Article

Battery Sizing Optimization in Power Smoothing Applications

Asier Zulueta [1], Decebal Aitor Ispas-Gil [2], Ekaitz Zulueta [2,*], Joseba Garcia-Ortega [2]
and Unai Fernandez-Gamiz [1]

[1] Department of Nuclear and Fluid Mechanics, University of the Basque Country (UPV/EHU) Nieves Cano, 12, 01006 Vitoria-Gasteiz, Spain; azulueta@arrasate.eus (A.Z.); unai.fernandez@ehu.eus (U.F.-G.)

[2] System Engineering & Automation Control Department, University of the Basque Country (UPV/EHU) Nieves Cano, 12, 01006 Vitoria-Gasteiz, Spain; dispas001@ikasle.ehu.eus (D.A.I.-G.); jgarcia395@ikasle.ehu.eus (J.G.-O.)

* Correspondence: ekaitz.zulueta@ehu.eus; Tel.: +34-945-014-066

Abstract: The main objective of this work was to determine the worth of installing an electrical battery in order to reduce peak power consumption. The importance of this question resides in the expensive terms of energy bills when using the maximum power level. If maximum power consumption decreases, it affects not only the revenues of maximum power level bills, but also results in important reductions at the source of the power. This way, the power of the transformer decreases, and other electrical elements can be removed from electrical installations. The authors studied the Spanish electrical system, and a particle swarm optimization (PSO) algorithm was used to model battery sizing in peak power smoothing applications for an electrical consumption point. This study proves that, despite not being entirely profitable at present due to current kWh prices, implanting a battery will definitely be an option to consider in the future when these prices come down.

Keywords: swarm optimization; battery sizing; power smoothing; battery management system

1. Introduction

In this work, the authors propose an electrical battery model. Furthermore, the authors have modelled the electrical consumption and the consequent bills with reference to the different penalties that apply when the maximum power level is exceeded. This battery model is cognizant of the state of health losses via a charge and discharge policy for managing the reduction in maximum power. Therefore, the authors introduce a power smoothing technique. Despite the fact that there is other research work related to energy storage policies, all of them are developed from the electric energy producer's point of view [1–3]. Generally, these works do not study the problem considering multiple distributed little electrical energy consumption points.

The authors propose a model that is optimized by employing a particle swarm optimization algorithm. Sandhu et al. [4] employed this type of optimizer in power smoothing applications for sizing the battery energy storage system according to the level of smoothing power requirement. The optimization attempts to reduce electrical bills by making maximum power consumption cuts using an electrical battery as a power smoother. There are similar optimization problems that are solved by conventional algorithms [5], but they prove that intelligent algorithms must be utilized in order to apply real-world non-linear restrictions. The most important conventional optimization is the gradient descent-based optimization. This algorithm is widely applied in artificial neural network training [6,7], and it has different versions, such as stochastic descent gradient [8,9] or batch gradient descent [10]. This algorithm appropriately solves the neural network training problem as the loss functions are mathematically known, continuous and derivable. Usually, the gradient descent algorithm does not manage restrictions. However, restrictions are introduced when modifying the loss function with regularization techniques, such as L1 and L2, or batch

normalization techniques [11–13]. These algorithms are able to manage very extensive dimension optimization problems. Nevertheless, they need many iterations or epochs of optimization in order to reach an admissible solution. In fact, these algorithms are usually executed in graphical process units as they need plenty of computational resources. Often, these algorithms reach, yet do not leave, local minima points. Their convergence decreases heavily when they reach certain local optima, although momentum techniques are very common [14]. In this gradient descent algorithm, explicit mathematical expressions are needed in order to obtain the gradient components. In addition, these expressions must be finite as they are used to update the weights of the neural network. There are other conventional optimization algorithms, such as the Nelder–Mead simplex method [15]. This algorithm has heuristics that can manage discontinuities in cost or loss function. Moreover, it is able to consider inequality restrictions in loss or cost function and, in addition, it does not need any explicit mathematical expressions of cost or loss function regarding the variable set. It also has a very good optimization performance if the initial variable set values are close to the global solutions. Furthermore, it has some capability to avoid local minima, but it is not easy for it to avoid all local minima.

In order to solve the optimization problems, a particle swarm optimization (PSO) algorithm has been developed. This algorithm belongs to the family of heuristic algorithms, which are commonly used to solve complex multi-objective and non-linear problems. A very good introduction to, and application of, an example of particle swarm optimization can be found in [16]. Furthermore, there are other intelligent algorithms that are able to solve these problems, such as differential evolution [17]. However, the authors have chosen PSO because the changes in particles or solutions are smoother compared with that in differential evolution or genetic algorithms. In [18], a PSO-based optimization was employed to solve a problem regarding material dynamics behaviour identification. This work shows that this kind of algorithm can solve complex optimization problems.

At first, the PSO algorithm was applied in order to simulate the flocking process of birds. With time, the algorithm was discovered to be highly successful as an optimizer [19]. This algorithm has solved many optimization problems, as shown in [20]. Unfortunately, PSO does not guarantee the achievement of the absolute optimal solution. Nevertheless, this algorithm always improves at each iteration. PSO is similar to a genetic algorithm (GA), where the system is initialized with a population of random solutions. For each solution, a randomized velocity is assigned, and the original solutions, which are called particles, are then flown through the problem space. Each particle keeps track of its coordinates in the problem space, which are associated with the best solution (fitness) it has achieved so far [19]. The authors have applied PSO successfully in other optimization problems related to energy applications, such as [21–23].

The core of the current study consists of evaluating the installation of an electrical battery in the local council of Arrasate-Mondragon in order to reduce the peak power. The significance of this work resides in the expensive terms of the energy bill, which are due to the excessive maximum power levels assigned in electrical contracts and aggravated by certain maximum power levels. A detailed description of the European electrical energy market can be found in [24]. Thus, a particle swarm optimization (PSO)-based algorithm is used for battery sizing in peak power smoothing applications for different electrical consumption points. Therefore, the original process for the implementation of particle swarm optimization consists of changing the following steps of Eberhart et al. [19]. First, the population of particles with random positions and velocities is initialized. Next, the desired optimization fitness function is evaluated. The particle's fitness evaluation is then compared with the best solution achieved so far (*pbest*). If the current value is better than the current best, then *pbest* is set equal to the current value. The current value's location is also stored as *pbest's* location in the space. The fitness evaluation is compared with the population's previous best (*gbest*). If the current value is better than the current best, then *gbest* is reset to the current particle's array index and value. Subsequently, the velocity and position of the particle is changed. Finally, loop to the second step until a criterion is met.

2. Electrical Bill Model

The authors chose the 3.0A tariff since it is the most common one for 15 kW electrical links. In fact, 3.0A is the tariff that Arrasate-Mondragon's town council employs as a studied electrical consumption point [25]. This tariff is very common for low-voltage electrical consumption points with 15-kW contracted power or more, and it has three time spans. The time span hours of the day are specified in the Spanish electrical system. Therefore, each hour is assigned to one of these three time spans [25]. A similar work was presented by Martinez-Rico et al. [26], but from the point of view of the electrical energy producer. These time spans have their own energy price and maximum power price. In this tariff, the maximum contracted power at least in one time span grants 15 kW or more, see Equation (1):

$$\aleph_i(PcHired_i > 15 \ \wedge 1 \le i \le 3) \ge 1 \tag{1}$$

The 3.0A tariff has several penalties or bonuses depending on the maximum power reached at each time span. The Spanish electrical system defines all penalties and bonuses according to the ratio between the maximum power consumption reached at a given time span (at least for 15 consecutive minutes), known as Pg_i, and the maximum power contracted at that time span, $PcHired_i$. This ratio is defined as "c":

$$c = \frac{\max(Pg_i)}{PcHired_i} \tag{2}$$

where Pg_i is the power consumption at the i-th instant, while $PcHired_i$ is the maximum power contracted. If the power consumption at the i-th instant Pg_i overcomes $PcHired_i$ for more than 15 consecutive minutes, the electrical consumer must pay this maximum Pg_i power over the whole month for that time span. Note that all days are divided into three time spans: peak, valley, and medium consumption time spans. Each time span has its own $PcHired_i$. All the days in a year and all the hours in a day are classified considering these three time spans by the Spanish Government.

Based on this ratio, three different cases are defined. The first one occurs when $c < 0.85$, and the power is billed as stated:

$$PcBill_i = 0.85 \cdot PcHired_i \tag{3}$$

Second, in the case where $0.85 \le c \le 1.05$ is fulfilled, the maximum power consumed is considered as the hired power:

$$PcBill_i = \max(Pg_i) \tag{4}$$

Finally, if $1.05 < c$, the following equation is applied to measure the bill:

$$PcBill_i = 3 \cdot \max(Pg_i) - 2.1 \cdot PcHired_i \tag{5}$$

Due to the variety of penalizations and the number of periods from the tariff, a brute force optimization is proposed, as it only requires an adjustment of the hired power for each maximum power consumption period. In the case where no maximum power is higher than 15 kW, three iterations would be enough to obtain the optimum power to hire. Equations (6) and (7) illustrate the correct method to resolve this optimization issue:

$$1 \le NIt \le 3 \tag{6}$$

$$PcHired_i = \begin{cases} 15.01, & NIt = i \ \wedge \max(Pg_i) \le 15 \\ \max(Pg_i), & NIt \ne i \ \vee \max(Pg_i) > 15 \end{cases} \tag{7}$$

where NIt is the number of iterations accomplished by the proposed optimization algorithm.

