# Learning to control a battery through reinforcement
## Balancing lifetime and profit

Neves, Catarina Santos; Čović, Nikolina; Cremer, Jochen L.

**DOI**
10.1109/PowerTech59965.2025.11180449

**Publication date**
2025

**Document Version**
Final published version

**Published in**
2025 IEEE Kiel PowerTech, PowerTech 2025

# Learning to control a battery through reinforcement: balancing lifetime and profit

Catarina Santos Neves
*Dep. of Electrical Sustainable Energy*
*Delft University of Technology*
Delft, Netherlands
c.s.neves@outlook.pt

Nikolina Čović
*Dep. of Energy and Power Systems*
*University of Zagreb*
Zagreb, Croatia
Nikolina.Covic@fer.hr

Jochen L. Cremer
*Dep. of Electrical Sustainable Energy*
*Delft University of Technology*
Delft, Netherlands
j.l.cremer@tudelft.nl

2025 IEEE Kiel PowerTech | 979-8-3315-4397-6/25/$31.00 ©2025 IEEE | DOI: 10.1109/PowerTech59965.2025.11180449

***Abstract*—Battery energy storage systems offer control over energy use and enable energy arbitrage (EA) helping to lower energy costs. However, battery owners currently fail to optimally exploit these systems for EA as the battery lifetime decreases, and many EA approaches incorrectly assume constant battery capacity. Battery performance declines over time resulting in reduced capacity that limits the economic benefits. Therefore, considering battery degradation is key to balancing economic profit and lifetime. In response, this work applies reinforcement learning to control a battery providing residential EA services and proposes a semi-supervised learning model to consider degradation. Case studies investigate three scenarios: 1) the approach is trained on a battery with an unrealistic constant maximum capacity to serve as a baseline, 2) the actions from the first scenario are applied to a real-world environment with a battery experiencing capacity decay to acknowledge the effect of neglecting degradation and 3) the approach considers a battery with a real decreasing capacity. Results show not considering degradation when operating a battery (scenario 2), leads to profits 13% lower than the ones obtained in the ideal case (scenario 1). If degradation is considered (scenario 3), the profits are only 4% lower than the profits obtained in the ideal case (scenario 1) and the battery's lifetime is extended by 20% compared to the lifetime achieved when not considering degradation (scenario 2).**

***Index Terms*—reinforcement learning, energy storage system, energy arbitrage, battery degradation, state of health.**

## I. Introduction

RISING global demands for energy, especially renewable energy, and growing energy costs have sparked interest in investments in battery energy storage systems (BESSs) [1]. BESSs provide homeowners with higher control over energy use by storing excess energy generated from renewable sources for use during periods of high demand or when there is no renewable generation. More relevant to this work, BESSs can also be used for energy arbitrage (EA). Through EA, storing energy during off-peak hours and using and/or selling it during peak hours, homeowners can reduce energy expenses [2]. BESSs increase household independence from the traditional energy grid, leading to a more efficient, reliable grid, with less congestion issues and outages.

Additionally, over time, as BESSs age, their maximum energy storage capacity decreases. This capacity drop is measured by the battery's state of health (SOH). Models with constant SOH that neglect degradation are far from accurately fitting the experimental data. Numerous works have tried to create accurate degradation models, either considering physical factors contributing to degradation [3], or adopting physics-agnostic approaches [4]. The current most relevant physics-based battery model is the Doyle-Fuller-Newman (DFN) model [5], an electrochemical model that can be extended to consider different degradation mechanisms. This DFN model's equations can be solved by the Python Battery Mathematical Modelling package (PyBaMM) [6]. PyBaMM is an open-source modeling and simulation Python package, featuring built-in numerical solvers. In [4], PyBaMM solves the DFN model coupled with four degradation mechanisms. In contrast to these physics-based battery models, empirical models are presented as less computationally expensive alternatives. In [7], parameters of the battery's electrical circuit, such as the open-circuit voltage and internal resistance, are estimated over time to determine the state of degradation. In the realm of Machine Learning (ML), various techniques have been used. [8] and [9] use neural networks (NNs) to exploit historical data and estimate the SOH of electric vehicle batteries cycled arbitrarily. [8] uses current, voltage and recurrent NNs (RNNs) and estimates the SOH with an average error lower than 2.46%. [9] uses the same inputs as [8], along with temperature, and feedforward NNs (FNNs) instead of RNNs to estimate the SOH with an average error lower than 2.18%.

