

Article

Method for SoC Estimation in Lithium-Ion Batteries Based on Multiple Linear Regression and Particle Swarm Optimization

Diego Castanho ^{1,†} , Marcio Guerreiro ^{2,†} , Ludmila Silva ^{3,†} , Jony Eckert ^{3,†} , Thiago Antonini Alves ^{4,†} , Yara de Souza Tadano ^{4,†} , Sergio Luiz Stevan, Jr. ^{1,†} , Hugo Valadares Siqueira ^{1,2,*,†}  and Fernanda Cristina Corrêa ^{1,†} 

¹ Graduate Program in Electrical Engineering (PPGEE), Federal University of Technology—Paraná (UTFPR), R. Doutor Washington Subtil Chueire, 330—Jardim Carvalho, Ponta Grossa 84017-220, PR, Brazil

² Graduate Program in Industrial Engineering (PPGEP), Federal University of Technology—Paraná (UTFPR), R. Doutor Washington Subtil Chueire, 330—Jardim Carvalho, Ponta Grossa 84017-220, PR, Brazil

³ Graduate Program in Mechanical Engineering, University of Campinas (UNICAMP), Campinas 13083-970, SP, Brazil

⁴ Graduate Program in Mechanical Engineering (PPGEM), Federal University of Technology—Paraná (UTFPR), R. Doutor Washington Subtil Chueire, 330—Jardim Carvalho, Ponta Grossa 84017-220, PR, Brazil

* Correspondence: hugosiqueira@utfpr.edu.br

† These authors contributed equally to this work.

Abstract: Lithium-ion batteries are the current most promising device for electric vehicle applications. They have been widely used because of their advantageous features, such as high energy density, many cycles, and low self-discharge. One of the critical factors for the correct operation of an electric vehicle is the estimation of the battery charge state. In this sense, this work presents a comparison of the state of charge estimation (SoC), tested in four different conduction profiles in different temperatures, which was performed using the Multiple Linear Regression without (MLR) and with spline interpolation (SPL-MLR) and the Generalized Linear Model (GLM). The models were calibrated by three different bio-inspired optimization techniques: Genetic Algorithm (GA), Differential Evolution (DE), and Particle Swarm Optimization (PSO). The computational results showed that the MLR-PSO is the most suitable for SoC prediction, overcoming all other models and important proposals from the literature.

Keywords: state of charge; lithium-ion battery; computational intelligence; electric vehicle; MLR



Citation: Castanho, D.; Guerreiro, M.; Silva, L.; Eckert, J.; Antonini Alves, T.; Tadano, Y.d.S.; Stevan, S.L., Jr.; Siqueira, H.V.; Corrêa, F.C. Method for SoC Estimation in Lithium-Ion Batteries Based on Multiple Linear Regression and Particle Swarm Optimization. *Energies* **2022**, *15*, 6881. <https://doi.org/10.3390/en15196881>

Academic Editor: Xianke Lin

Received: 12 July 2022

Accepted: 12 September 2022

Published: 20 September 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The automotive industry aims to develop environmentally friendly and sustainable future mobility. Based on that, developers are moving to new vehicle technologies that are more clean and efficient [1,2]. Therefore, hybrid and electric vehicles are the most suitable solutions for sustainable development [3–5]. Nowadays, several automotive manufacturers already offer battery-powered vehicles [6,7]. Two main advantages of using these vehicles are zero-emission of polluting gases [8] and more significant energy conservation [9,10]. Moreover, the data presented by Hazra and Reddy [11] indicate that the electric vehicle production will increase by 28% until 2026, reaching 7% of the market by 2030.

However, Electric Vehicles (EVs) have some disadvantages related to batteries, especially in terms of limited autonomy and charging time [12,13]. These issues, associated with a relatively lower improvement rate of the electric motors and battery technologies [14] and high battery costs [13,15], may hinder EVs popularization.

On the other hand, new lithium-based batteries already present acceptable performance regarding their safety and energy density [16], and many studies focus on solutions to estimate battery degradation [17] and extend their lifetime [18–20].

Some authors point out that the main characteristics of batteries are temperature, specific energy, and voltage. Meanwhile, the main battery types are Lead-acid, Nickel-Metal

Hydride (NiMH), and Lithium-Ion (Li-ion) [21,22]. Still, lithium-ion batteries have a long service life, reaching 3000 cycles at 80% discharge, having a fast charge, high voltage, and a high energy density compared to other types available for automotive applications [23,24]. They still have low self-discharge when subjected to idle periods, concerning NiCad and NiMH batteries [25].

Another common approach to EV studies is power management control (PMC), which aims to enhance the power distribution among the traction motors and the available energy storage devices [26,27]. On the other hand, Li et al. [28] showed that the power source degradation also influences power management strategies. Therefore, the PMC is a crucial feature to mitigate high-frequency, and high-current discharge rate cycles, which reduce the battery cycle life [29–31].

In this context, the battery state of charge (SoC) estimation accuracy is fundamental to ensure the performance of the PMC [32], as in the ones developed by Veneri et al. [16], Li et al. [33], and Eckert et al. [34,35]. In addition, the SoC is a crucial variable in the operation of microgrids and needs to be monitored in real time [36]. Considering a hybrid energy storage system in which the battery power is shared with other devices as ultra-capacitors [35], the energy management control also needs accurate inputs of the SoCs and efficiency of the all energy storage units to minimize the discharges [12,34]. Therefore, it is possible to conclude that the battery SoC estimation is crucial for establishing efficient control systems.

To date, recent and essential studies on SoC have been found in the literature. Xiong et al. [37] evaluated and summarized the different types of SoC estimation methods, such as Looking-Up Table Based Methods, Ampere-Hour Integral Method, and Model-Based Estimation Methods.