To sum up the optimizing modelling for the 3.0A tariff, the function to minimize (Z) represents the cost of the hired powers:

$$Z = \sum_{1}^{3}(PcHired_i \cdot PriceP_i) \cdot DtBill \tag{8}$$

where $PriceP_i$ is the price of the hired power for the i period, which is the number of billed days during the consumption periods considered for the experiments performed in this study.

3. Battery Modelling

3.1. Charge and Discharge Dynamic Modelling

The depth of discharge (DoD) has been taken into account as the aging factor for each cycle of the battery to model the battery. Thus, the authors take into account the number of cycles of each DoD. However, in order to simplify the model, no other aging factors have been considered, such as temperature. The first restriction of the model is related to the quantity of energy stored in the battery at the t instant:

$$Eb(t) = Eb(t-1) - Pb(t) \cdot \Delta t \tag{9}$$

where $Pb(t)$ is the demanded power at the instant t, while Δt is the sample period of the power consumption data from the consumption point. The energy stored in the battery is limited by its nominal capacity (Eb_{max}). This way, the second restriction is determined:

$$0 \le Eb(t) \le Eb_{max} \tag{10}$$

Referring to the demanded power at each time, the next restriction is imposed to define the minimum and maximum possible values:

$$-Pb_{Nom} \le Pb(t) \le Pb_{Nom} \tag{11}$$

The battery will be discharged once $Pb(t)$ is positive. On the other hand, the battery will be charged for the negative values of $Pb(t)$. Furthermore, the $Pb(t)$ value is defined by a control function, stated below:

$$Pb(t) = F\left(Pb_{Nom}, Eb(t-1), Pc(t), \overline{Pc_{Med}}, \alpha\right) \tag{12}$$

where Pb_{Nom} is the nominal power of the battery and $Eg(t-1)$ is the stored energy at each previous instant. $Pc(t)$ is the consumed power by the consumption point at t instant, $\overline{Pc_{Med}}$ is the average consumed power by the consumption point, and α is an aggressiveness coefficient, the value of which goes from 0 to 1. The control function is composed of the three Equations (13)–(15)

$$Pb(t) = \alpha \cdot \left(Pc(t) - \overline{Pc_{Med}}\right), \quad -Pb_{Nom} < \alpha \cdot \left(Pc(t) - \overline{Pc_{Med}}\right) < Pb_{Nom} \wedge Eb(t-1) \neq 0 \tag{13}$$

$$Pb(t) = Pb_{Nom}, \quad Pb_{Nom} \le \alpha \cdot \left(Pc(t) - \overline{Pc_{Med}}\right) \wedge Eb(t-1) \neq 0 \tag{14}$$

$$Pb(t) = -Pb_{Nom}, \quad -Pb_{Nom} \ge \alpha \cdot \left(Pc(t) - \overline{Pc_{Med}}\right) \vee Eb(t-1) = 0 \tag{15}$$

The last case is also considered when the battery is discharged. This is to avoid having the battery completely discharged for a long time, which would make its use impossible when the cases in Equations (13) and (14) take place, due to the continuous execution of those two cases without the case in Equation (15) occurring.

As this is a simulation-headed model, additional restrictions are considered when programming. These restrictions are not directly related to the control function of the battery. However, they are applied to the function, and then evaluated, and their goal consists of

maintaining coherence between simulation and reality. On one hand, the energy stored in the battery must be positive or equal to zero. This is further explained in Equation (16):

$$Pb(t) \cdot \Delta t \leq Eb(t-1) \tag{16}$$

This restriction is derived from Equation (10), and it is always accomplished when the battery is charging. However, it is also necessary to ensure its fulfilment when in the process of discharging. This is achieved by limiting the demanded power to the battery. The proposed method is as shown below:

$$Pb(t) = \frac{Eb(t-1)}{\Delta t}, \quad Pb(t) \cdot \Delta t > Eb(t-1) \tag{17}$$

The case in Equation (17) will always eventuate once the cases in Equations (13) and (14) occur, without the case in Equation (16) taking place.

On the other hand, it is mandatory to ensure that the stored energy in the battery never surpasses its nominal capacity. As in the previous restriction in Equation (17), a new one is derived from Equation (10), and it must be satisfied in every charging process of the battery:

$$- Pb(t) \cdot \Delta t \leq Eb_{max} - Eb(t-1) \tag{18}$$

Once again, it is necessary to limit the power supplied to the battery, with the following method being proposed for this purpose:

$$- Pb(t) = \frac{Eb_{max} - Eb(t-1)}{\Delta t}, \quad -Pb(t) \cdot \Delta t > Eb_{max} - Eb(t-1) \tag{19}$$

This case needs Equation (15) to happen, while the restriction in Equation (10) does not.

3.2. Modelling Battery Aging

The battery's aging modelling has been performed with the intention of taking purchase amortization into account in the cost function. Therefore, only the *DoD* and the number of cycles for every *DoD* level from the battery are considered. The battery operating temperature changes are not relevant in the current study, as the authors assume that the energy storage systems are going to be in a temperature-controlled environment. A cycle of a battery is defined as the accumulated discharge equal to 100% of the nominal capacity of the battery. That is to say, a complete cycle can be discharging the battery until 80% of its capacity, and then charging to 100% five times, the same way discharging the battery completely and charging it again to 100% equals a cycle. The depth of discharge (*DoD*) of a cycle *i* can be defined as the difference between the maximum and minimum charges during the cycle, divided by the nominal capacity of the battery [27]:

$$DoD_i = \frac{\max(Eb(t)_i) - \min(Eb(t)_i)}{Eb_{max}} \tag{20}$$

where $Eb(t)_i$ is the history of the battery's charge during the cycle *i*. This is the method proposed for the estimation of *DoD*. It is likely that, in other similar studies, slightly different definitions of *DoD* and cycle are given. This is because they are both often discussed concepts, being dependent on the battery manufacturer and type, so there is no common definition.

The number of cycles a battery can carry out is proportionally inverse to the *DoD* endured in its cycle. This relationship comes from an exponential function that must be provided by the maker of the battery. Figure 1 shows this exponential function for a concrete battery model.
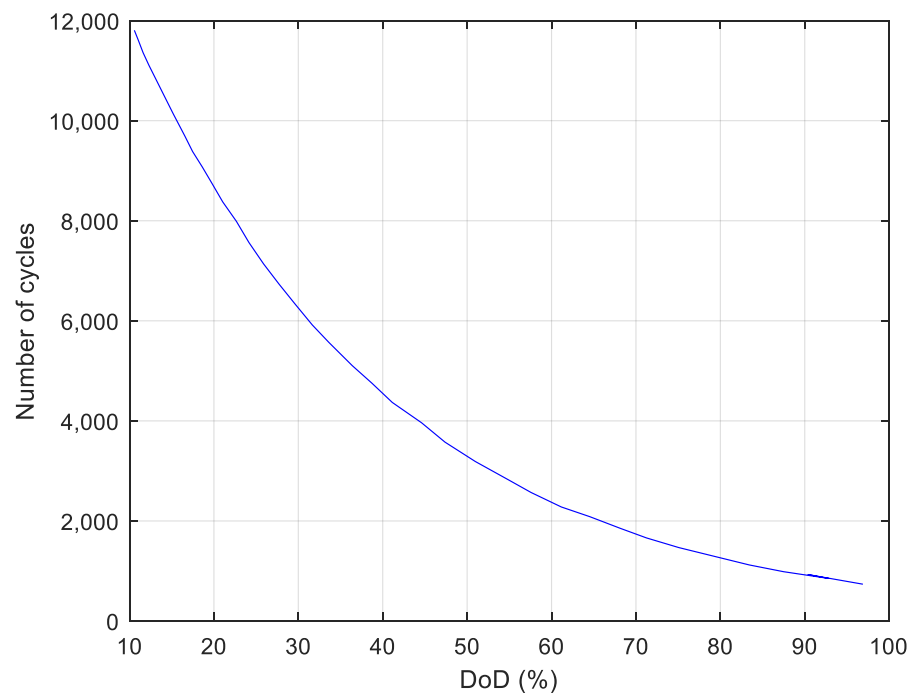
Figure 1. Example of the relationship between the cycles and depth of discharge (*DoD*) of a commercial battery [28].