However, homeowners cannot keep track of their batteries' performance and degradation state, nor adjust their behavior to benefit the devices' efficiency and longevity. While modern batteries typically include battery management systems (BMSs), these lack advanced degradation models and robust control models to optimize operation based on energy price dynamics. Integrating smart meters, as well as BMSs capable of incorporating control models for optimal decision-making could be a solution. One of the available methods for building such control models is reinforcement learning (RL) [10], a powerful decision-making tool in unknown and complex environments that has been widely applied in this area of research [11]. In [12], Deep Reinforcement Learning (DRL) performs EA applying a semi-empirical battery degradation model, approximating degradation as a linear function of the number of cycles performed during a short period. In [13], Deep Q-Learning (DQL) optimizes the operation of a BESS doing EA and providing frequency regulation services. Again, the

degradation cost is defined as being linearly dependent on the amount of performed cycles. In [14], a DRL approach, based on proximal policy optimization (PPO), optimizes the schedule of a PV-BESS system performing frequency regulation, EA and PV charging considering degradation and costs.

In the literature, several studies operate batteries for EA using RL [15], however, only a few consider degradation [12] which motivates this work. Among those considering degradation, most lack research on more detailed models that better capture the progression of degradation over time. Specifically, research on the nuanced impact of degradation on BESSs' profitability is limited, as degradation costs are often defined as constant per cycle. Moreover, there is a gap in understanding the potential benefits of operating BESSs with awareness of degradation. Addressing the identified gap requires determining a degradation model that seamlessly integrates into an RL framework and defining the RL framework itself. The RL framework includes environments for battery operation both with and without degradation, to evaluate its impact. Therefore, this paper aims to answer the following research questions:

- How do different degradation models compare for online degradation estimation?
- What is the impact of degradation on the lifetime and profitability of a BESS?
- Does considering degradation reduce its negative effects on the BESS's lifetime and profitability of its operations?

To answer these questions, this work explores a residential EA problem considering energy market price fluctuations and battery degradation. Degradation is included through an empirical model that calculates SOH in near real-time. The case study investigates three scenarios: one where degradation is unknown to the BMS, a second one where degradation is known and a third one that results from taking the actions obtained in the first scenario, applying them in a real-world environment with a battery experiencing capacity decay and calculating the realistic output. The analysis of the different scenarios demonstrates the developed model increases the battery's lifetime and overall profitability.

The paper is organized as follows: Section II provides insight into battery degradation theory and modelling; Section III provides theoretical background on RL and a description of the proposed RL algorithm; In Section IV, the results of a case study are presented; Section V provides overall conclusions.

## II. Battery Degradation Modelling

Battery capacity can be measured by the value of SOH [16]:

$$SOH_t = \frac{C_t}{C_{rated}} \times 100\%, \tag{1}$$

where $C_t$ is the capacity at time period $t$, and $C_{rated}$ is the rated capacity. With time, the rated capacity reduces due to battery degradation, which can be described by certain mechanisms related to the physical and chemical changes occurring within a cell. There are five lithium-ion battery degradation mechanisms, namely, solid electrolyte interphase (SEI) layer growth and lithium plating. SEI layer growth arises from reactions between the electrolyte and fresh carbon surfaces, forming a layer on the negative electrode that thickens over time, especially under high temperatures and currents [17]. Lithium plating occurs as a side reaction when metallic Li forms a layer on the negative electrode's surface, instead of intercalating into it [18]. Degradation mechanisms result in capacity fade, which does not detail the physical processes in the cell but is easier to measure. A battery reaches its end of life (EOL) when the SOH value is below 80%.

Battery degradation modelling can be split into two main approaches: empirical and physics-based. The former involves deriving equations and parameters directly from experimental data, while the latter relies on equations known to describe the underlying physics of degradation mechanisms. More recently, ML models have emerged as empirical models with improved flexibility to deal with complex non-linear relationships within the experimental data [3]. Given the variety of approaches to battery degradation modeling, defining the most suitable model *a priori* is challenging. This paper works with three different models — one physics-based model and two empirical ones — that balance well simplicity and accuracy:

*1) PyBaMM Model:* As described, PyBaMM can solve the DFN model's equations, coupled with degradation mechanisms such as SEI layer growth and lithium plating, the most significant calendar and cycling ageing mechanisms [4].

*2) Supervised Learning Model:* A simple modeling approach assumes a constant degradation rate. In [19], the loss of SOH is deemed constant per cycle, with the loss rate depending on the cycling C-rate. The model uses a piece-wise linear function, derived from a data fit, which takes the cycling C-rate as input and outputs the percentage of SOH degradation caused by that cycle.