The study by Lipu et al. [38] presents a detailed classification of the recent data-driven state of charge estimation highlighting algorithm, input features, configuration, execution process, strength, weakness, and estimation error.

Analytical battery models such as the Shepherd model [39,40] can accurately define battery SoC based on the discharge profile. Moreover, this model can also estimate the battery life cycle by simulating the charge–discharge cycle profile. On the other hand, this model presented high computation cost [41,42] due to the complexity of solving the applied equations. The non-linearity in battery charging behavior is discussed in the work of Delfino et al. [43].

Hannan et al. [44] identified that conventional estimation methods are hardly affected by aging, temperature and external disorders. They noted that an adaptive filter algorithm could predict a non-linear dynamic state with satisfactory precision, lower computational cost, and high efficiency.

Zhang et al. [45] proposed an improved backpropagation neural network (BP) method to estimate SoC based on current, voltage, and temperature. In addition, a particle swarm algorithm was adopted to caliber the weighting coefficients. The study pointed out that the experimental results of the developed algorithm presented reasonable accuracy in normal driving conditions.

Ali et al. [46] presented an algorithm and the process of executing model-based and data-driven SoC estimation methods. Still, the authors did not discuss data-driven SoC estimation methods based on deep learning. Finally, Vidal et al. [47] estimated the SoC using machine learning. However, the study did not consider SoC estimation methods and their main problems.

A variety of approaches for estimation battery SoC and capacity with their merits and drawbacks are presented below.

Lu et al. [48] use a geometrical approach based on the Laplacian Eigenmap method for Li-ion battery capacity estimation. The authors utilize four geometrical features sensitive to slight changes in the performance degradation of a Li-ion battery. The maximum and minimum of errors estimated capacity and measured capacities are 4.48% and 1.06%, respectively.

Yang et al. [49] proposed a Gaussian Process Regression model based on the charging curve. Four specific parameters were extracted from these curves and used to calibrate the model rather than use the cycle numbers. The estimation error obtained was less than 6%.

Li et al. [50] proposed a random forest regression as a technique to battery capacity estimation based on features extracted from the charging voltage and capacity measurements. The presented technique was able to evaluate the health states of different batteries under varied cycling conditions, obtaining a performance analysis based on RMSE of less than 1.3%.

Stroe and Schaltz [51] reported an incremental capacity analysis (ICA) technique for estimating the capacity fade and subsequently the SoH of LMO/NMC-based EV lithium-ion batteries, using four SoH estimation models. They reported a root mean square error less than 3% normalized.

A method based on transfer learning was proposed to monitor the state of health of batteries by Li et al. [52], considering the data sets of four batteries in different working conditions. The reported results presented that the state of health was estimated effectively with an RMSE less than 2.5% after being processed by using a semi-supervised transfer component analysis algorithm.

A prediction model for voltage profiles proposed by Shu et al. [53] used the extreme learning machine (ELM) algorithm with the short-term charging data. The reported results demonstrated that the maximum absolute error was restricted to 2%.

Li et al. in [54] presented a neuron selection method using a recursive algorithm to prune the transferred model. The accuracy results and computational efficiency presented were compared with another Convolutional Neural Network (CNN), obtaining a size reduction greater than 68% and achieving more than 80% computational savings. The results presented an RMSE smaller than 6.5% for different tested charge profiles. Meanwhile, in [55], a CNN-based battery capacity estimation method using a partial charging segment of the direct measurements (current, voltage, and cell surface temperature) was proposed. The method was applied to two battery degradation datasets and achieves RMSE values of less than 2.93%.

Cheng et al. [56] presented a prognostic ensemble framework that combined multiple individual prognostic algorithms to improve the accuracy and robustness of battery capacity estimation. The validation data were based on induced ordered weighted averaging (IOWA). The proposed ensemble prognostic was compared with four individual prognostic methods (BF neural network, the SVM, the GM, and the ARIMA model), and the results presented a decreased prognostic error in terms of the root mean square error of 5.08%.

Li et al. [57] proposed a Bayesian nonparametric approach based on an advanced smoothing method based on a Gaussian filter algorithm for lithium battery SoH estimation based on the health indexes of partial charging IC curves. The mean and covariance functions of the proposed method were applied to predict the battery state of health and the model uncertainty. The results presented a maximum relative error of 3%.

Shen et al. [58] proposed a deep convolutional neural network (DCNN) for battery-capacity estimation based on the voltage, current, and charge capacity measurements during a partial charge cycle. The results presented an RMSE smaller than 2%. Zhao et al. [59] used an online estimation model based on an approximate belief rule base and a hidden Markov model (ABRB-HMM), using historical data and expert knowledge, applied to a satellite lithium-ion battery. The HMM model is used to modify model parameters by indirect observations and complete the iteration calculation of capacity. The authors report an MSE of 0.0056.

In this way, Table 1 presents a summary of the main works of the state-of-the-art of estimation battery SoC and capacity.

As noted in the literature review, the studies cited were limited to presenting the proposals developed for estimating the SoC, pointing out only the variables involved and the model and algorithm used. Therefore, this study proposes to analyze a model to estimate SoC in lithium-ion batteries for electric vehicles, seeking better accuracy and

processing speed results, presenting tests, advantages, and improvement opportunities for future studies. This work performed the training of models through the initial capacity curve. The prediction from four internationally recognized test profiles was used for testing and validating the results.

Table 1. State-of-the-art summary of SoC estimation methodologies.

