithm. These results show the feasibility and effectiveness of the proposed algorithm.**

***Index Terms*—Energy management, microgrid, policy iteration, reinforcement learning.**

I. Introduction

The decreasing cost of PhotoVoltaic (PV) systems and rising concerns on global warming have been a motivation for the development of microgrids powered predominantly by intermittent Renewable Energy Sources (RES). A microgrid is a group of interconnected loads and distributed energy resources (like photovoltaic systems, batteries) within clearly defined electrical boundaries that can be controlled independently with respect to the main utility grid [1]. Microgrids can operate connected to or disconnected from the main utility grid. Residential microgrids stem for the prosumer context i.e. a household with energy generating sources, energy storage facilities - batteries, and loads. Due to the dual operation mode of microgrids, intermittency of RES, and uncertainties related to residential energy consumption patterns, there is a need for energy management systems in microgrids. Energy management systems provide amongst others effective control for the operation of energy storage facilities within the microgrid. This control influences the microgrid's interaction with the main utility grid while optimizing energy use.

Developing an efficient algorithm to control the operation of a battery in a microgrid is challenged by the presence of uncertainties and non linearities in the microgrid system. Previous research in this domain has mostly revolved around rule and model based controllers. With rule based methods, a set of rules is required to establish various kinds of scenarios for the system's operation [2]. Model based optimization techniques such as linear programming [3], dynamic programming [4], and model predictive control [5] have been successfully applied to solve energy optimization problems in microgrids. Aittahar *et al.* [6] proposed an imitative learning algorithm for operation of long and short term energy storage devices in a microgrid. Wei *et al.* [7] proposed a mixed iterative dynamic programming approach in which they combine policy and value iteration algorithms to solve the optimal battery energy management in a residential microgrid.

However, both rule and model based methods suffer from a common major disadvantage as they assume the presence of a reliable model of the system. While it may be economically viable to develop a detailed microgrid model for large commercial microgrids, it is rarely cost-effective to undertake the same effort for residential microgrids. Additionally, even when a model is available, problems can arise due to the nonadaptive nature of the model to operational/hardware changes or an inaccurate model can lead to a less performant microgrid.

Work on model-free techniques such as reinforcement learning (RL) [8] have been proposed in the recent years as a way to remedy complications arising due to the inflexibility and requirements of model and rule based methods. Recently used RL techniques in microgrid energy management include deep RL [9], batch RL [10] and Q-learning [11].

Motivated by the self-learning and adaptivity characteristic of policy iteration [12], an RL algorithm, this work contributes in the application of policy iteration in battery energy management. The main contribution of this paper is that we propose a policy interation algorithm for battery energy management which (i) uses a regression algorithm to compute approximations of the Q-function, and (ii) incorporates an epsilon-greedy algorithm for policy selection in order to balance exploration with exploitation during the learning phase.

The rest of the paper is structured as follows. Section II presents the microgrid system and formulates the energy management problem (battery operation mode scheduling) as a Markov decision process. Policy iteration, the RL control method considered in this work is presented in Section III. The simulation results are presented and discussed in Section IV. Finally, Section V summarizes the work with a conclusion and future directions.