*3) Semi-Supervised Learning Model:* A third model, formulated in [9], is a semi-supervised learning approach. A NN takes current ($I$) and voltage ($V$) data (retrieved from a BMS) as input and provides the SOH (%) as output. Current and voltage data are directly related to SOH in two ways: high current and voltage accelerate degradation, while a higher density of current and voltage samples indicates frequent cycling and, consequently, increased degradation.

Using k-means clustering, historical distributions of ($I$,$V$) samples are grouped into $k$ clusters. Then, the input to the NNs is a 1-dimensional array with $k$ elements, corresponding to the density of points in each cluster. The model is trained on these historical distributions and then the model can trace SOH in real-time, by clustering new current and voltage samples.

## III. Reinforcement Learning to Operate a Battery

An RL problem is typically formulated as a Markov Decision Process (MDP), a framework for sequential decision-making modelling states and actions describing the transitions between states. An MDP is defined by an action space $\mathcal{A}$, state space $\mathcal{S}$, transition probability function $p$ and reward function $r$. An episode of an MDP consists of a finite sequence of timesteps $t$ from 0 to $T$.

In RL, an agent observes the environment's state, selects an action and, in turn, receives the reward $r$ and transits to the next state. The agent aims to learn the optimal policy and maximize the cumulative discounted return, by considering immediate and future rewards multiplied by a discount factor $\gamma$. Deep Q-Network (DQN) is a variant of DRL able to deal with large or continuous states spaces. Using NNs, DQNs take state as input and output an estimate of the Q-values of state-action pairs, $q(s, a)$. Once trained, the agent can effectively act under policy $\pi$:

$$\pi(s) = \max_a q(s, a) \tag{2}$$

When training such a DQN agent, the $\epsilon$-greedy strategy initially selects actions. In this strategy the agent randomly selects an action with probability $\epsilon$, decaying at rate $\delta$. With this strategy, extensive exploration is enabled early during training and policy refinement occurs later on.

### A. RL Environment

The battery operation model considered in this paper is a combination of the ones presented in [19] and [20]:

$$0 \leq \eta^{ch} \cdot ch_t \leq P^{ch}, \ \forall t \in \mathcal{T} \tag{3}$$

$$0 \leq dis_t / \eta^{dis} \leq P^{dis}, \ \forall t \in \mathcal{T} \tag{4}$$

Constraints (3) and (4) regard charging and discharging of BESSs. $P^{ch}$ ($P^{dis}$) is the maximum charging (discharging) power, in kW. Furthermore, $ch_t$ ($dis_t$) is defined as the power bought from (sold to) the grid, in kW, and $\eta^{ch}$ ($\eta^{dis}$) is the charging (discharging) efficiency, with value between 0 and 1.

$$e_t^{ch} = \eta^{ch} \cdot ch_t \cdot \Delta t, \ \forall t \in \mathcal{T} \tag{5}$$

$$e_t^{dis} = dis_t / \eta^{dis} \cdot \Delta t, \ \forall t \in \mathcal{T} \tag{6}$$

Equations (5) and (6) define the charged ($e_t^{ch}$) and discharged energy ($e_t^{dis}$) in kWh at each timestep $t$.

$$0 \leq \underline{SOC} \leq soc_t \leq \overline{SOC}_t \leq SOH_t, \ \forall t \in \mathcal{T} \tag{7}$$

Constraint (7) governs the battery's state of charge ($soc_t$, in %), which must always be within the bounds $\underline{SOC}$ and $\overline{SOC}_t$, the minimum and maximum SOC levels the battery is allowed to operate at, respectively. In turn, these bounds must be $\geq 0$ and $\leq SOH_t$. $\underline{SOC}$ will be kept constant and thus is not dependent on $t$. On the other hand, $\overline{SOC}_t$ will be defined in relation to $SOH_t$, which will be made to decrease with $t$, for the case when the agent is aware of degradation, and will be kept constant, for the case where the agent is unaware.

$$\Delta soc_t = (e_t^{ch} - e_t^{dis}) / C_{rated} \cdot 100\%, \ \forall t \in \mathcal{T} \tag{8}$$

$$soc_{t+1} = soc_t + \Delta soc_t, \ \forall t \in \mathcal{T} \tag{9}$$

Equation (9) updates the battery's state of charge based on the energy charged or discharged in that timestep, as defined by (8), where $C_{rated}$ is the rated capacity (in kWh).

The financial balance (fb) is defined as the difference between income stemming from discharging and the costs of charging. For episode $i$, fb is given by

$$\text{fb}_i = \sum_{t=0}^{23} p_t (e_t^{dis} - e_t^{ch}) \tag{10}$$

where $p_t$ is the energy price (in cents/kWh) in timestep $t$. A positive financial balance indicates a profit from EA.