Methodology of SoC Estimator/Forecast	Battery Type Used	Performance Index	Precision Related	Reference
Geometrical approach based on Laplacian eigenmap method	NASA 18650	RMSE	<3.84%	[48]
Gaussian process regression model	NASA 18650	RMSE	<3.45%	[49]
Random forest regression	Nickel-Manganese-Cobalt (NMC) batteries	RMSE	1.3%	[50]
Incremental capacity analysis technique	Prismatic Li-ion Battery	RMSE	2.99%	[51]
Semi-supervised transfer component analysis	NASA 18650	MAE	<1.29%	[52]
Short-term charging profiles	NCM/graphite	RMSE	2%	[53]
Convolutional Neural Networks (CNN)	Commercial lithium-ion	RMSE	<2.54%	[55]
An ensemble prognostic method	NASA 18650	RMSE		[56]
Incremental capacity analysis and Gaussian process regression	NASA 18650	RMSE	1.38%	[57]
Deep Convolutional Neural Network (DCNN)	NASA 18650	RMSE	<2%	[58]
Approximate belief rule base and hidden Markov model—(ABRB-HMM)	Cylindrical battery (20 Ah, 4.1 V)	MSE	0.056	[59]

Therefore, the Multiple Linear Regression (MLR) and the Generalized Linear Model (GLM) are used, which allows the implementation of low-order linear models, characterizing the dynamic behavior observed in the battery system. The GLM was chosen because the automobile industry prefers performances based on simplified, easily identifiable models calibrated through practical experiments [60].

Cervelló Royo and Guijarro [61] employed four Machine Learning-based prediction methods including gradient boosting machines (GBM), random forest (RF), generalized linear models (GLM) and deep learning (DL) to address the estimation of market trends as a comparison analysis by accuracy rate (%). Antón et al. [62] proposed an SoC estimation method based on a new hybrid model that combines multivariate adaptive regression splines (MARS) and particle swarm optimization (PSO). The results presented equivalent accuracy values to other more sophisticated techniques but at a lower computational cost.

Belotti et al. [63] presented an improved version of the GLM to estimate the number of hospital admission for respiratory disease caused by PM10 air pollution and meteorological variables as ambient temperature and relative humidity. To calculate the GLM optimum parameters, they used three bio-inspired optimization algorithms (PSO, GA and DE (Differential Evolution)). The computational results presented that the new GLM purpose achieved better performances when compared to the traditional approach.

Meanwhile, Ahmadzadeh et al. [64] applied PSO to Multilayer Perceptron (MLP) ANN to optimize the values of weights and biases parameters in order to prevent the basic gradient descent method from being trapped in the local minimum, and the proposed method's

application was evaluated for prediction application. The experimental results show that the proposed method exhibits superior performance to MLR for prediction applications.

In the light of the aforementioned, it is noted that the MLR and GLM models are used in several applications; in this way, this research provides the first occasion for which these models will be used for SoC estimation, and combining them with optimizers seeks satisfactory results from this investigation. Most of the linear approaches are tuned using linear predictors. So, three computational intelligence techniques were used to optimize MLR and GLM models: Genetic Algorithm (GA), Differential Evolution (DE), and Particle Swarm Optimization (PSO) [65]. These techniques are well known in the optimization literature, presenting important results in real-world problems [66,67].

Furthermore, the method allows for a presentation of the operating cost in two stages: (i) training that can take place offline, requiring a particular time to be completed, and (ii) real-time testing, requiring low cost that can be sent to any micro-controlled system, as it uses simple mathematical operations.

The reminder of this paper is as follows: Section 2 presents the models addressed to predict SoC; Section 3 shows the bio-inspired methods used to calibrate the prediction models; Section 4 presents the methodology addressed to perform the predictions; Section 5 shows the results and a critical analysis for a case study containing 24 scenarios; Section 6 contains the main conclusions and outlines future works.

2. SoC Prediction Models

2.1. Multiple Linear Regression Models—MLR

MLR are used to analyze the relationship between more than two variables. The goal is to model the association between the dependent variable (response variable) and the desired output to be predicted with more than one independent variable (predictor or explanatory variable) [68].

In this sense, the response variable y will depend on two or more independent variables as follows (Equation (1)):

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_q x_q \quad (1)$$

where β_0 represents the intersection and the other parameters β_i , $i = 1, 2, 3, \dots, q$, are called the regression coefficients. Usually, the regression coefficient estimates are calculated by the minimum squares method. However, in this work, three different optimization models were applied.

Sometimes, an MLR is not appropriate. One way to overcome its disadvantages and improve the predictions is to divide the dataset into subgroups and fit each subgroup into a separate model instead of one for the entire dataset. Then, we chose to apply linear splines to the multiple linear regression model (Spline-MLR), which was employed by dividing the dataset into two subgroups (three knots), 50% of the database on each subgroup. A model, described by Equation (1), is adjusted for each subgroup, resulting in different regression coefficients. As in the MLR case, three optimization models will calculate the regression coefficients.

2.2. Generalized Linear Model—Poisson Regression

GLM can be understood as a class capable of representing continuous responses between one or more predictor variables in a wide variety of reactions that are generally not distributed [69,70]. GLM is appropriate for estimating the behavior of non-linear systems through non-linear relationships between predictors and expected values [70]. They are defined as a probability distribution member of the exponential family of distributions, according to Equation (2):

$$\eta = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_q x_q \quad (2)$$

where η is the link function, a function of the response variable mean that can have different distributions; β values are the regression coefficients to be estimated, and the x_i , $i = 1, 2, 3, \dots, q$ are the independent variables.

The slight difference between MLR and GLM is the link function, which is a function of the response variable average that can have different distributions.