978-1-5386-4505-5/18/$31.00

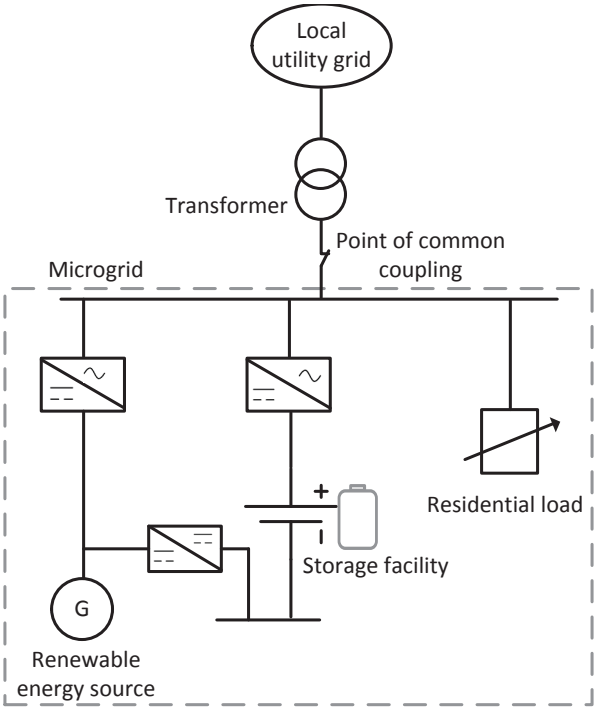

Figure 1. Microgrid model with DC/AC converters, for interfacing PV generation and storage to the load and the main utility grid [10].

II. System Description and Problem Formulation

This section presents the microgrid system considered, and formulates the battery operation mode scheduling problem for energy management as a Markov Decision Process (MDP).

A. Microgrid model

The microgrid considered in this work is grid-connected, and consists of a PV system, residential loads, power electronic converters, a transformer and a battery pack, as illustrated on Fig. 1. The power electronic converters are an essential component of the microgrid as they interface the distributed energy resources (PV system and battery) to the load and main utility grid. The converters also interface the PV system and the battery. Due to the non-linear nature of the inverters' efficiency profile as depicted on Fig. 2, these inverters introduce a non-linearity into the microgrid system. This non linearity has an impact on the microgrid's performance, and thus, should be taken into account when developing a control strategy for energy management in a microgrid.

The residential load can be met by using the energy produced by the local PV system, discharging the battery or partly covered by purchasing energy from the local utility grid. Excess electricity produced during low electricity demand and/or high production can be stored in the battery and reused during peak demand. The grid connection through the transformer provides information on electricity prices to the microgrid user and to the controller.

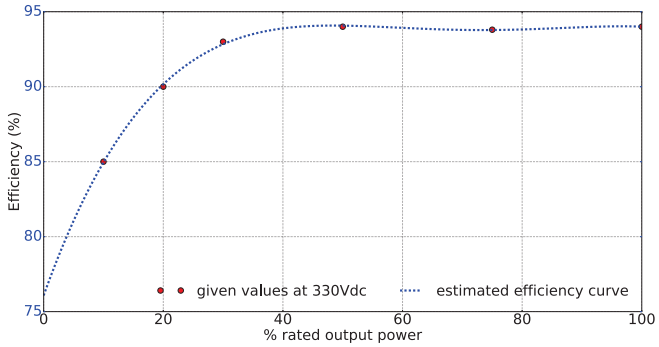

Figure 2. Efficiency curve of a 4kW AC/DC inverter [10].

Energy optimization is done by suitably designing the charging/discharging operation modes of the battery. The battery has three operation modes.

1) Charging mode: charge the battery directly with energy from the PV system and excess PV is used to supply the load. In the charging mode, the charging rates are discretized to 8 uniform values in the interval]0,2]. We assume that the battery cannot be charged with energy from the grid.
2) Idle mode: energy from the power grid directly supplies the load while the battery's SoC remains unchanged.
3) Discharging mode: the battery discharges to supply the load during periods of peak consumption, and with little or no PV production. We consider 8 discharging rates uniformly sampled in the interval [-2,0[

This paragraph presents a battery model that shows the dynamics of the battery regarding its mode of operation. The model provides information on the battery's energy level at every time step as shown in (1).

$$E_{t+1} = E_t + \eta P_{charge} \Delta_t - \frac{P_{discharge} \Delta_t}{\eta}, \tag{1}$$

where t = current timestamp, P_{charge} = charging rate, $P_{discharge}$ = discharging rate, η = charge/discharge efficiency, $\eta \in [0, 1]$, Δ_t = length of a control period or the time step, and E_t = the energy level of the battery at the beginning of timestamp t. To avoid a negative impact on the battery's lifetime, the battery is subjected to the following constraints.