As the battery loss of life on each cycle *i* depends on the endured *DoD* (DoD_i) on that cycle, the next method is applied to calculate the loss of life (LL_i) in each cycle:

$$LL_i = \frac{1}{F(DoD_i)} \tag{21}$$

where *F* is the exponential function that relates *DoD* with the number of cycles afforded. The total life loss (*LL*) of the battery is estimated via Equation (22):

$$LL = \sum_{i=1}^{nCycles} LL_i \tag{22}$$

where *nCycles* represents the number of cycles performed. The *LL* value starts on 0, and increases to a maximum of 1, which would mean the life span of the battery has reached an end, and, hence, the battery will no longer be used.

By obtaining the life loss of the battery, its life span, measured as the quantity of years it might last, is estimated. To do this, Equation (23) is proposed:

$$Lifetime = \frac{HoU}{LL} \cdot \frac{1\ Year}{8760\ hours} \tag{23}$$

where *HoU* is the number of hours since the battery was bought until the exact moment this expression is evaluated, while *Lifetime* represents the number of years the battery is estimated to last. This expression is chosen as it is possible that a year has not yet passed when the expression is evaluated. The value of *LL* is the one obtained by evaluating the cycles during the *HoU*. This way, the annual amortization of the battery is calculated via the following equation:

$$Annual\ amortization = \frac{PriceBattery}{Lifetime} \tag{24}$$

For the calculation of the battery's price, Equation (25) is applied, as it is useful to estimate the price of the kWh of the battery in day d, which represents the day the battery was acquired, having as a reference 1 January 2018 (see Martinez-Rico et al. [29]):

$$Price_d = 162.3e^{-0.1029d/365} \tag{25}$$

Therefore, the price of the battery is defined by Equation (26):

$$PriceBattery = Eb_{max} \cdot Price_d \tag{26}$$

3.3. Modelling Electric Grid Demand

Due to the use of the battery, the demanded power from the electric grid is no longer the same as that demanded by the consumption point. Thus, two new restrictions emerge. The first is shown below:

$$Pg(t) = \begin{cases} Pc(t) - Pb(t), & Pb(t) > 0 \\ Pc(t), & Pb(t) \leq 0 \end{cases} \tag{27}$$

where $Pg(t)$ is the demanded power from the grid at instant t. This value will be different to $Pc(t)$ when the battery is discharged. The second restriction is as follows:

$$0 \leq Pg(t) \tag{28}$$

In order to fulfil this restriction, the power supplied by the battery must be limited in the case of discharge. To accomplish this, the following method is proposed:

$$Pb(t) = Pc(t), \quad Pb(t) > Pc(t) \tag{29}$$

To sum up this part, the energy consumed is defined by the next equation:

$$Eg(t) = Pg(t) \cdot \Delta t \tag{30}$$

where $Eg(t)$ is the consumed energy from the grid at instant t.

3.4. Cost Function

The cost function proposed for the optimization problem is acceptable for a multi-objective optimization as it has three different goals: maximizing the battery life in order to minimize the annual amortization; minimizing the cost of the hired power; and minimizing the cost of the energy demanded from the grid.

With Z being the cost of the hired power for the consumption point, and $PriceE_i$ being the price of the energy for period i, the cost function is as follows:

$$\min \sum_{i=1}^{3} (Eg(t)_i \cdot PriceE_i) + \frac{Eb_{max} \cdot Price_d}{Lifetime} + Z \tag{31}$$

4. Optimization Process

In the optimization process, the variables to change are the battery capacity Eb_{max}, the nominal power of the battery Pb_{Nom}, and the power smoothing coefficient of the power smooth policy, α.

The authors propose a particle swarm optimization (PSO) algorithm as optimizer. This algorithm was proposed by Kennedy and Eberhart [30] and is based on swarm intelligence. This algorithm tries to optimize the cost function using a particle set. Each particle has a solution for the optimization problem, and there is a vector that contains three values: battery capacity, nominal power, and power smoothing coefficient.

The algorithm has a set of particles, and each particle has a proposed solution for the optimization problem. This solution is the generalized position of the particle x_i. Each

particle has a position given by its solution. Furthermore, each particle has a speed vector, v_i, which is calculated in order to reach better solutions or positions.

During optimization iterations, each particle evaluates its position or solution. This evaluation is made with the cost function defined previously. Each *i*-th particle has its positions and speed vector, and the best position reached by this particle is $x_{i,best}$. All particles know the best position, x_{best}, reached by the whole set of particles until that iteration. The particle's position is defined via the following equation at the *t*-th iteration:

$$x_i(t+1) = x_i(t) + \Delta t \cdot t \cdot v_i(t+1) \tag{32}$$

where v_i is the speed at each t iteration. The speed of the *i*-th particle changes to $x_{i,best}$ and to the best position x_{best} reached by whole set. There are three parameters that modulate the behaviour of the PSO algorithm: the weight of the *i*-th particle w_i, $\varphi_{1,max}$, and $\varphi_{2,max}$. The last two parameters modulate the exploration or exploitation behaviours of the algorithm. $\varphi_{1,max}$ is the maximum value that can achieve φ_1, which is a positive uniform random number at each iteration and particle. $\varphi_{2,max}$ is also the maximum value that can achieve φ_2, which is a positive uniform random number at each iteration and particle:

$$v_i(t+1) = w \cdot v_i(t) + \varphi_1 \cdot (x_{i,best} - x_i(t)) + \varphi_1 \cdot (x_{best} - x_i(t)) \tag{33}$$

The weight of each particle has been examined in many studies [31–34] as it can heavily affect the PSO optimizer's solutions. In this case, the authors have proposed a time variable weight policy, as suggested in [32]. The weights start at a high value and decrease time linearly until a minimum value is reached. This time variable policy is described in Equation (34):

$$w(t) = \frac{it_{max} - t}{it_{max}} (w_{max} - w_{min}) + w_{min} \tag{34}$$

where it_{max} is the number of iterations needed to change the weight from w_{max} to w_{min}. In fact, the weight decreases when the number of optimization algorithm iterations increases (see Figure 2).
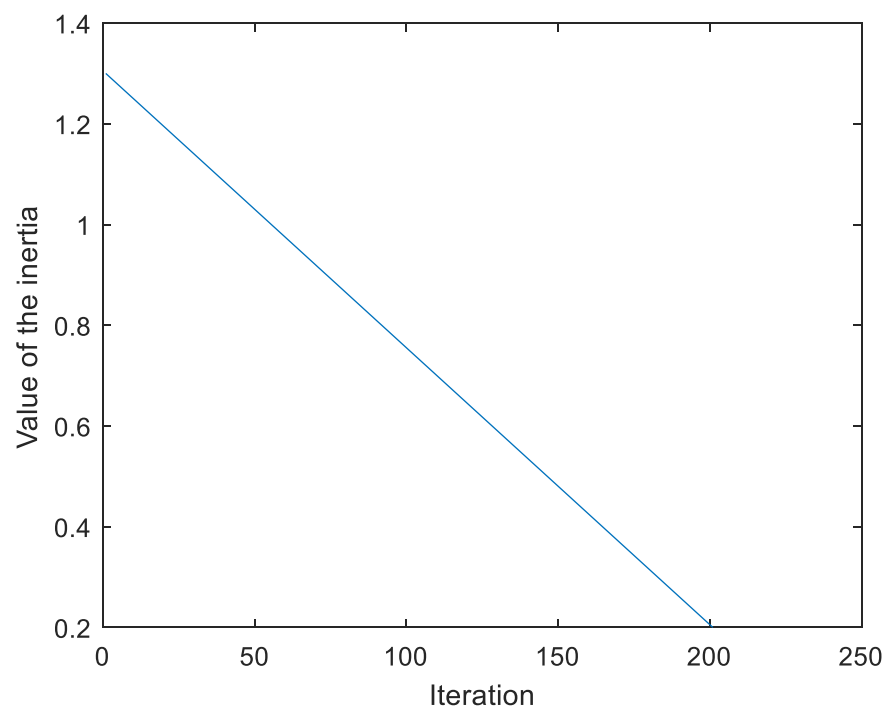

Figure 2. Weight changes through time (the inertia is a dimensionless parameter).

5. Study Cases and Consumption Data

5.1. Consumption Data

The authors collected electrical consumption data from the northern Spanish town of Arrasate-Mondragon. This town council is in the Basque Country and has about 22,000 inhabitants. The town council of Arrasate has more than 200 electrical consumption points. Some of them are public illumination systems, while others are different public services such as sports centres. The consumptions that are taken into account in this work are real power consumption values measured in real time. The authors proposed the sports centre of Musakola, Arrasate as being the most important power consumption point to study. The authors collected 4922 power consumption samples between 20 May 2020 and 11 December 2020, with a sample time of one hour.

A web application was created by the authors that was based on web scraping, in order to collect all the data from the electrical company (Iberdrola) automatically. The authors created this application using C# programming language, as it was very easy to develop a frontend for the town council's technical personnel. The electrical power consumptions are shown in Figure 3.

Figure 3. Power consumption (kW) per time (hours) at the sports centre in Musakola, Arrasate.