One very often used link function is the logarithmic one, resulting in the Poisson regression, which is given by Equation (3):

$$\log(\mu) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_q x_q \quad (3)$$

in which μ is the response variable average, and β represents the regression coefficients to be determined considering x_i , $i = 1, 2, 3, \dots, q$ independent variables.

Usually, the maximum likelihood method is applied to find the regression coefficients. In this work, different methods will be applied.

3. Optimization Models

This section presents the optimization techniques used to calibrate the models discussed in Section 2.

3.1. Genetic Algorithm

The first proposal of a bio-inspired optimization method was the Genetic Algorithm (GA) [71]. At present, it remains the most used evolutionary approach [66,72,73].

The roots of the inspiration came from the evolution by natural selection theory developed by Charles Darwin [74]. The initialization of the method is performed by creating a set of candidate solutions named individuals or chromosomes, forming a population. Each individual is a vector containing genes representing the values of the problem's parameters addressed [75]. The degree of adaptation of a chromosome is named fitness, which is directly related to the quality of the solution regarding the system of interest.

The GA presents three operations: selection, crossover, and mutation. The first defines the chromosomes that will pass through crossover and mutation, which are the genetic operators. The most used selection techniques are roulette wheel and binary tournament [76]. The last approach can be used considering the "death" or the deletion of the loser.

Then, the crossover is applied considering those two individuals (parents). These exchange some genes, creating another two chromosomes, or the offspring. The crossover is repeated until the number of individuals in the current population achieves the original value. This operator is an exploitation method, because it performs a local search [71].

The other operator is the mutation. It occurs with a small probability. Some genes are changed considering the entire population of genes, perturbing their values according to some pre-defined distribution. When the entire loop is complete, the new generation will pass through the same steps until it achieves the stop criterion. This procedure allows performing a global search or finding some unexplored region of the cost function [75].

Regarding the crossover, we addressed the one-point crossover approach. In this work, we chose the binary tournament procedure in the selection stage. Initially, two individuals are selected at random. They compete, and the one with the highest fitness is the winner. The selective pressure is lower than in the roulette wheel [74]. For the mutation, we used the Gaussian distribution to define the perturbation in the selected genes.

3.2. Differential Evolution

Differential Evolution (DE) is another evolutionary approach for minimizing non-differentiable and non-linear continuous functions. It is also based on selection, crossover, and mutation operators as in GA, but it presents different characteristics [77]. Each agent is named a vector.

The DE operation occurs as follows [77]: start with a uniform and randomly generated population. Each vector must have genes (parameters values). At random, a vector is

chosen and named the Trial Vector. The mutation process is carried out, resulting in the Donor Vector, which is defined by Equation (4).

$$\mathbf{x}^{\text{Donor}} = \mathbf{x}^{\text{Best}} + (F(\mathbf{x}^{\text{k1}} - \mathbf{x}^{\text{k2}})) \quad (4)$$

where $\mathbf{x}^{\text{Best}}$ represents the best vector evaluated at each iteration, F is a variable that adjusts to the universe of speech, which is defined according to the magnitude of the values involved in the optimization (usually between 0.1 and 2), and $\mathbf{x}^{\text{k1}}$ and $\mathbf{x}^{\text{k2}}$ are two vectors selected randomly within the population.

The crossover is carried out gene by gene, resulting in the Trial Vector. A value between 0 and 1 is randomly selected. When the value is less than or equal to the crossover rate, the Donor Vector gene is selected. Otherwise, the gene for the Trial Vector itself is chosen.

Finally, the Trial Vector and the current individual apply the selection process through a greedy selection considering the fitness (the best of the two passes onto the next generation). This process continues until the pre-established stop condition, which is usually the maximum number of iterations.

3.3. Particle Swarm Optimization (PSO)

In contrast to the evolutionary approaches, Particle Swarm Optimization (PSO) is a swarm-inspired metaheuristic. The inspiration came from the social behavior of groups of animals, such as schools of fish or flocks of birds. It was firstly presented by Kennedy and Eberhart [78].

The main characteristic of the PSO is that simple agents named particles work collectively in an intelligent way [79]. Each agent presents a position and a velocity, which are presented in Equations (5) and (6).

$$\mathbf{v}_i^{n+1} = \omega \mathbf{v}_i^n + r_1 c_1 (\mathbf{p}_{best} - \mathbf{x}_i^n) + r_2 c_2 (\mathbf{g}_{best} - \mathbf{x}_i^n) \quad (5)$$

$$\mathbf{x}_i^{n+1} = \mathbf{x}_i^n + \mathbf{v}_i^{n+1} \quad (6)$$

where $\mathbf{x}_i^n$ and $\mathbf{v}_i^n$ are the position and velocity of the i -th particle in the n -th iteration, ω is the inertial weight, r_1 and r_2 are integer numbers generated in the interval $[0, +1]$, c_1 is the personal acceleration factor, c_2 is the social acceleration factor, and $\mathbf{p}_{best}$ is the best per.

4. Methodology

The methodological procedure of this work begins with determining the charging and discharging behavior of a battery set based on an experimental database. This behavior was modeled through linear modeling algorithms (GLM and MLR), which were adjusted through three optimization techniques (GA, DE and PSO). Each model has been tested on 24 different bases generated in 4 different usage profiles, 3 different temperatures, and 2 different load situations. Finally, the data were evaluated through 3 different metrics based on errors analysis to obtain the model with the best representation of the tested data.

Thus, first, the database is described, later, the calibration procedure of the prediction models is outlined, and finally, the evaluation metrics are presented.