(i) The capacity limits:

$$E_{min} \leq E \leq E_{max}, \tag{2}$$

where E_{max} and E_{min} are the maximum and minimum energy levels of the battery respectively, and E the energy level of the battery.

(ii) The charging and discharging power limits satisfy;

$$P_{min} \leq P \leq P_{max}, \tag{3}$$

where P_{min} and P_{max} are the minimum and maximum discharging and charging rates of the battery respectively, and P is the (dis)charging rate of the battery

(iii) The battery cannot be simultaneously charged and discharged.

The main objective of this microgrid system is to maximise self-consumption of the local PV production by controlling the operation modes of the battery. The operational mode control is a sequential decision-making problem and can be formulated as an MDP.

B. Markov decision process

Reinforcement learning problems can be formally formulated as MDPs. An MDP is defined by its state space $\mathcal{S}$, action space $\mathcal{A}$, transition function f and reward function ρ. The transition function describes the system's dynamics when the system moves from a state s_k to state s_{k+1} in response to a control action $a_k \in \mathcal{A}$.

$$s_{k+1} = f(s_k, a_k), \forall k \in \{0, 1, \ldots, T-1\}, \tag{4}$$

where T is the optimization horizon.

A reward/cost function, ρ, is associated to each state transition and evaluates the benefits/penalties of being in state s_k and taking the action a_k, (5).

$$c_k = \rho(s_k, a_k), \forall k \in \{0, 1, \ldots, T-1\}. \tag{5}$$

Reinforcement learning approaches aim to find a control policy $h : S \rightarrow A$, that minimizes the sum of the cost over the optimization horizon. The control policy is characterized by its state-action value function, called the Q-function. This Q-function computes the longterm cost of taking a control action a_k when in state s_k and following the policy h thereafter.

$$Q^h(s_k, a_k) = \rho(s_k, a_k) + \gamma R^h(f(s_k, a_k)), \tag{6}$$

where R^h is the aggregated sum of the of costs over the entire optimization horizon defined as follows.

$$R^h(s_0) = \sum_{k=0}^{T-1} (\gamma^k \rho(s_k, h(s_k))), \tag{7}$$

with γ being the discount factor, $\gamma \in [0, 1]$. γ takes into account the uncertainty about the future costs.

The state space, action space and cost function in the microgrid context are described below. Since we consider a model-free control strategy, there is no explicit representation for the transition function.

C. State Space ($\mathcal{S}$)

The state space, $\mathcal{S}$, consists of a non-controllable exogenous component, S_x, and a controllable component, S_c.

$$\mathcal{S} = S_x \times S_c \tag{8}$$

1) Exogenous features, S_x: the exogenous features, S_x, contain the observable information that has an impact on the system's dynamics and the cost function, but cannot be influenced by the control actions. This includes a load component, S_x^l and PV production component, S_x^{PV} . A timing component, S_t, also forms part of the exogenous features and provides the learning agent with information on the day of the week and quarter-hour of the day as follows:

$$S_t = S_t^d \times S_t^q, S_t^d \subseteq \{0, \ldots, 6\}, S_t^q \subseteq \{0, \ldots, 95\}, \tag{9}$$

where $s_t^q \in S_t^q$ represents the quarter-hour of the day, and $s_t^d \in S_t^d$ the day of the week.
Thus, the exogenous feature is defined as:

$$S_x = S_t \times S_x^l \times S_x^{pv}, \tag{10}$$

where $\forall s_x \in S_x, s_x = \{s_t^d, s_t^q, load, PV\}$.