This power consumption point has a three-period electrical bill system [25]. A three-period bill system specifies three different prices during the day. Generally, the first period is related to the low general consumption hours of the day, the second one is related to the medium general consumption hours of the day, and, finally, the third period is related to the high general consumption hours of the day. Therefore, each period has its own maximum power. Usually, these powers are in ascending order. In this case, the town council has the same maximum power levels for each period (see Table 1).

Table 1. Maximum power consumption initially contracted with the electrical company.

	First Period	Second Period	Third Period
Maximum power levels contracted (kW)	150	150	150

The battery health loss has been modelled as per [29]. According to the depth of discharge (*DoD*), the authors applied a health loss function, which is defined in Table 2. This table is defined by an Arrhenius curve as per [29] (see Figure 1).

Table 2. *DoD* vs. maximum number of cycles at each *DoD*.

DoD Range (%)	Admitted Cycles
10	15,000
20	8750
30	5500
40	3800
50	3000
60	2250
70	1750
80	1500

The authors made an exponential regression of this data in order to calculate the health loss per cycle of discharge for a given *DoD*, following Equation (35):

$$15435.1216 \cdot e^{-0.0303 \cdot x} \tag{35}$$

where x is the *DoD* of the discharge cycle of the battery. The authors did not take into account aging factors such as temperature, so as to keep the battery model simple.

The maximum power level prices per kW and the energy prices in euros per kWh of power consumption are shown in Table 3 for 1 February 2020. The prices of the three-period bill system are also displayed (the 3.0A tariff, to be precise, see [25]).

Table 3. Prices of maximum power levels and energy prices.

	Energy (EUR/kWh)	Maximum Power Level (EUR/kW per day)
1st period	0.1131	0.1170
2nd period	0.0938	0.0702
3rd period	0.0649	0.0468

The exact time spans of each day for each period and the maximum power levels for the three-period bill system (the 3.0A tariff bill system) are defined in the Spanish electrical market law.

5.2. Electrical Battery Economical Return Analysis

Next, the authors reveal the economic returns caused by implanting an electrical battery in order to smooth the power demand curve. The analysis was carried out using prices and costs from 20 May 2020. At that time, the relative cost of an energy unit's capacity was EUR 126.9976 per kWh [29]. The authors interpolated the cost using the prediction equation in [29].
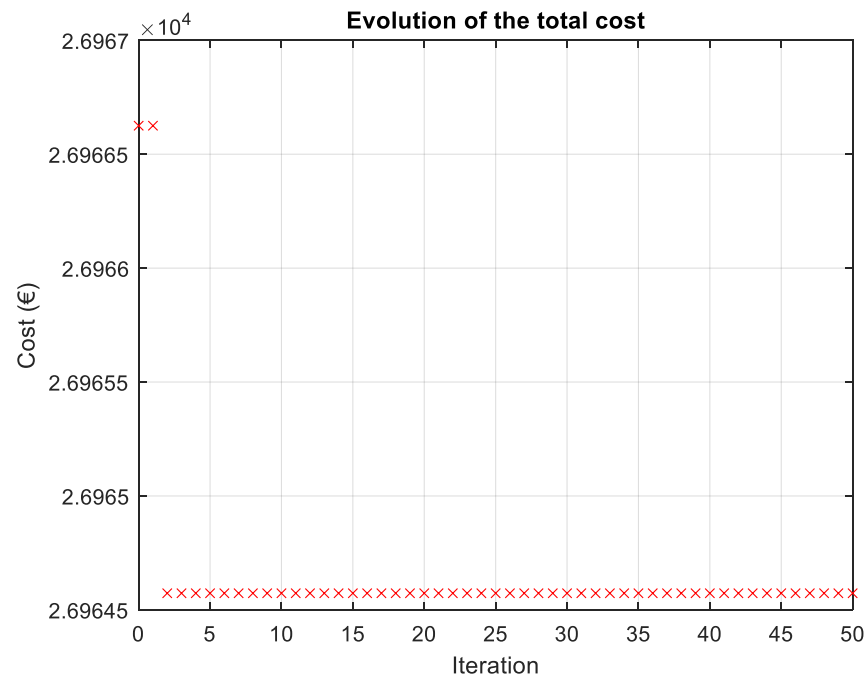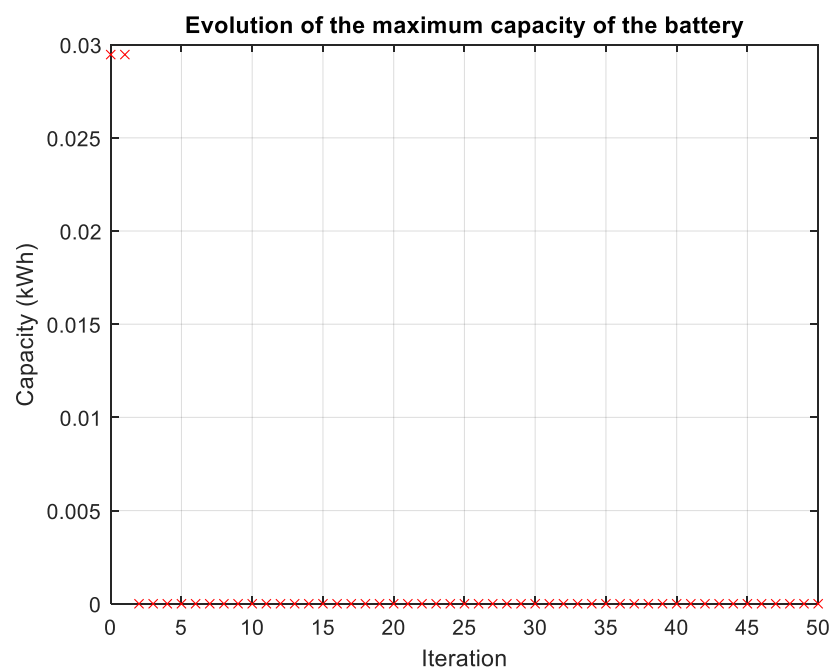
The parameter set applied to the PSO optimization algorithm is shown in Table 4.

The particle position initialization follows a uniform random distribution.

The authors executed the same optimization many times in order to verify that the obtained optimal solution was really the best solution with different initialization points. The same optimal solution was always achieved. This solution is shown in Figures 4 and 5.

Table 4. Particle swarm optimization (PSO) optimizer parameter set for the first study case.

Parameter	Value
Particle number	40
Iteration number	50
φ_1	0.5
φ_2	0.5
dt	1
Initial inertia	1.2
Final inertia	0.3

Figure 4. Cost evolution (€ = EUR) along PSO optimizer iterations.

Figure 5. Obtained optimal capacity (kWh) along the PSO optimizer iterations.

It is clear that, nowadays, it is not worth applying battery systems in order to smooth power peaks. However, the authors carried out a sensitivity analysis of the solutions changing the price of the battery capacity.

5.3. Sensitivity Analysis of Battery Capacity Price (EUR/kWh)

In this section, the authors pose the following question: Which battery capacity price is worth a power curve smoothing battery?

In order to answer this question, the authors defined a set of optimization problems. Each optimization problem has its own capacity price. A PSO optimizer was applied to each optimization problem. The capacity price was uniformly increased for each case. Each PSO optimization applies a capacity price as per Equation (36). Therefore, the authors achieve the optimal battery capacity for a given capacity price:

$$PricekWh = maxPrice - nIt\frac{maxPrice - minPrice}{numOfIterations} \tag{36}$$

where *numOfIterations* is the number of optimization problems, *nIt* is the optimization problem index, and *maxPrice* and *minPrice* are the maximum and minimum capacity prices, respectively.

The particles are initialized around the best solutions achieved in the previous optimization. The initialization process at each optimization problem generates a particle with a better solution than before. If an initialized particle does not achieve a better solution, then the best solutions reached in the previous optimization problem are taken.

In order to ensure the coherence of the algorithm in the first iteration, a particle with a non-optimal fitness is introduced. This way, it is guaranteed that the optimal cost obtained during the brute force algorithm's iterations is decreasing. Figure 6 illustrates the flux diagram of the process.

Table 5 shows the parameters used for the algorithm. The number of particles is reduced because, as it is executed several times, it ends up being computationally expensive, and it would otherwise take a lot of time to conclude the sensitivity study.

Table 5. Sensitivity algorithm parameters.

Parameter	Value
Brute force iterations	99
Battery maximum kWh cost	EUR 126.9974
kWh price change per iteration	EUR–1.1700
Battery minimum kWh cost	EUR 10
Number of particles	40
Number of iterations	50
φ_1	0.5
φ_2	0.5
dit	1
Initial inertia	1.2
Final inertia	0.3

Figure 7 displays the results obtained for the optimal costs of the problem's solution for each price of the battery.