4.1. Database

We use a database used in this research that was made available by the CALCE[®] Battery Research Group—University of Maryland [80]. It used cylindrical battery samples of INR 18650-20R Li-NiMnCoO₂/Graphite Lithium-ion cells in their tests. It presents two parameters (Low-current in open-circuit voltage (OCV), denominated estimator 1; and Incremental-current OCV, denominated estimator 2) for different dynamic tests and in different temperatures.

The samples have a nominal voltage of 3.6 V and a capacity of 2000 mAh. The tests were made in voltage between 2.5 and 4.2 V, with a maximum current of 22 A at 25 °C and an operating temperature ranging from 0 to 50 °C [81].

The four test profiles were used to validate the prediction models. The data were collected according to different test profiles specified in the US Advanced Battery Consortium (USABC) test procedures, occurring from 80% and 50% of initial SoC up to the final value of 10%. This SoC range occurs at operation in electric vehicles [82,83]. The four test profiles are:

- **Federal Urban Driving Schedule—FUDS:** It is a variable energy discharge test applied to represent the driving effects of an EV. This test includes regenerative braking, which is one of the essential characteristics of electric vehicles;
- **Dynamic Stress Test—DST:** It is a test profile of the variable energy discharge regime, where the battery is charged and stabilized at a controlled temperature, according to the procedure specified by the manufacturer, varying according to the model, technology, and employability of the battery under study [83];
- **Highway Driving Schedule—US06:** Represents an aggressive driving test carried out on the highway, with a maximum speed of 129.2 km/h and a maximum acceleration of 3.2 m/s^2 [81,84];
- **Beijing Dynamic Stress Test—BDST:** Indicates a quantity of information about the vehicle's operation and usage patterns, including the acceleration, speed, and deceleration, among others [81].

Thus, the database available on the CALCE website [80] comprises 25 independent datasets. The first one, with voltage, current, load capacity (Ah), and discharge capacity, among other data, was measured with the “Arbin BT2000 battery test system” to describe the initial capacity of the test battery. Subsequently, 1 set of data was created for each of the 4 test profiles (DST, FUDS, US06 and BJDST) for 3 temperatures (0° , 25° and 45°C) and 2 battery levels (50% and 80%).

4.2. Training Phase

The training set comprises 13,400 samples containing information about the charge and discharge capacities (to obtain SoC), voltage, and current. An initial capacity curve obtained experimentally by [80] was used to calibrate the models described by different algorithms, obtaining information about the battery behavior. Figure 1 presents this proposal.

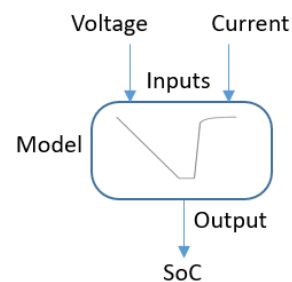


Figure 1. Battery SoC Model, based on the voltage and current information.

Figure 2a presents the target SoC model, while Figure 2b shows the current and voltage profiles obtained through the initial capacity test. The negative current indicates the battery is discharged, and the positive current shows its charge. It can be seen that even at a constant current and controlled temperature, the voltage has no linear behavior.

The training task is to find the coefficients of the prediction models. They were implemented in the object-oriented Python language. The structure used in each model is: each algorithm was executed 50 times, choosing the best execution, or the configuration that presented the lowest *AE* was selected. We highlight that we also tested the *MAE* as a cost function, but *AE* performed better.

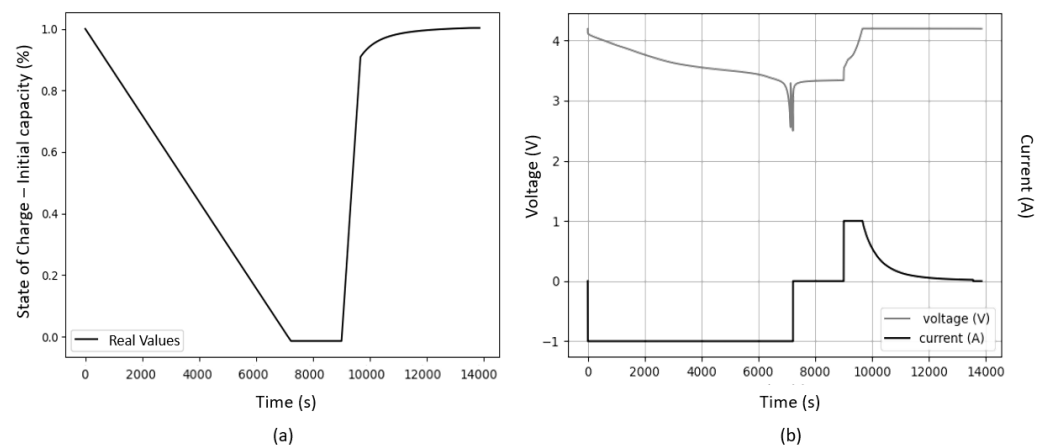


Figure 2. Initial capacity of the battery. (a) Battery SoC model. (b) Current and voltage profiles for the initial capacity.