2) Controllable features, S_c: provide state information on system quantities whose values can be influenced by the control actions. In this case, the controllable component consist only of the battery's SoC:

$$\forall s_c \in S_c, s_c = \{SoC\}, \tag{11}$$

where the SoC is defined as:

$$SoC = \frac{E}{E_{max}}. \tag{12}$$

Thus, the microgrid's state at time step k is defined by the vector:

$$s_k = (s^d, s^q, SoC, load, PV) \in \mathcal{S}, \forall k \in \{0, 1, \ldots, T-1\}. \tag{13}$$

D. Action Space ($\mathcal{A}$)

At each time step, the learning agent takes an action relating to the operation mode of the battery. The battery's operation modes are discussed in section II-A. This action represents a (dis)charging rate of the battery depending on its sign. As such, the action space is defined consisting of 17 discrete samples in kilowatts between [-2,2], and includes a value of 0 (idle mode of the battery).

$$a_k \in \{-2, -1.75, -1.5, \ldots - 2\}, \forall k \in \{0, 1, \ldots, T-1\}. \tag{14}$$

E. Backup Controller (BUC)

The backup controller is a correction mechanism which ensures that the microgrid's constraints are always respected. The constraints come in the form of battery constraints. The BUC automatically starts a charging or discharging cycle if the capacity limits are violated or automatically changes the charge/discharge rate if the limits are exceeded. The settings of the backup controller are unknown to the learning agent.

F. Cost Function

The aim of this work is to maximize the self-consumption of the locally produced energy from the PV system. This is reflected in the net energy costs, c, for energy exchanges with the main utility grid. The cost function, ρ is defined as:

$$\rho(s, a) = \lambda_{imp} E_{imp} + \lambda_{inj} E_{inj}, \tag{15}$$

λ_{imp} and λ_{inj} represent the price of importing or injecting a unit of energy during a 15-minute period ($\frac{kWh}{4}$) from or to the

Algorithm 1 Approximate policy iteration with Q-functions

Input: discount factor γ, optimization horizon T, number of iterations it

1: Initialize policy to $\tilde{h}_0$ and exploration schedule ϵ
2: **for** $i = 0$ to $it - 1$ **do**
3: Generate samples using policy, $\tilde{h}_i$ $\{(s_l, a_l, s'_l, c_l)|l = \{0, \ldots, F-1\}\}$
$\tilde{s}'_l \leftarrow (s^d, s^q, SoC, s'_x)$.
4: **for** $l = 0$ to $F - 1$ **do**
5: Compute $Q^{\tilde{h}_i}_{i+1}$, the Q-function of the policy $\tilde{h}_i$
$Q^{\tilde{h}_i}_{i+1}(s_l, a_l) = c_l + \gamma \tilde{Q}^{\tilde{h}_i}_i(s'_l, \tilde{h}_i(s'_l, a))$,
6: **end for**
7: Use a regression algorithm to build $\tilde{Q}^{h_i}_{i+1}$ from $\mathcal{TS} = \{((s_l, a_l), Q^{\tilde{h}_i}_i), l = \{0, \ldots, F-1\}\}$
8: Improve the current policy $\tilde{h}_i$ to π using $\tilde{Q}^{h_i}_i$
$\pi = \arg\min_{\forall a \in \mathcal{A}} \tilde{Q}^{\tilde{h}_i}_i(s, a)$
ϵ-greedy exploration for policy selection:
9: Update ϵ and perform ϵ-greedy policy selection

$$\tilde{h}_{i+1} \leftarrow \begin{cases} \pi \text{ with probability} 1 - \epsilon \\ \text{random policy with probability } \epsilon. \end{cases}$$

10: **end for**

Output: $\tilde{h}^* = \tilde{h}_{it-1}$

grid, respectively, and E_{imp} and E_{inj} represent the amount of energy in $\frac{kWh}{4}$ imported from or injected to the utility grid respectively. The net cost, $c = \rho(s, a)$.

III. Control Strategy - Policy Iteration

In this work, an approximate policy iteration algorithm with Q-functions [12] is considered. A batch of data samples in the form of 4 tuples $\mathcal{F} = \{(s_l, a_l, s'_l, c_l)|l = 0, \ldots, F-1\}$ is generated using a policy, $\tilde{h}$. These data samples represent observations from the microgrid when under the control of the policy $\tilde{h}$. We developed an accurate microgrid simulator based on the model described in section II, with which the RL interacts to generate the data samples.