The state space includes five elements:

$$S_t = (soc_t, \text{diff}_t, SOH_t, s_t^{\text{fb}}, step) \tag{11}$$

- $soc_t$: state of charge at timestep $t$ - this term allows the agent to make decisions that will not violate constraint (7);
- $\text{diff}_t = \tilde{p_i} - p_t$: the difference between the current price $p_t$ and the daily median $\tilde{p_i}$ - this term tells if the current price is lower or higher compared to the daily median that this work assumes to be known in advance;
- $SOH_t$: state of health at timestep $t$ - this term can either indicate the degradation state of the battery by varying over time or remain fixed at its maximum value;
- $s_t^{fb}$: binary variable - this term is 0 for a negative financial balance at timestep $t$ and 1 otherwise;
- step: binary variable - this term is 1 in the final timestep of each episode and 0 otherwhise.

The action space consists of a set of discrete energy amounts for charging or discharging. At each timestep $t$, the agent selects an energy action from a range of values between -$E_{max}^{dis}$ (discharge) and +$E_{max}^{ch}$ (charge), with 0 indicating no action. The action space is mathematically represented as $\mathcal{A} = (-E_{max}^{dis}, ..., 0, ..., E_{max}^{ch})$, where -$E_{max}^{dis}$ ($E_{max}^{ch}$) corresponds to discharging (charging) at maximum discharging (charging) power $P^{dis}$ ($P^{ch}$) for a period of time $\Delta t$.

The reward function $r$ =

$$\begin{cases} -3 & \text{if } soc_t < \underline{SOC} \vee soc_t > \overline{SOC_t} \quad (12) \\ 2 \cdot \dfrac{\text{diff}_t}{\text{diff}_{\max}} \cdot \left( \dfrac{ch_t}{\overline{SOC_t}} - \dfrac{dis_t}{\overline{SOC_t}} \right) & \\ \qquad \text{if } \underline{SOC} < soc_t < \overline{SOC_t} \quad (13) \\ -2 & \text{if } t = 23 \wedge |SOC_{24} - SOC_0| > 10\% \quad (14) \\ -2 & \text{if } t = 23 \wedge \text{fb}_i < 0 \quad (15) \end{cases}$$

has four terms designed to ensure the battery's functioning:

- SOC limits: a penalty is given for going over $\overline{SOC_t}$ or under $\underline{SOC}$, which in a real-world application is impossible;
- price sensitivity: a term is defined to reward charging (discharging) events when current prices are lower (higher) than the daily median and to penalize otherwise, ensuring efficient battery operation for EA;
- continuity preservation: a reward is given if the $soc_t$ in the first and final timestep are equal and a penalty is given otherwise, to maintain continuity between days, ensuring no energy is lost or gained;
- profit maximization: the agent is penalized if the net financial balance at the final timestep is negative, to maximize the profits of EA.

This paper defined these numerical values based on the following rationale that considers the importance in real-world

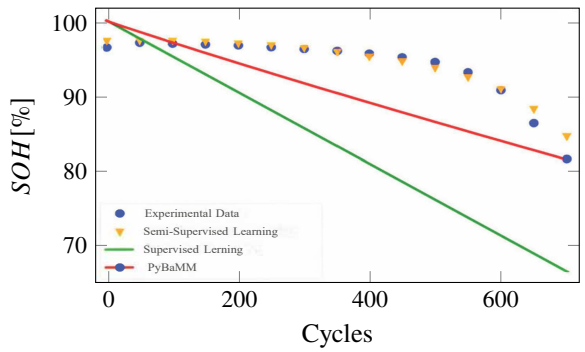


Fig. 1: Accuracy of the three degradation models.

implementation; however, these values can be tuned and here serve as examples only. The algorithm's primary goal is for the battery to operate within its physical limitations for SOC. Thus, the penalty for going over the SOC limits (12) is the largest. The next goal is to optimize battery operation for EA. The agent receives a reward for charging (discharging) the battery at low (high) prices and a penalty otherwise, but only when it respects the SOC limits (13). Penalties for violating SOC limits are harsher than for suboptimal (dis)charging, prioritizing physical constraints over financial gains. In (14), note that the discrete action space, combined with (dis)charging efficiencies, can challenge the agent to charge and discharge equal amounts in one episode. Thus, we introduce a 10% margin where no penalty is given. Finally, as in the final timestep of each episode ($t = 23$) we consider equally important to have $SOC_{24} = SOC_0$ and $fb_i \geq 0$, the penalty for violating either condition is the same and equal to $-2$ as demonstrated in (14) and (15).