As mentioned, two linear models are used—MLR and GLM—being the first endowed or not of the linear spline. These three approaches were combined with three different optimization techniques, GA, DE, and PSO, totaling nine variations, as presented below:

- MLR-GA: Multiple Linear Regression optimized by a genetic algorithm using 1000 individuals, one-point crossover at a rate of 70%, selection by a death tournament, and a mutation rate of 10%;
- MLR-DE: Multiple Linear Regression optimized by a differential evolution using 1000 individuals, binary crossover at a rate of 20%, greedy selection where the target vector and trail vector are compared, and the one with the highest fitness value is selected and a mutation of the best type, where the individual of more excellent fitness is added to the changeover at a rate of 20%;
- MLR-PSO: Multiple Linear Regression optimized by a PSO, where 100 particles were used, the constants c_1 and c_2 are equal to 2 and the constant ω is 0.8;
- SPL-MLR-GA: Multiple Linear Regression with linear interpolation with 3 nodes and 14 degrees of freedom optimized by a genetic algorithm, in which 1000 individuals were used, one-point crossover at a rate of 70%, death tournament selection and a mutation rate of 10%;
- SPL-MLR-DE: Multiple Linear Regression with linear interpolation with 3 nodes and 14 degrees of freedom optimized by differential evolution, using 1000 vectors, binary crossover at a rate of 20%, greedy selection and a mutation of the best type, being the individual with the highest fitness added in the mutation to a rate of 20%;
- SPL-MLR-PSO: Multiple Linear Regression with linear interpolation with 3 nodes and 14 degrees of freedom optimized by a PSO, where 100 particles were used, constants c_1 and c_2 equal to 2 and ω is 0.8;
- GLM-GA: Generalized Linear Model with Poisson regression optimized by a genetic algorithm, with 1000 individuals, point crossover at a rate of 70%, death tournament selection and a mutation rate of 10%;
- GLM-DE: Generalized Linear Model with Poisson regression optimized by differential evolution, 1000 vectors, binary crossover with a rate of 20%, greedy selection, and a mutation of the best type, in which the individual with the highest fitness is added to the mutation at a rate of 20%;
- GLM-PSO: Generalized Linear Model with Poisson regression optimized by a PSO, where 100 particles were used, constants c_1 and c_2 are equal to 2, and the constant ω is 0.8.

Note that the free parameters of the optimization models were chosen based on previous tests. A grid search was performed considering the parameters of the models. Therefore, we selected the best configuration for each algorithm. The search was as follows:

- GA—crossover rate: from 50% to 90%; mutation rate: from 5% to 20%;
- DE—crossover rate: from 10% to 50%; mutation rate: from 5% to 20%;
- PSO—constants c_1 and c_2 : from 1.5 to 2.5; ω : from 0.5 to 0.9.

The intervals used to define the parameters are well known in the specialized literature [75]. For all algorithms, we tested from 100 to 1000 agents.

4.3. Metrics for Performance Evaluation

As metrics for evaluating performance, Absolute Error (AE), Mean Square Error (MSE) and Mean Absolute Error (MAE) are commonly used [85].

The AE is presented in Equation (7). This metric allows the exact notion of the distance from the training curve to the actual curve, allowing to obtain, depending on the model. This curve represents precisely the behavior of the function to convey.

$$AE = \sum_{t=1}^N |d_t - y_t| \quad (7)$$

such that N is the number of samples, d_t is the real value of SoC, and y_t is the value predicted by one of the linear models.

The MAE, presented in Equation (8), represents the average gap between actual and predicted values.

$$MAE = \frac{1}{N} \sum_{t=1}^N |d_t - y_t| \quad (8)$$

Finally, the MSE presents as a characteristic a more significant penalty for the highest errors because they largely influence the forecast's final value and lesser punishment for the model with more minor errors [85]. Equation (9) presents such a metric:

$$MSE = \frac{1}{N} \sum_{t=1}^N |d_t - y_t|^2 \quad (9)$$

5. Results

This section presents the computational results achieved considering the nine model variations regarding the training step (for the training set) and the test for the 24 independent databases.

5.1. Model's Training

The training is performed considering as output the initial capacity curve depicted in Figure 2a and the current and voltage as inputs. The optimization techniques (GA, DE, and PSO) tune the coefficients of the three proposals (MLR, MLR with spline interpolation and GLM) to minimize the absolute error among the real output data and the model's response.

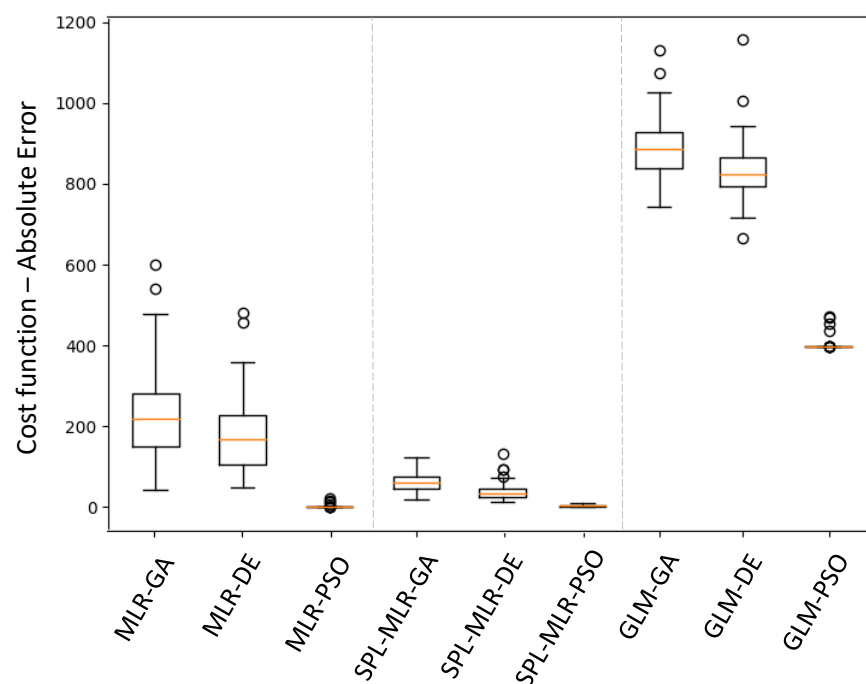
The absolute error—AE (also understood as accumulative error) was used because it leads to more pronounced differences in the approximations of the curves obtained with GA and DE. In these cases, some values are more discrepant.