According to Figure 7, the total optimal cost starts descending faster as of iteration 20. With regard to the power term, it stays stable for all iterations. The savings come in terms of energy. Figure 8 shows the evolution of the battery parameters along the iterations.
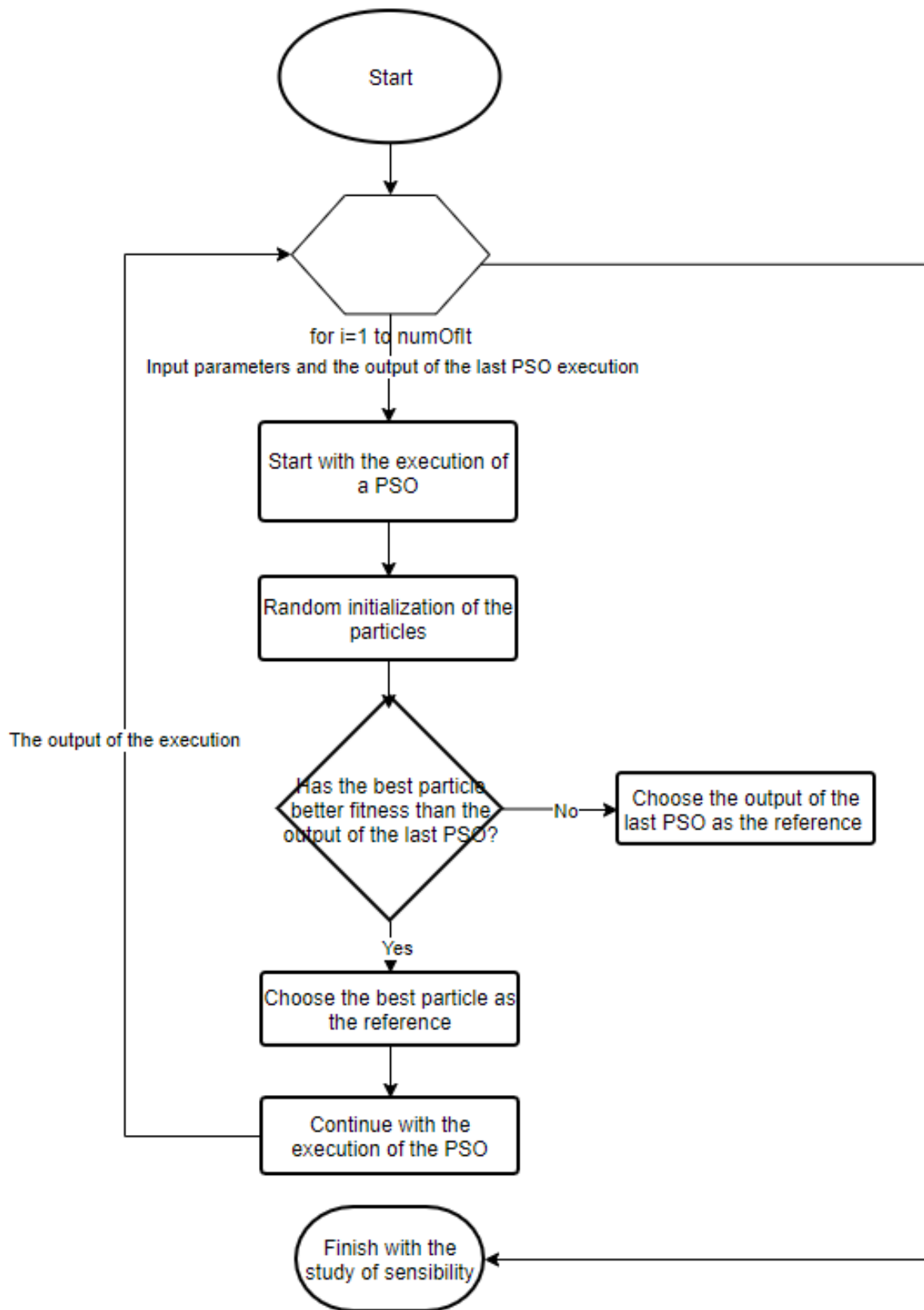
Start

for i=1 to numOfIt

Input parameters and the output of the last PSO execution

Start with the execution of a PSO

Random initialization of the particles

The output of the execution

Has the best particle better fitness than the output of the last PSO? — No — Choose the output of the last PSO as the reference

Yes

Choose the best particle as the reference

Continue with the execution of the PSO

Finish with the study of sensibility

Figure 6. Flux diagram of the developed PSO algorithm.

Figure 7. Optimal cost (€ = EUR) for each PSO algorithm's execution during the sensitivity algorithm.

Figure 8. Optimal battery capacities (kWh) and nominal power (kW) along the iterations of the sensitivity algorithm.

For this study, in which the sample periods are an hour, a nominal power value higher than the battery's capacity means that the optimal nominal power is equal to the battery's capacity. The parameter α kept a constant value of 1 during all the iterations of the sensitivity algorithm.

Regarding the power consumption of the electrical grid, Figures 9–12 show a comparison with and without the battery of iterations 30, 50, 80, and 100, respectively, after implanting a battery.
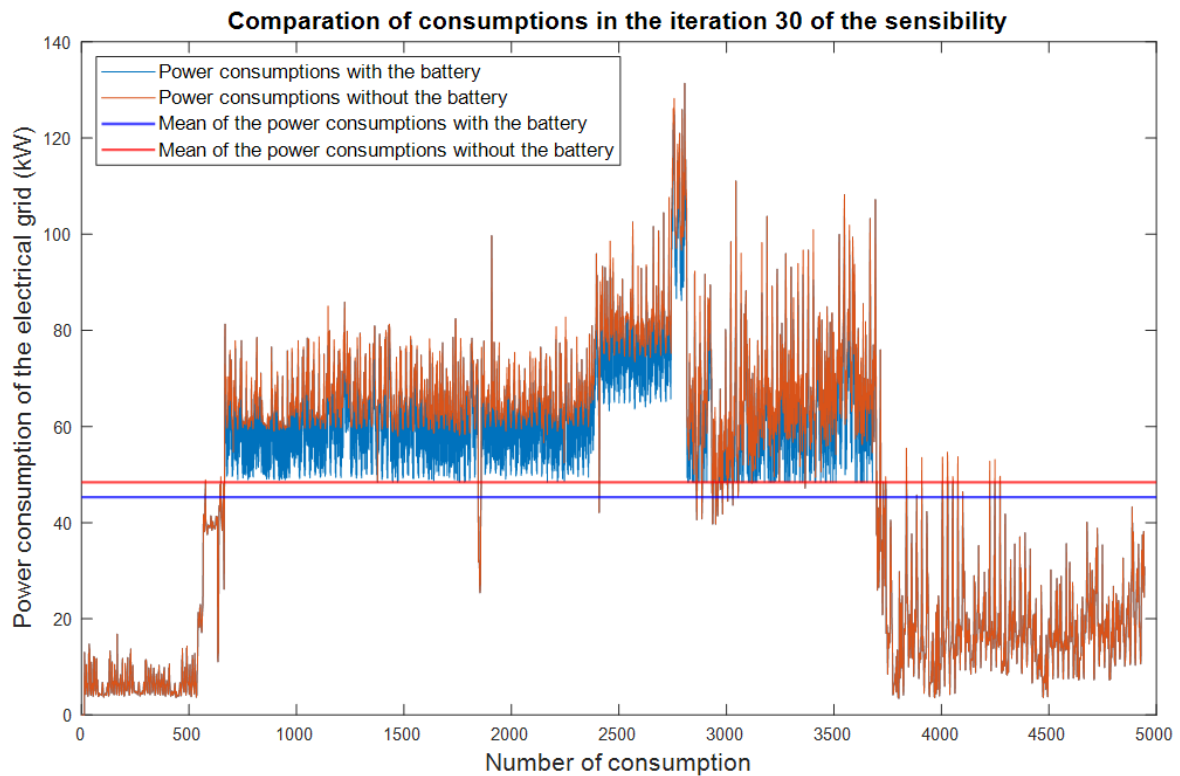