## IV. Case Study

### A. Settings and Model Assumptions

The case study considers the battery ageing models from Section II. These models were tested on data from lithium iron phosphate (LFP) cells cycled to failure under various charging policies [21]. The performance of these degradation models was evaluated on computational time and prediction accuracy. The root mean square error (RMSE) gives the difference between predicted and real values.

The RL environment considers the battery to perform EA based on hourly price data for July 2023 in Portugal, retrieved from [22] and normalized to a range of 0 to 10. The battery's SOH was predicted using the semi-supervised learning model, using voltage and current data as input to the NN. Charged/discharged energy is related to the battery's current ($I$) and voltage ($V$) through the following equation: $e^{ch(dis)} \approx I \cdot \frac{(V_{init}+V_{final})}{2} \cdot \Delta t$ [23]. With the possible values of charged and discharged energies defined in the action space, either $V$ or $I$ is needed to calculate the remaining parameter. Thus, voltage was calculated from [24]'s model for constant current that relates SOC changes to voltage changes during charge/discharge events at 0.2C intervals from 0 to 1C, where C is C-rate. The environment modelled a *Tesla Powerwall*, a home battery with a rated capacity of 7 kWh (17.5 Ah) and a nominal voltage of 400V. The action space represents the energy change in the battery when cycling at the stated C-rates for 1 hour, ranging from $-7.0$ kWh to 7.0 kWh. The learned agent was analysed under three scenarios:

- **scenario 1:** the agent is trained observing a constant maximum value for $SOH_t$ (100%) and $\overline{SOC_t}$ (80%)[1].
- **scenario 2** is fabricated from scenario 1, i.e., without training, by *a posteriori* using the actions obtained from scenario 1 in a real-world environment. In this environment $SOH_t$ and $\overline{SOC_t}$ decreases due to degradation, to acknowledge the effect that neglecting degradation can have on the battery operation and lifetime.
- **scenario 3:** the agent is trained observing changes in $SOH_t$ and $\overline{SOC_t}$ and learns to act accordingly.

The DQN, implemented with PyTorch [25], is a FNN with an input layer of 5 neurons (state space), two hidden layers of 32 neurons and one output layer of 11 neurons (action space). The learning rate $\alpha$, discount rate $\gamma$ and $\epsilon$-decay rate $\delta$ were 0.001, 0.9 and 0.999, respectively. These values were obtained through grid search. Each training and testing process ran until SOH dropped below 80%, in episodes with $T = 24$ timesteps of 1 hour, and took 25 minutes each on a 1.80 GHz, Intel 4-core i7-8550 CPU with 16 GB RAM, running Windows 11. All tables show average values over 10 runs of the training and testing procedures.

TABLE I: Computational time and accuracy of the three models

| Model | Comp. time [sec/cycle] | RMSE [%] |
|---|---|---|
| PyBaMM | 1.0 | 5.3 |
| Supervised Learning | $1.0 \times 10^{-4}$ | 12.0 |
| Semi-Supervised Learning | 0.4 | 2.5 |

### B. Selecting Degradation Model

This case study investigates the performance (accuracy and computational time) of the three models for degradation to select the best one. Table I shows the supervised learning model is the fastest, taking just 0.1 milliseconds to compute the SOH after each cycle. However, the semi-supervised learning model shows significantly higher accuracy than the supervised model. Figure 1 shows the semi-supervised predictions follow closely the experimental data (ground truth), especially when compared to the other models. Thus, the semi-supervised learning model is selected to be included in the RL environment having the best accuracy and a low computational time of just 0.4 seconds per cycle.

### C. Balancing Profit and Battery Maintenance

This case study investigates the balance between financial profit and the battery lifetime. Figure 2 shows the hourly price, in eurocents (¢) per kWh, and the SOC response of a day when $\overline{SOC_t}$ is nearing 64%, i.e., the battery is near its EOL (when $\overline{SOC_t}$ drops below 80% x 80% = 64%), as indicated by the

[1]The model assumes charging and discharging at constant current. In real world applications, batteries are typically cycled this way only up to 80% SOC. Above this value, batteries are usually cycled at constant current constant voltage (CC/CV). Thus, $\overline{SOC_t}$ is set at 80 % to ensure congruence with real world battery physics.