The objective of this training is to obtain the coefficients that represent the set of batteries. These coefficients can be determined by approximating the initial capacity curve.

Table 2 presents the best AE regarding 50 independent executions of each optimization algorithm. Note that the fitness function is the normalized AE in that the fitness must be maximized, lying in the interval $[0, +1]$. Figure 3 shows the boxplot graphic regarding the dispersion of the 50 runs considering the AE. The acronyms follow the definitions of Section 4.2.

Table 2. Results obtained from the optimization performed for the training algorithms.

Model	Absolute Error (AE)	Fitness
MLR-GA	4.31×10^1	0.0226
MLR-DE	4.84×10^1	0.0202
MLR-PSO	6.00×10^{-8}	0.9999
SPL-MLR-GA	2.26×10^1	0.0424
SPL-MLR-DE	1.46×10^1	0.0641
SPL-MLR-PSO	1.10×10^{-3}	0.9989
GLM-GA	7.41×10^2	0.0013
GLM-DE	6.64×10^2	0.0015
GLM-PSO	3.97×10^2	0.0025

**Figure 3.** Boxplot for 50 executions of each metaheuristic.

Friedman's test was applied to verify if the results were significantly distinct. The p -values were much lower than 0.05, indicating that using different models and optimizers led to different results statistically.

In relation to the results obtained in Table 2 and Figure 3, it is possible to analyze important aspects. The first is that the MLR proposals overcame the GLM considering the average error and the best general performance. This is clear in terms of AE and fitness.

Regarding the use of the spline interpolation in MLR, we observe that its application led to a decrease in the dispersion and the final error when the GA and DE are the optimizers. However, for the PSO, we noted that results are close for both MLR proposals in terms of dispersion, best, and average performance, the standard MLP being slightly better. In this sense, it is clear that the PSO was capable of finding the best set of coefficients, reaching the minor error for all proposals.

Figures 4–6 present the curves obtained during the training step, considering each optimizer, the target (SoC) and the output response of the model, considering the best of 50 simulations.

Figure 4 shows the best approximation obtained for training through the GA. It can be seen that the GLM-GA did not obtain a good approximation. The SPL-MLR-GA had difficulty representing the curve close to the nodes, and the MLR-GA presented a high error at the initial point and approximately 7000 s. Observe the zoom at the last inflection point (close to 10,000 s) and how close the MLR was to the real curve.

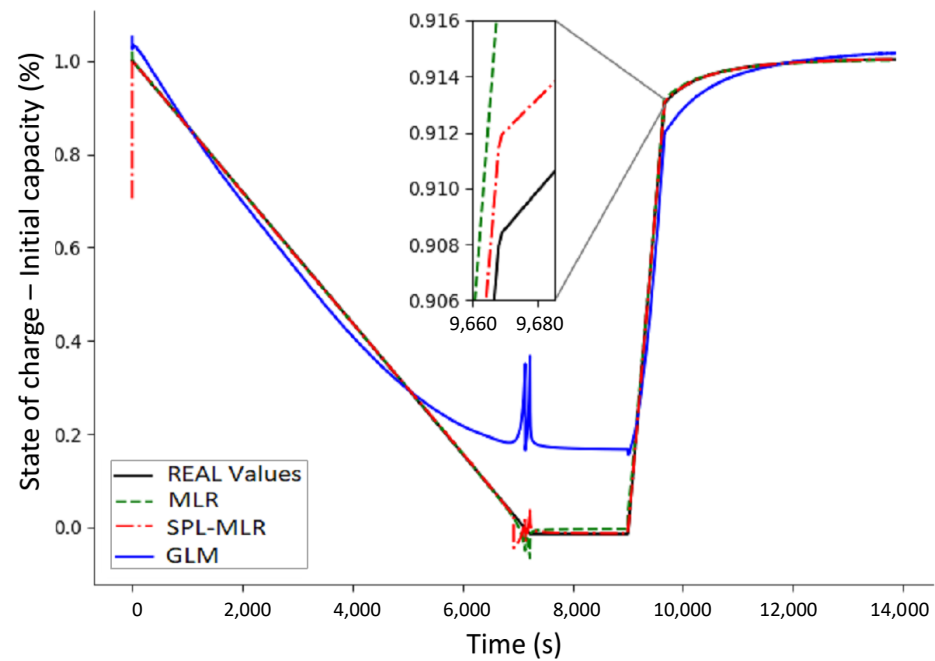


Figure 4. Initial Capacity Curve for GA Training.

Figure 5 presents the best approximation obtained for training through DE. The approximation of each model was better at the initial point and the first inflection point (7000 s). The zoom reveals the MLR without spline followed close to the real curve. However, once against, the GLM did not perform well.