Figure 9. Power consumption (kW) comparison with and without the battery (iteration 30).
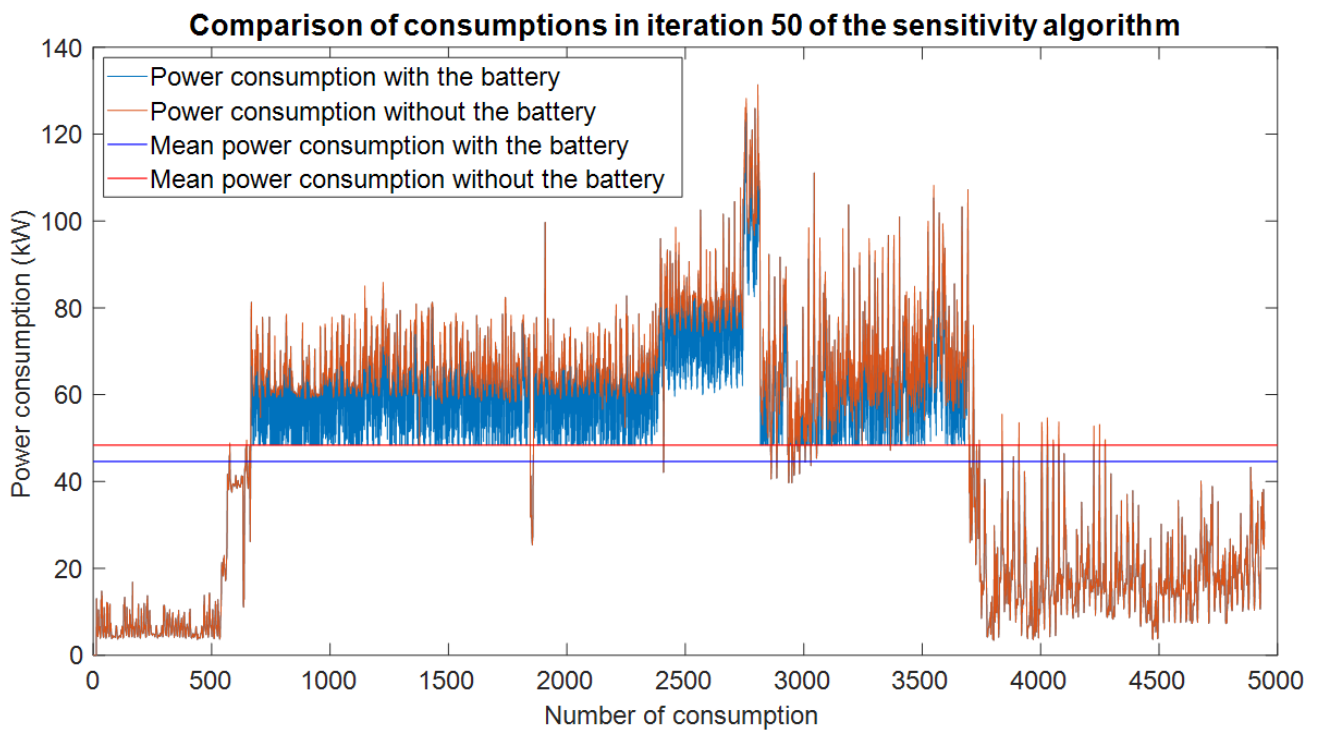

Figure 10. Power consumption (kW) comparison with and without the battery (iteration 50).
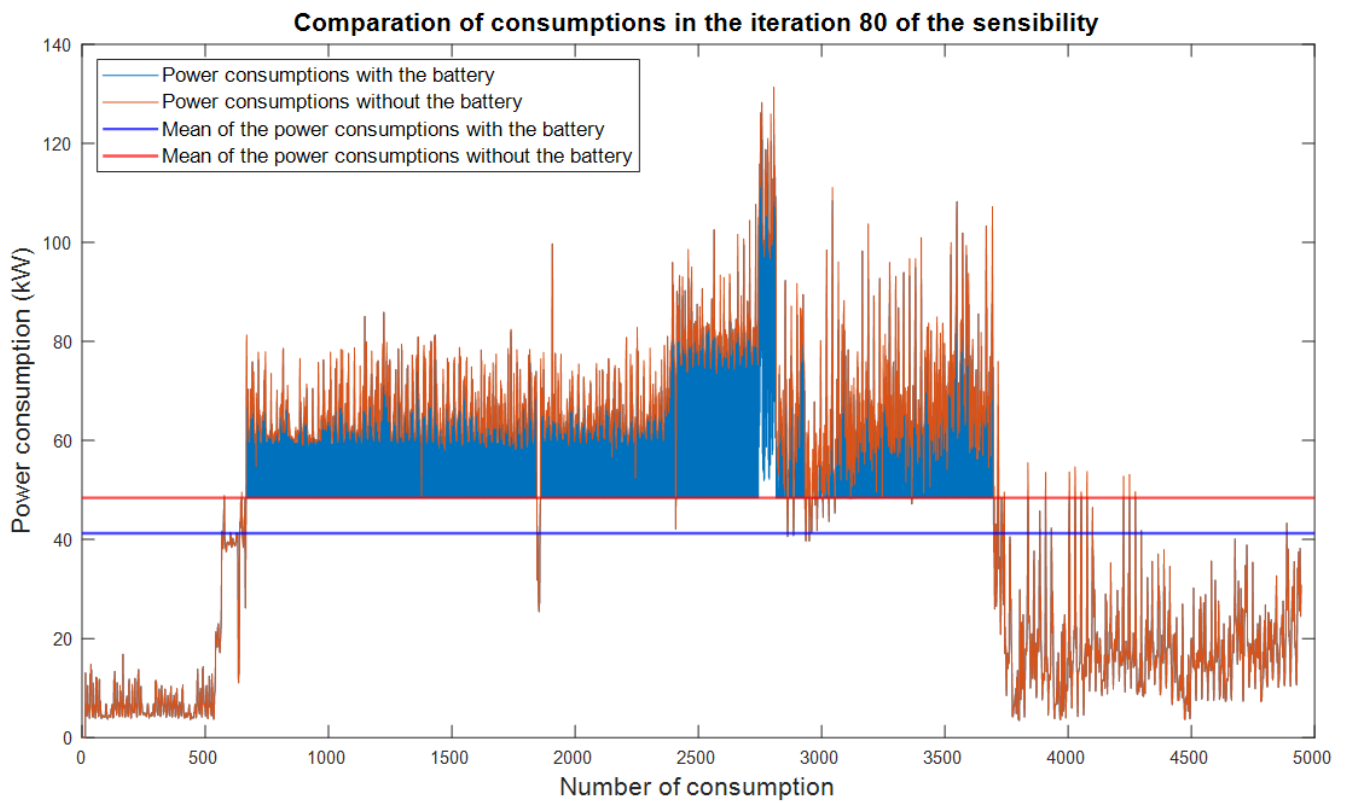
Figure 11. Power consumption (kW) comparison with and without the battery (iteration 80).
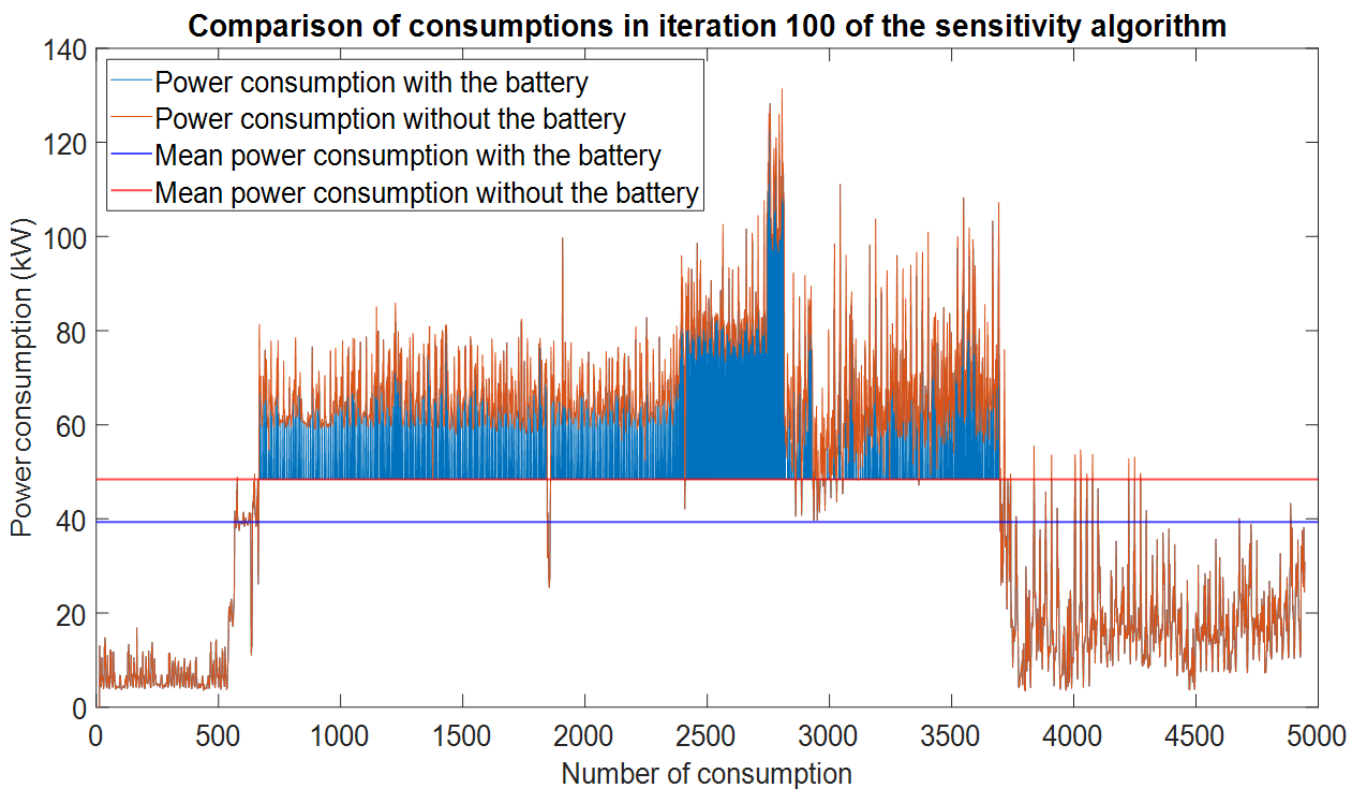

Figure 12. Power consumption (kW) comparison with and without the battery (iteration 100).