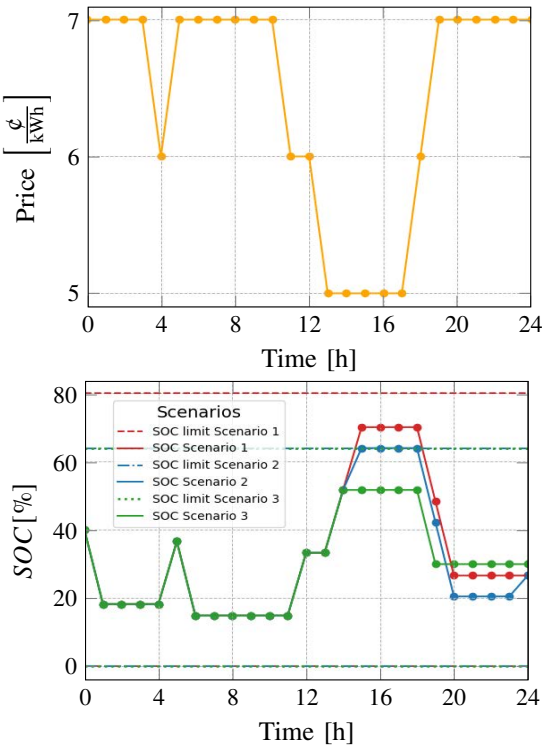


Fig. 2: SOC when using an aged battery. Top figure is the price and bottom figure the response from agents differing in degradation model scenarios.

upper SOC limits in scenarios 2 and 3 (blue and green dotted lines). The SOC response for scenarios 1 and 2 are represented by the red and blue lines in Figure 2, respectively. These overlap until 14:00h when the agent charges from about 55% to 70% as this action would be profitable if the battery had not aged yet. In reality, the agent only charges up to 64%, due to capacity decay. These occurrences become more frequent and important as the battery ages. Scenario 3, represented by the green line in Figure 2, shows that the agent strays away from charging the battery to the highest SOC possible as the agent is informed of the battery's state of degradation. Hence, the agent can achieve a better balance between profit and battery maintenance.

Figure 3 illustrates the SOH evolution through a testing run. Figure 3 shows the SOH responses obtained in scenarios 2 and 3 are far from constant as expected. Table II presents the average lifetime of the battery obtained from 10 repetitions of the experiments. The left column of Table II shows that the lifetime is higher in scenario 3, approximately 4948 days, compared to about 4065 days in scenario 2 as the agent charges/discharges less to avoid overcharging/discharging leading to lower voltages and currents, reducing degradation and prolonging the battery's lifetime.

### D. Maximising Profit

This case study investigates the financial profit that may be achieved with this approach. Figure 4 presents the average financial balance per episode for all scenarios. Initially, for a new battery, the financial balance per episode in scenarios 1 and 2 is the same. Later, in scenario 1, it remains

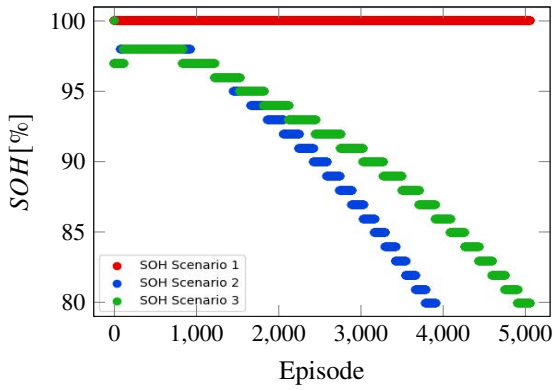


Fig. 3: SOH per episode

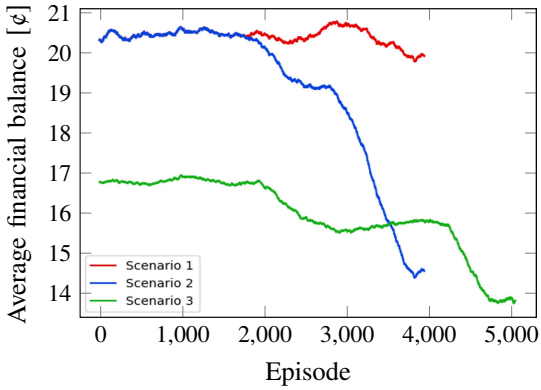


Fig. 4: Average financial balance per episode

approximately constant, because, if capacity stays fixed at its maximum value, the agent constantly exploits the same type of charge/discharge events. As a result, the battery's profitability remains largely unchanged over time, fluctuating only due to dynamic pricing. However, the same does not occur later in scenario 2. As degradation increases, the possible charged/discharged energy becomes lower, reducing the battery's profitability compared to the ideal case of scenario 1. In scenario 3, the financial balance per episode is lower due to the previously mentioned limitation on over-charging, further decreasing over time due to degradation. However, as the lifetime is longer in this scenario, the cumulative financial balance is greater - 82.37 compared to 83.53 x 1000 ¢ in scenario 2, as shown in Table II.