Algorithm 1 describes the policy iteration algorithm proposed in this work. An initial policy for example, a policy that discharges the battery all the time in order to reduce energy cost is considered. This policy is iteratively evaluated by constructing its value function. An improved policy greedy in the Q-function is computed at each iteration. A training set ($\mathcal{TS}$) is iteratively constructed with all state-action pairs (s, a) in $\mathcal{F}$ as the input, as well as the targeted values consisting of the corresponding Q-values. The Q-values are based on approximations of the Q-function from the previous iteration, for the next states and all actions, $\min_{a \in \mathcal{A}} \tilde{Q}(s', a)$.

At every iteration, approximations of the Q-function are obtained by employing a function approximator in the form of a regressor function. In this work, we have opted to use extremely randomized trees [13] to build $\tilde{Q}^{h_i}_i(s, a)$. Thus, Algorithm 1 uses the function approximator's generalization ability to remedy the curse of dimensionality problem encountered with large or continuous state/action spaces.

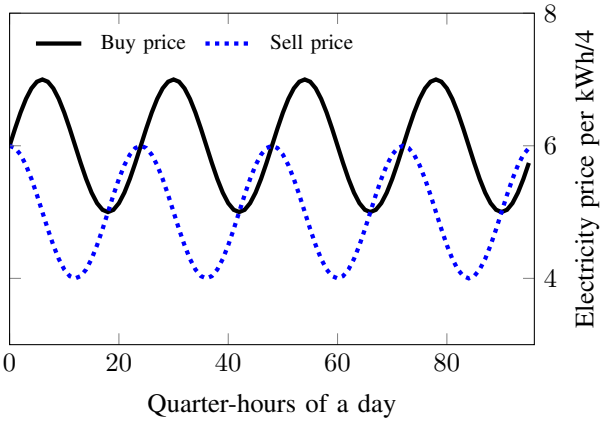

Figure 3. Sinusoidal electricity price profile with four periods to consider the effect of similar price variations during different periods of the day.

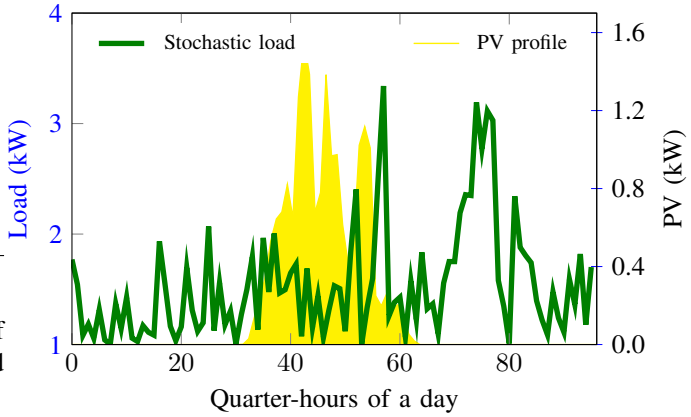

Figure 4. Load and PV profiles for a winter day. The load profile is derived from data of Belgian residential consumers with a noise signal added to introduce uncertainty.

Algorithm 1 also considers an ϵ-greedy policy exploration strategy. This allows the agent to exploit its current policy as well as explore other possible policies on the system in order to obtain a control policy with a better performance. We use a decreasing value of ϵ over the number of iterations.

IV. Results

In this section, we present the case study and simulation results using load and PV generation profiles from Belgian residential consumers. The performance of the proposed method is compared with the fitted Q-iteration algorithm, another RL control method, is also discussed in this section.