According to the previous figures, the bigger the battery, the better the control function works. In addition, Figure 13 illustrates how all the optimal batteries are submitted to very deep cycles, increasing the cycle's depth as the battery's capacity also increases.
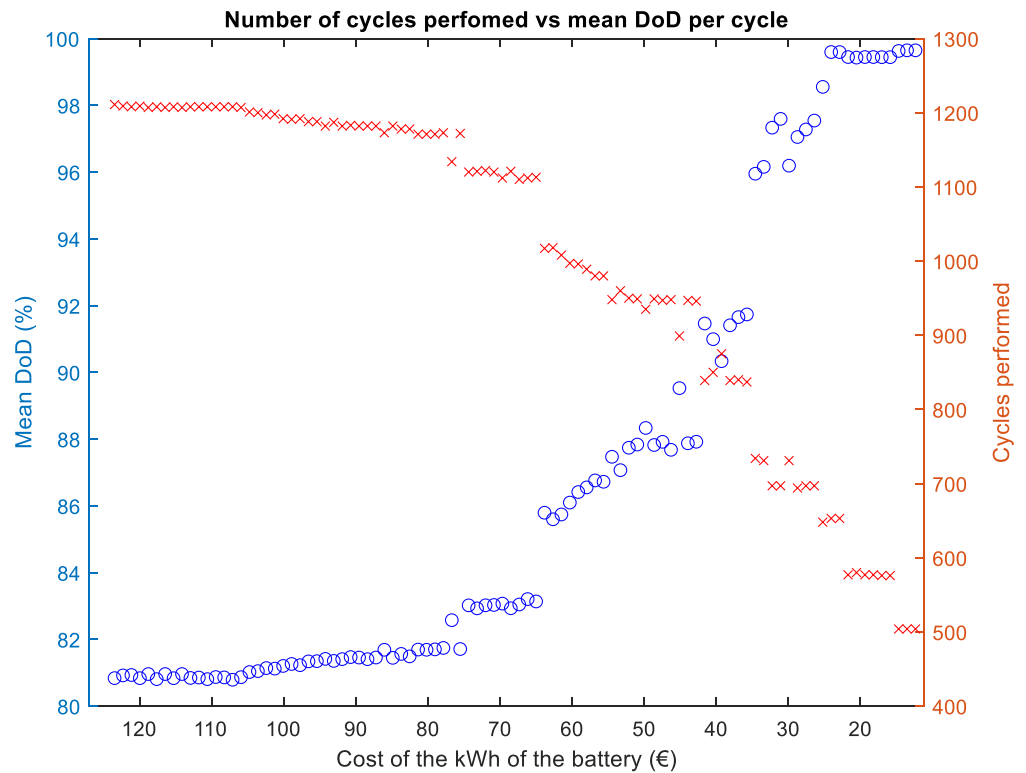

Figure 13. Performed cycles by battery, and average endured *DoD* along the performed cycles.

Another interesting relationship is between the average minimum and maximum charges along the cycles and the battery's capacity. From this, it can be observed that, in batteries with a *DoD* distinct from 100%, the complete charge of the battery has prevailed over the complete discharge, or vice versa. Figure 14 represents the average maximum and minimum charge along the cycles related to the nominal capacity.
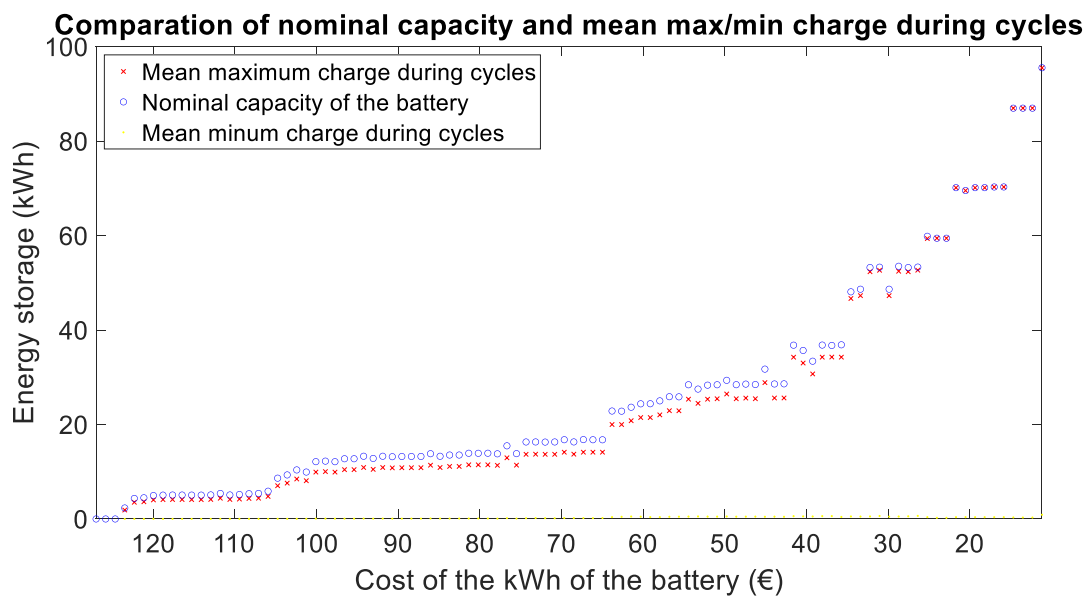

Figure 14. Average maximum and minimum charge (kWh) along the cycles related to the nominal capacity.

6. Conclusions

Performing a profitability analysis after implanting a battery at a consumption point has shown that, at the present time, it is still not profitable.

Subsequently, the sensitivity of the price per kWh of the battery has been analysed to determine the prices at which the implantation of a battery could be profitable, and it has been observed that, despite it not being profitable at this time, it is likely to become so soon enough, with profitability increasing faster as the kWh price decreases. However, this profitability has come from the non-consumption of energy obtained from the electrical grid rather than from a lower demand for maximum power peaks.

On one hand, there are set characteristics for the data from the consumption point, where there are three different zones. The first zone involved the first 600 consumption data, where all the consumption values are inferior to the average consumption. The second zone comprised the next 3000 data, where practically all the consumptions are superior to the average. The third included the last 1500 data, where most consumptions are below the average. In the second zone, the higher consumptions are caused by the fact that the battery is continuously undergoing a discharging process, so the moment the maximum points are reached, there is no, or very little, energy stored. Therefore, it is impossible to reduce the maximum peaks.

On the other hand, the type of model employed is based on releasing energy indiscriminately when the demanded consumption is superior to the average.

In terms of the aging of the batteries, the sensitivity study concluded that the use of short lifespan batteries may be profitable in the future. Despite not being highly profitable, the mere fact that a result like this may become profitable means that the price of the kWh would lower until the indiscriminate use of these batteries becomes profitable. This demonstrates that, in 10 or 20 years, it will be a factor to keep in mind.

One interesting result that was obtained is the optimal nominal power that has been assigned to every optimal battery. Considering that the employed sample period was an hour, it was observed during most sensitivity iterations that the optimal nominal power was slightly lower than the optimal capacity of the battery. Furthermore, the average maximum charge per cycle was inferior to the maximum capacity of the battery. Therefore, the authors conclude that, for this particular study, the nominal power of the battery performed as a limiting mechanism to lower the depth of discharge of the battery.

Author Contributions: Conceptualization, A.Z.; methodology, A.Z. and D.A.I.-G.; software, E.Z. and A.Z.; validation, D.A.I.-G., E.Z., and U.F.-G.; formal analysis, D.A.I.-G. and A.Z.; investigation, J.G.-O.; resources, U.F.-G. and E.Z.; data curation, A.Z.; writing—original draft preparation, J.G.-O.; writing—review and editing, J.G.-O. and U.F.-G.; visualization, J.G.-O. and E.Z.; supervision, U.F.-G.; project administration, A.Z. and U.F.-G.; funding acquisition, A.Z. and U.F.-G. All authors have read and agreed to the published version of the manuscript.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