The degradation cost defines the profit difference between scenarios 1 and 2, when the agent is unaware of degradation, and the profit difference between scenarios 1 and 3 when the agent is aware. Given this definition, we conclude the degradation cost is lower when the agent is aware of degradation, indicating it has effectively mitigated financial losses.

### E. Evaluating Reward Function

This case study evaluates the agent's response to the four components of the reward (Equations (12) to (15)). We define four metrics: episodes per testing run where $SOC_{24} \neq SOC_0$; timesteps where $soc_t < \underline{SOC}$ or $> \overline{SOC_t}$; timesteps where a suboptimal event occurred (charge (discharge) at high (low) prices); and episodes with net negative financial balance. Table

TABLE II: Lifetime and total financial balance

| Scenario | Lifetime [days] | $\sum_i \text{fb}_i$ [× 1000 ¢] |
|---|---|---|
| 1 | 4065 ± 563 | 86.99 ± 6.95 |
| 2 | | 82.37 ± 6.43 (-5%) |
| 3 | 4948 ± 504 (+22%) | 83.53 ± 6.55 (-4%) |

TABLE III: Performance Metrics

| Scenario | $\text{SOC}_{24} \neq \text{SOC}_0$ | $\text{soc}_t < \underline{SOC} \vee \text{soc}_t > \overline{SOC_t}$ | suboptimal (dis)charge | $\text{fb}_i$ <0.0 |
|---|---|---|---|---|
| 1, 2 | 99.43% | 0.00% | 0.99% | 1.15% |
| 3 | 84.82 % | 0.00% | 1.89% | 3.66% |

III shows the average percentage of occurrences. The last three metrics' values are low or zero, showing the agent respects SOC limits, correctly (dis)charges, and maintains positive episodic financial balances. Yet, $\text{SOC}_{24} \neq \text{SOC}_0$ occurs in almost every episode, reducing the model's viability.

### *F. Combining With Backup Control*

This case study investigates correcting the limitation on the model's viability to avoid mismatches of $\text{SOC}_{24} \neq \text{SOC}_0$. We calculate the charged/discharged energy necessary in the last timestep for the constraint $\text{SOC}_{24}=\text{SOC}_0$ to be satisfied. In a real-life scenario, a backup controller could address this limitation in the algorithm. Using the calculated charge/discharge event required, we computed corrected lifetimes and total financial balances (cfb). Table IV shows the average values obtained. Scenario 3's lifetime is still about 20% higher than the one in scenario 2, equivalent to delaying a battery replacement by about 700 days or nearly 2 years. Regarding total financial balance, the value in scenario 3, 35.17 x 1000 ¢, remains higher than the 32.10 x 1000 ¢ in scenario 2. Using the previously stated definition of degradation costs, and as shown in Table IV, the costs decrease from 4.63 x 1000 ¢ (about 13% of 36.73 x 1000 ¢) when the agent is unaware of degradation to 1.56 x 1000 ¢ (about 4% of 36.73 x 1000 ¢) when the agent is aware of degradation. This indicates reduced degradation costs for scenario 3, resulting in profits that approach the potential profits in a scenario without battery degradation, such as scenario 1.

## V. Conclusion

This paper proposes a DQN algorithm to optimize battery operation for EA, considering battery degradation. Degradation is estimated through a semi-supervised learning algorithm, which outputs an SOH value after each charging or discharging event. The RL algorithm shows that, while battery degradation impacts the profitability of batteries, awareness of this degradation can help reduce degradation costs from 13% to 4% while also increasing the battery's lifetime by about 20%. Future work should provide a reward function that can effectively teach the agent to maintain continuity of SOC between days and should include a SOC range from 0 to 100%.

TABLE IV: Corrected lifetime and total financial balance

| Scenario | Corrected Lifetime [days] | $\sum_i \text{cfb}_i$ [× 1000 ¢] |
|---|---|---|
| 1 | 3423 ± 447 | 36.73 ± 4.88 |
| 2 | | 32.10 ± 4.28 (-13%) |
| 3 | 4110 ± 361 (+20%) | 35.17 ± 8.45 (-4%) |

## References


[1] S. Europe, "European market outlook for residential battery storage 2022-2026," 2022.

[2] C. Byrne and G. Verbic, "Feasibility of residential battery storage for energy arbitrage," in *2013 Australasian Universities Power Engineering Conference (AUPEC)*, 2013, pp. 1–7.