A. Case study

The simulation considers the single-user microgrid discussed in section II. A 40 kWh battery of efficiency $\eta = 90\%$, $SoC_{max} = 0.9$ and $SoC_{min} = 0.2$ is considered, with an initial SoC of 0.85. The maximum charging rate of the battery is set at $P_{max} = 2kW$, while the minimum discharge

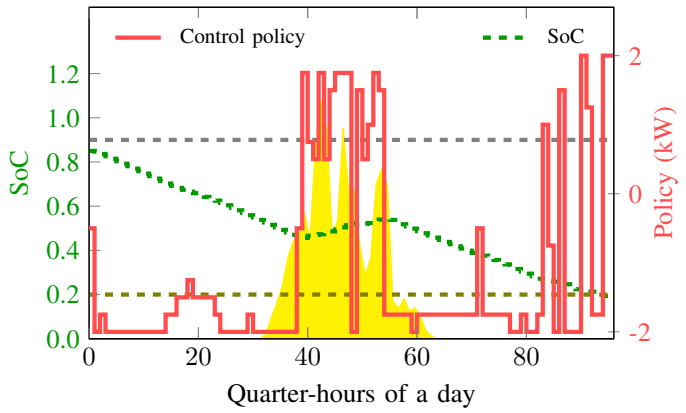

Figure 5. Typical control policy on a winter day. The yellow background area represents the PV production.

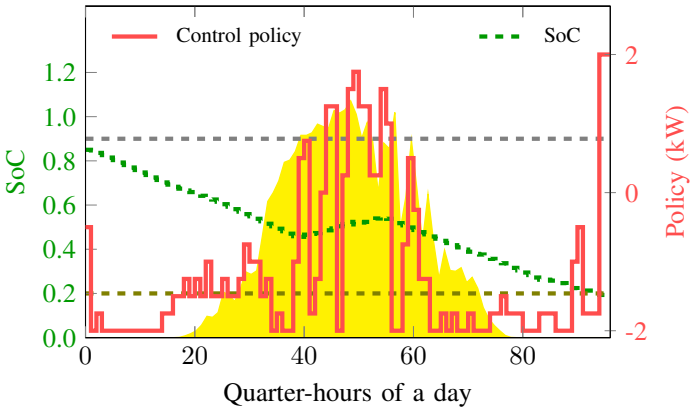

Figure 6. Typical control policy on a summer day. The yellow background area represents the PV production.

rate is set at $P_{min} = -2kW$. A 4 kW AC/DC inverter is considered with efficiency profile as shown on Fig. 2. This work focuses on energy optimization of the battery in the microgrid setting, and as such voltage and frequency control of the microgrid are not considered as part of this work. The energy from the PV system can be used to simultaneously charge the battery and supply the load. However, the option of charging the battery directly from the grid even during periods of very low electricity prices is not considered. We consider a sinusoidal electricity price signal as illustrated on Fig. 3. The price profile considers situations during which the price of buying electricity is lower than the price of selling energy, a possible scenario in the imbalance market. The load and PV generation profiles consist of data from Belgian residential consumers. This is real world data which has been generated from some unknown stochastic process. A noise signal is introduced in the deterministic load profile to have additional uncertainty on the load. An example of a load and PV profile considered in this work is shown on Fig. 4. The battery energy management problem is treated as a discrete-time problem and a time step of 15 minutes is considered for the simulations i.e. the learning agent takes control actions on the operation mode of the battery every 15 minutes. The inverter's non-linearity,

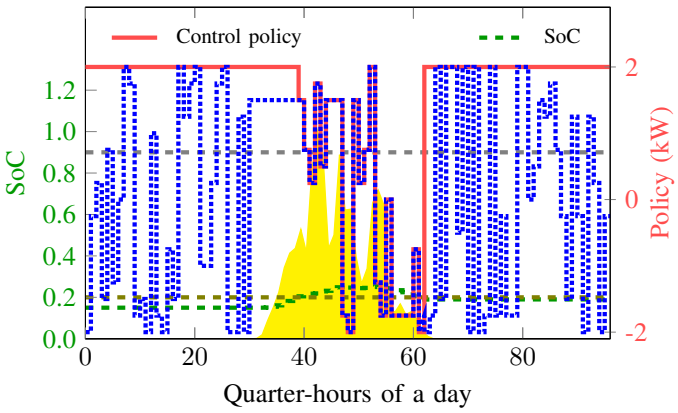

Figure 7. Typical control policy on a winter day with the effect of the BUC clearly shown. The dotted blue plot shows the policy learned by the agent. The yellow background area represents the PV production.