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

References

1. Saez-De-Ibarra, A.; Milo, A.; Gaztanaga, H.; Debusschere, V.; Bacha, S. Co-Optimization of Storage System Sizing and Control Strategy for Intelligent Photovoltaic Power Plants Market Integration. *IEEE Trans. Sustain. Energy* **2016**, *7*, 1749–1761. [CrossRef]
2. Gonzalez-Garrido, A.; Saez-De-Ibarra, A.; Gaztanaga, H.; Milo, A.; Eguia, P. Annual Optimized Bidding and Operation Strategy in Energy and Secondary Reserve Markets for Solar Plants with Storage Systems. *IEEE Trans. Power Syst.* **2018**, *34*, 5115–5124. [CrossRef]
3. Wang, J.; Zhong, H.; Tang, W.; Rajagopal, R.; Xia, Q.; Kang, C.; Wang, Y. Optimal bidding strategy for microgrids in joint energy and ancillary service markets considering flexible ramping products. *Appl. Energy* **2017**, *205*, 294–303. [CrossRef]
4. Sandhu, K.S.; Mahesh, A. A new approach of sizing battery energy storage system for smoothing the power fluctuations of a PV/wind hybrid system: A New Approach of Sizing Battery Energy Storage System. *Int. J. Energy Res.* **2016**, *40*, 1221–1234. [CrossRef]
5. Khishvand, M.; Khamehchi, E.; Nokandeh, N.R. A Nonlinear Programming Approach to Gas Lift Allocation Optimization. *Energy Sources Part A Recover. Util. Environ. Eff.* **2015**, *37*, 453–461. [CrossRef]
6. Andrychowicz, M.; Denil, M.; Gomez, S.; Hoffman, M.W.; Pfau, D.; Schaul, T.; Shillingford, B.; De Freitas, N. Learning to Learn by Gradient Descent by Gradient Descent. Advances in Neural Information Processing Systems. *arXiv* **2016**, arXiv:1606.04474.
7. Jian, J.; Yang, L.; Jiang, X.; Liu, P.; Liu, M. A Spectral Conjugate Gradient Method with Descent Property. *Mathematics* **2020**, *8*, 280. [CrossRef]
8. Amari, S.-I. Backpropagation and stochastic gradient descent method. *Neurocomputing* **1993**, *5*, 185–196. [CrossRef]
9. Zhou, J.; Wei, W.; Zhang, R.; Zheng, Z. Damped Newton Stochastic Gradient Descent Method for Neural Networks Training. *Mathematics* **2021**, *9*, 1533. [CrossRef]
10. Needell, D.; Ward, R. Batched Stochastic Gradient Descent with Weighted Sampling. In *Approximation Theory XV: San Antonio 2016*; Fasshauer, G.E., Schumaker, L.L., Eds.; Springer: Berlin/Heidelberg, Germany, 2017; Volume 201, pp. 279–306. [CrossRef]
11. Shah, P.; Khankhoje, U.; Moghaddam, M. Inverse Scattering Using a Joint Norm-Based Regularization. *IEEE Trans. Antennas Propag.* **2016**, *64*, 1373–1384. [CrossRef]
12. Souza, P.V.D.C.; Torres, L.C.B.; Silva, G.R.L.; Braga, A.D.P.; Lughofer, E. An Advanced Pruning Method in the Architecture of Extreme Learning Machines Using L1-Regularization and Bootstrapping. *Electronics* **2020**, *9*, 811. [CrossRef]
13. Wu, S.; Jiang, H.; Shen, H.; Yang, Z. Gene Selection in Cancer Classification Using Sparse Logistic Regression with L1/2 Regularization. *Appl. Sci.* **2018**, *8*, 1569. [CrossRef]
14. Mahboubi, S.; Ninomiya, H.; Asai, H. Momentum acceleration of quasi-Newton based optimization technique for neural network training. *Nonlinear Theory Its Appl. IEICE* **2021**, *12*, 554–574. [CrossRef]
15. Lagarias, J.C.; Reeds, J.A.; Wright, M.H.; Wright, P.E. Convergence Properties of the Nelder—Mead Simplex Method in Low Dimensions. *SIAM J. Optim.* **1998**, *9*, 112–147. [CrossRef]
16. Yoshida, H.; Kawata, K.; Fukuyama, Y.; Takayama, S.; Nakanishi, Y. A particle swarm optimization for reactive power and voltage control considering voltage security assessment. *IEEE Trans. Power Syst.* **2000**, *15*, 1232–1239. [CrossRef]
17. Centeno-Telleria, M.; Zulueta, E.; Fernandez-Gamiz, U.; Teso-Fz-Betoño, D.; Teso-Fz-Betoño, A. Differential Evolution Optimal Parameters Tuning with Artificial Neural Network. *Mathematics* **2021**, *9*, 427. [CrossRef]
18. Sánchez-Chica, A.; Zulueta, E.; Teso-Fz-Betoño, D.; Martínez-Filgueira, P.; Fernandez-Gamiz, U. ANN-Based Stop Criteria for a Genetic Algorithm Applied to Air Impingement Design. *Energies* **2019**, *13*, 16. [CrossRef]
19. Shi, Y. Particle Swarm Optimization: Developments, Applications and Resources. In Proceedings of the 2001 Congress on Evolutionary Computation (IEEE Cat. No.01TH8546), Seoul, Korea, 27–30 May 2001; Volume 1, pp. 81–86. [CrossRef]
20. Poli, R. Analysis of the Publications on the Applications of Particle Swarm Optimisation. *J. Artif. Evol. Appl.* **2008**, *2008*, 1–10. [CrossRef]
21. Martínez-Filgueira, P.; Zulueta, E.; Sánchez-Chica, A.; Fernández-Gámiz, U.; Soriano, J. Multi-Objective Particle Swarm Based Optimization of an Air Jet Impingement System. *Energies* **2019**, *12*, 1627. [CrossRef]
22. Zulueta, E.; Kurt, E.; Uzun, Y.; Lopez-Guede, J.M. Power control optimization of a new contactless piezoelectric harvester. *Int. J. Hydrogen Energy* **2017**, *42*, 18134–18144. [CrossRef]
23. Uriarte, I.; Zulueta, E.; Guraya, T.; Arsuaga, M.; Garitaonandia, I.; Arriaga, A. Characterization of recycled rubber using particle swarm optimization techniques. *Rubber Chem. Technol.* **2015**, *88*, 343–358. [CrossRef]
24. Gomez, T.; Herrero, I.; Rodilla, P.; Escobar, R.; Lanza, S.; De La Fuente, I.; Llorens, M.L.; Junco, P. European Union Electricity Markets: Current Practice and Future View. *IEEE Power Energy Mag.* **2019**, *17*, 20–31. [CrossRef]
25. BOE.es—BOE-A-2001-20850 Real Decreto 1164/2001, de 26 de Octubre, Por el Que se Establecen Tarifas de Acceso a las Redes de Transporte y Distribución de Energía Eléctrica. Available online: https://www.boe.es/buscar/act.php?id=BOE-A-2001-20850#:~{}:text=el%20art%C3%ADculo%209.3.-,3.,o%20inferior%20a%20450%20kW (accessed on 15 March 2021).
26. Martinez-Rico, J.; Zulueta, E.; Fernandez-Gamiz, U.; De Argandoña, I.R.; Armendia, M. Forecast Error Sensitivity Analysis for Bidding in Electricity Markets with a Hybrid Renewable Plant Using a Battery Energy Storage System. *Sustainability* **2020**, *12*, 3577. [CrossRef]
27. Muenzel, V.; de Hoog, J.; Brazil, M.; Vishwanath, A.; Kalyanaraman, S. Multi-Factor Battery Cycle Life Prediction Methodology for Optimal Battery Management. In Proceedings of the 2015 ACM Sixth International Conference on Future Energy Systems, New York, NY, USA, 14–17 July 2015; pp. 57–66. [CrossRef]

28. Batería Solar Estacionaria 236 Ah. C100, 6 vasos x 2 v. Sialsolhome. Available online: https://www.sialsolhome.com/producto/bateria-solar-estacionaria-gel-236-ah/ (accessed on 22 February 2021).

29. Martinez-Rico, J.; Zulueta, E.; de Argandoña, I.R.; Fernandez-Gamiz, U.; Armendia, M. Multi-objective Optimization of Production Scheduling Using Particle Swarm Optimization Algorithm for Hybrid Renewable Power Plants with Battery Energy Storage System. *J. Mod. Power Syst. Clean Energy* **2021**, *9*, 285–294. [CrossRef]

30. Kennedy, J.; Eberhart, R. Particle Swarm Optimization. In Proceedings of the ICNN'95—International Conference on Neural Networks, Perth, Australia, 27 November–1 December 1995; Volume 4, pp. 1942–1948. [CrossRef]

31. Chatterjee, A.; Siarry, P. Nonlinear inertia weight variation for dynamic adaptation in particle swarm optimization. *Comput. Oper. Res.* **2006**, *33*, 859–871. [CrossRef]

32. Nickabadi, A.; Ebadzadeh, M.M.; Safabakhsh, R. A novel particle swarm optimization algorithm with adaptive inertia weight. *Appl. Soft Comput.* **2011**, *11*, 3658–3670. [CrossRef]

33. Yu, X.; Liu, J.; Li, H. An Adaptive Inertia Weight Particle Swarm Optimization Algorithm for IIR Digital Filter. In Proceedings of the 2009 International Conference on Artificial Intelligence and Computational Intelligence, Shanghai, China, 7–8 November 2009; Volume 1, pp. 114–118. [CrossRef]

34. Yang, X.; Yuan, J.; Yuan, J.; Mao, H. A modified particle swarm optimizer with dynamic adaptation. *Appl. Math. Comput.* **2007**, *189*, 1205–1213. [CrossRef]