[3] J. S. Edge *et al.*, "Lithium ion battery degradation: what you need to know," *Phys. Chem. Chem. Phys.*, vol. 23, pp. 8200–8221, 2021.

[4] S. E. J. O'Kane *et al.*, "Lithium-ion battery degradation: how to model it," *Phys. Chem. Chem. Phys.*, vol. 24, pp. 7909–7922, 2022.

[5] M. Doyle, T. F. Fuller, and J. Newman, "Modeling of galvanostatic charge and discharge of the lithium/polymer/insertion cell," *Journal of The Electrochemical Society*, vol. 140, no. 6, p. 1526, 1993.

[6] V. Sulzer *et al.*, "Python Battery Mathematical Modelling (PyBaMM)," *Journal of Open Research Software*, vol. 9, no. 1, p. 14, 2021.

[7] Y.-H. Chiang, W.-Y. Sean, and J.-C. Ke, "Online estimation of internal resistance and open-circuit voltage of lithium-ion batteries in electric vehicles," *Journal of Power Sources*, vol. 196, no. 8, pp. 3921–3932, 2011.

[8] G.-W. You, S. Park, and D. Oh, "Diagnosis of electric vehicle batteries using recurrent neural networks," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 6, pp. 4885–4893, 2017.

[9] G.-W. You, S. Park, and D. Oh, "Real-time state-of-health estimation for electric vehicle batteries: A data-driven approach," *Applied Energy*, vol. 176, pp. 92–103, 2016.

[10] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA, USA: A Bradford Book, 2018.

[11] D. Zhang, X. Han, and C. Deng, "Review on the research and practice of deep learning and reinforcement learning in smart grids," *CSEE Journal of Power and Energy Systems*, vol. 4, no. 3, pp. 362–370, 2018.

[12] J. Cao *et al.*, "Deep reinforcement learning-based energy storage arbitrage with accurate lithium-ion battery degradation model," *IEEE Transactions on Smart Grid*, vol. 11, no. 5, pp. 4513–4521, 2020.

[13] Y. Miao *et al.*, "Co-optimizing battery storage for energy arbitrage and frequency regulation in real-time markets using deep reinforcement learning," *Energies*, vol. 14, no. 24, p. 8365, Dec 2021.

[14] B. Huang and J. Wang, "Deep-reinforcement-learning-based capacity scheduling for pv-battery storage system," *IEEE Transactions on Smart Grid*, vol. 12, no. 3, pp. 2272–2283, 2021.

[15] H. Wang and B. Zhang, "Energy storage arbitrage in real-time markets via reinforcement learning," in *2018 IEEE Power & Energy Society General Meeting (PESGM)*, 2018, pp. 1–5.

[16] D. Linden and T. Reddy, *Handbook of Batteries*, ser. McGraw-Hill handbooks. McGraw-Hill Education, 2001.

[17] S. Heiskanen, J. Kim, and B. Lucht, "Generation and evolution of the solid electrolyte interphase of lithium-ion batteries," *Joule*, vol. 3, 09 2019.

[18] X. Lin *et al.*, "Lithium plating mechanism, detection, and mitigation in lithium-ion batteries," *Progress in Energy and Combustion Science*, vol. 87, p. 100953, 2021.

[19] M. R. Sarker *et al.*, "Optimal operation of a battery energy storage system: Trade-off between grid economics and storage health," *Electric Power Systems Research*, vol. 152, pp. 342–349, 2017.

[20] H. Pandžić and V. Bobanac, "An accurate charging model of battery energy storage," *IEEE Transactions on Power Systems*, vol. 34, no. 2, pp. 1416–1426, 2019.

[21] K. A. Severson *et al.*, "Data-driven prediction of battery cycle life before capacity degradation," *Nature Energy*, vol. 4, pp. 383–391, 2019.

[22] "Ren data hub," https://datahub.ren.pt/en, accessed: 03-08-2023.

[23] C. Lin *et al.*, "Constant current charging time based fast state-of-health estimation for lithium-ion batteries," *Energy*, vol. 247, p. 123556, 2022.

[24] V. Bobanac, H. Bašić, and H. Pandžić, "One-way voltaic and energy efficiency analysis for lithium-ion batteries," in *13th Mediterranean Conference on Power Generation, Transmission, Distribution and Energy Conversion (MEDPOWER 2022)*, vol. 2022, 2022, pp. 261–266.

[25] A. Paszke *et al.*, "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems 32*. Curran Associates, Inc., 2019, pp. 8024–8035.