the stochastic nature of the residential load, the PV energy generated, electricity price and the battery's SoC are relevant quantities of the system that the RL agent learns to build a control policy.

B. Control policy

Figures 5 and 6 illustrate learned control policies for a winter and summer day respectively, corrected by the backup controller. The control policy is learned from an initial policy that charges the battery at all times to build an energy reserve. This policy shows the preferred action over the state space. We can see that with this policy, the agent avoids discharging the battery during periods of high PV production. This is because there is enough energy to charge the battery and supply the load simultaneously. Also, during periods of high PV production, depending on the battery's SoC, the battery is often charged at its maximum charging rate to maximize PV self-consumption, and reduce the amount of energy sold to the grid. As such, due to the uncertainty on the future production and load consumption patterns, the battery builds up its reserve during periods of high PV production. This permits to reduce the quantity of energy bought from the grid during periods of less PV production, peak demand, and/or high electricity prices. This effect can be clearly seen from the control policies. To avoid energy wastage, the RL agent ensures that the battery is discharged to SoC_{min} at the end of the optimization horizon. The effect of the non-linearity due to the inverter's efficiency can be seen where the battery discharges in the presence of PV production in order to achieve a high inverter efficiency to supply the load. From the figures, we can also notice that at the end of the optimization horizon, the backup controller requests to charge the battery at maximum charging rate P_{max}. This is because any attempt to discharge the battery leads to the violation of the capacity constraint. Simulations were also carried out for different initial SoCs for the battery and adequate results were obtained.

Figure 7 shows the control policy when the battery has an initial SoC of 0.15. This initial SoC is less than SoC_{min}, and as such, whatever action the learning agent takes is overruled by the BUC and an action to charge the battery at P_{max} is applied on the system. This can be seen during the first 32 quarter-hours of the day. However, the SoC remains fixed as there is no PV production. It is also important to note that the actual charging rate of the battery is the minimum of the action taken by the agent and the PV generated at that time step.

C. Performance evaluation

To evaluate the performance of the proposed control algorithm with the fitted Q-iteration proposed in our previous work, some performance indicators are considered for both cases. An optimization horizon ranging from January - February is considered. An initial SoC of 0.85 is considered.

Table I presents the performance indicators considered in this work. A 12% increase in the PV captured by the battery shows the effectiveness of the proposed algorithm with respect to the fitted Q-iteration. This is reflected in the 30% reduction in the net electricity costs paid by the microgrid user. This shows that a greater part of the PV generated during peak production is stored and reused to supply the load during periods of peak load and low PV production. The fitted Q-iteration is a purely deductive algorithm i.e. each value function is guaranteed to have a certain level of accuracy and used to deduce the policy. As such, any errors in the Q-function will have a direct impact on the control policy. Policy iteration on the other hand provides a policy at the end of the optimization period instead of a Q-function. However, the policy evaluation step in Algorithm 1 leads to additional computations and thus, a 91% increase in the training time of the RL agent.

TABLE I
PERFORMANCE INDICATORS FOR COMPARING THE POLICY ITERATION AND FITTED Q-ITERATION ALGORITHMS IN THE MICROGRID SETTING.

	Policy iteration	Fitted Q-iteration
PV captured by battery	73%	61%
Net electricity costs	45 euros	76 euros
Training time	6.1 hours	3.2 hours

V. Conclusion

This work has introduced the use of a RL method, policy iteration, together with extremely randomized trees and an epsilon-greedy policy selection to address the problem of managing the energy in a battery in a stochastic microgrid environment. This battery energy management has been done by scheduling the operation mode of the battery in the microgrid. Simulation results show that the policy iteration algorithm, when used with a supervised learning algorithm and a suitable policy selection algorithm can efficiently provide a control strategy that, efficiently manages unexpected changes in the load and PV production profiles to maximize PV self-consumption with a battery. This is seen with the 12% increase in the PV captured by the battery for reuse during peak consumption. The proposed control algorithm is a viable alternative to the fitted Q-iteration algorithm which we discussed in previous work.

Future work will investigate scenarios in which several single-user microgrids interact with each other and with the main utility grid. It would also be interesting to consider another type of supervised learning algorithm such as neural networks for the approximation of the Q-function.

VI. Acknowledgement

This research is supported by Vlaamse Instelling voor Technologisch Onderzoek (VITO) and partly funded by IWT-SBO-SMILE-IT, the Flemish Agency for Innovation through Science and Technology, promoting Strategic Basic Research.

References

[1] C. Bertrand, E. Damien, L. Warichet, and W. Legros, "Efficient management of a connected microgrid in belgium," in *24th International Conference on Electricity Distribution (CIRED)*, 2017. [Online]. Available: http://hdl.handle.net/2268/211726

[2] A. Kanwar, D. I. H. Rodríguez, J. von Appen, and M. Braun, *A Comparative Study of Optimization-and Rule-Based Control for Microgrid Operation*. Universitätsbibliothek Dortmund, 2015.

[3] V. C. J. Sankar, M. Raghunath, and M. G. Nair, "Optimal scheduling and energy management of a residential hybrid microgrid," in *2017 Innovations in Power and Advanced Computing Technologies (i-PACT)*, April 2017, pp. 1–6.

[4] L. M. Costa and G. Kariniotakis, "A stochastic dynamic programming model for optimal use of local energy resources in a market environment," in *2007 IEEE Lausanne Power Tech*, July 2007, pp. 449–454.

[5] A. Parisio, E. Rikos, and L. Glielmo, "A model predictive control approach to microgrid operation optimization," *IEEE Transactions on Control Systems Technology*, vol. 22, no. 5, pp. 1813–1827, Sept 2014.

[6] S. Aittahar, V. François-Lavet, S. Lodeweyckx, D. Ernst, and R. Fonteneau, "Imitative learning for online planning in microgrids," in *International Workshop on Data Analytics for Renewable Energy Integration*. Springer, 2015, pp. 1–15.

[7] Q. Wei, D. Liu, F. L. Lewis, Y. Liu, and J. Zhang, "Mixed iterative adaptive dynamic programming for optimal battery energy control in smart residential microgrids," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 5, pp. 4110–4120, May 2017.

[8] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA: MIT Press, 1998.

[9] V. François-Lavet, D. Taralla, D. Ernst, and R. Fonteneau, "Deep reinforcement learning solutions for energy microgrids management," in *European Workshop on Reinforcement Learning (EWRL 2016)*, 2016.

[10] B. V. Mbuwir, F. Ruelens, F. Spiessens, and G. Deconinck, "Battery energy management in a microgrid using batch reinforcement learning," *Energies*, vol. 10, no. 11, p. 1846, 2017.

[11] E. Kuznetsova, Y.-F. Li, C. Ruiz, E. Zio, G. Ault, and K. Bell, "Reinforcement learning for microgrid energy management," *Energy*, vol. 59, pp. 133–146, 2013.

[12] L. Busoniu, R. Babuška, B. De Schutter, and D. Ernst, *Reinforcement Learning and Dynamic Programming Using Function Approximators*. Boca Raton, FL: CRC Press, 2010.

[13] P. Geurts, D. Ernst, and L. Wehenkel, "Extremely randomized trees," *Machine Learning*, vol. 63, no. 1, pp. 3–42, 2006.