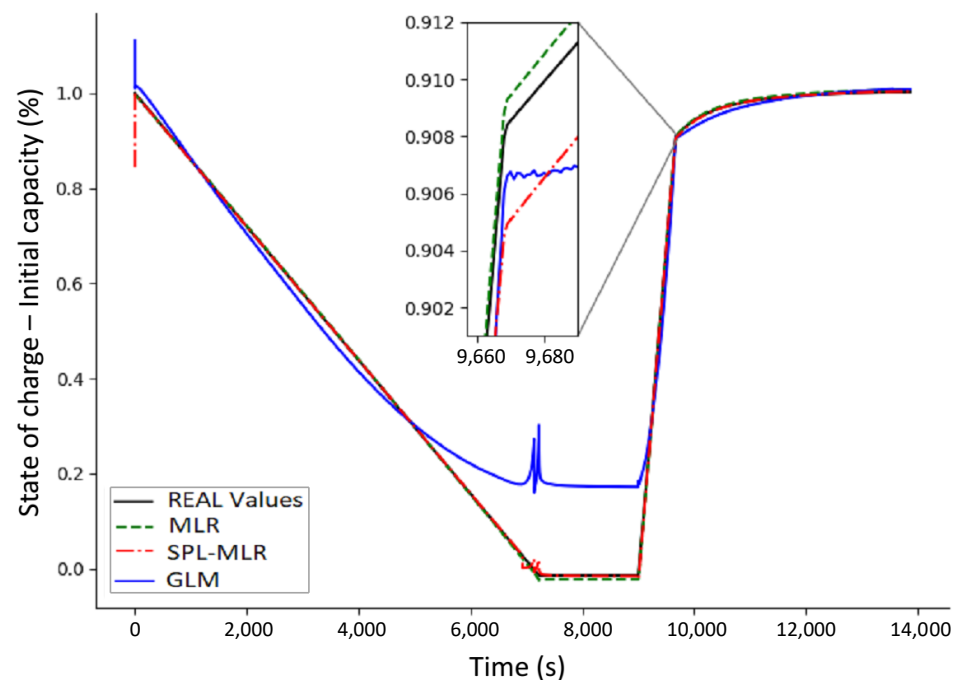


Figure 5. Initial Capacity Curve for DE Training.

Figure 6 presents the best approximation obtained for the PSO. Despite the GLM still not finding a good approximation, all models increase their performance. For both MLRs, the approximation is very close to the target, revealing that the PSO is the most satisfactory optimizer, especially for the MLR without splines.

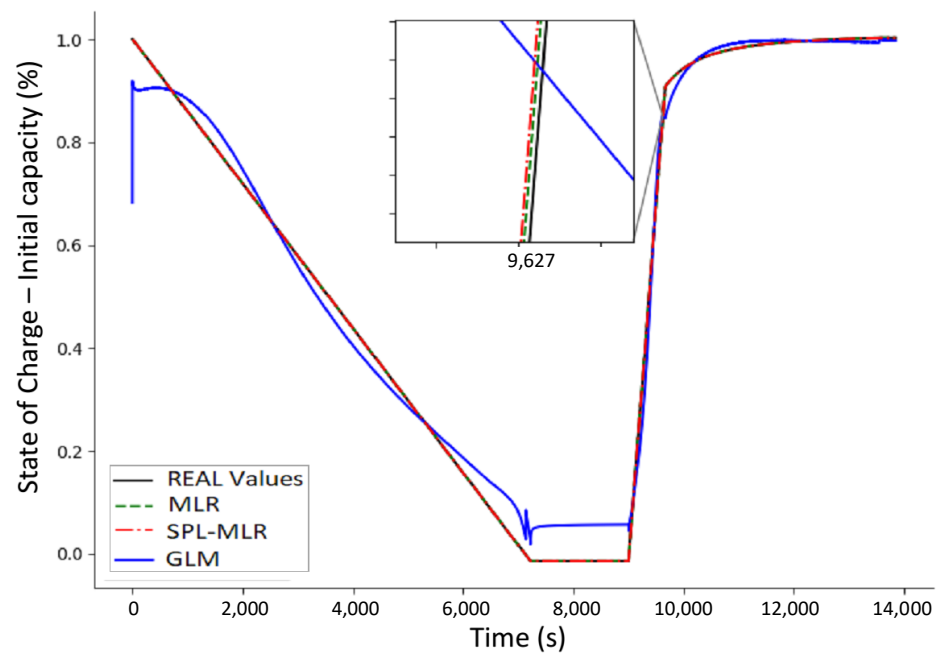


Figure 6. Initial Capacity Curve for PSO Training.

In summary, it is possible to state that the MLR-PSO algorithm obtained the best calibration, with an absolute error close to zero on the training step.

5.2. Models Evaluation and Discussion

After adjusting the nine model variants, we performed a test phase. We fixed the parameters found in training and applied 24 new databases that were not used before. The goal is to verify the approximation capability of the trained models, considering situations close to the real cases that can occur when using an electric vehicle.

The tests consider three temperatures controlled within the battery's operating range: 0 °C, 25 °C, and 45 °C, with the initial SoC at 50% and 80%. We use the databases DST, FUDS, US06, and BJDST previously described in Section 4.1. It is also considered an error metric for the AE, MAE, and MSE. Tables 3–5 show the results obtained, which present some patterns that can be discussed in considering distinct points of view.

A general comparison among models reveals that the MLR-PSO achieved the minor errors, which was followed by the SPL-MLR-PSO. In many cases, the errors are very close to zero, demonstrating the suitability of the proposal. Observing Table 2 (Section 4.2), it is possible to see that the results of MLR and SPL-MLR are not close despite being small. As shown in Figure 3, there is a difference of 1×10^{-5} ; that is, the SPL-MLR has an absolute error of 100,000 times the value of the MLR error. Furthermore, in validation, this difference is even more significant. Regarding the increase in the absolute error of the SPL-MLR-PSO, it is not direct to affirm whether it would be a bias problem; this increase is due to the model itself and not to distortion. The use of GLM, however, proved to be inadequate for this problem, since it reached the worse error values, considering all metrics. This behavior is similar to those found in the training process.

A complementary remark can be seen from Figure 7, which highlights the behavior of the models tested against the AE metric for the US06 cycle. Like this, the first point of discussion is that the initial value of the SoC does not interfere in the approximation capability of the models tested. The second point is that even with the variation in battery temperature, there were no significant changes in the approximation ability of the models, except for GLM-PSO, which produced a greater accumulation of errors at the 0 °C temperature.

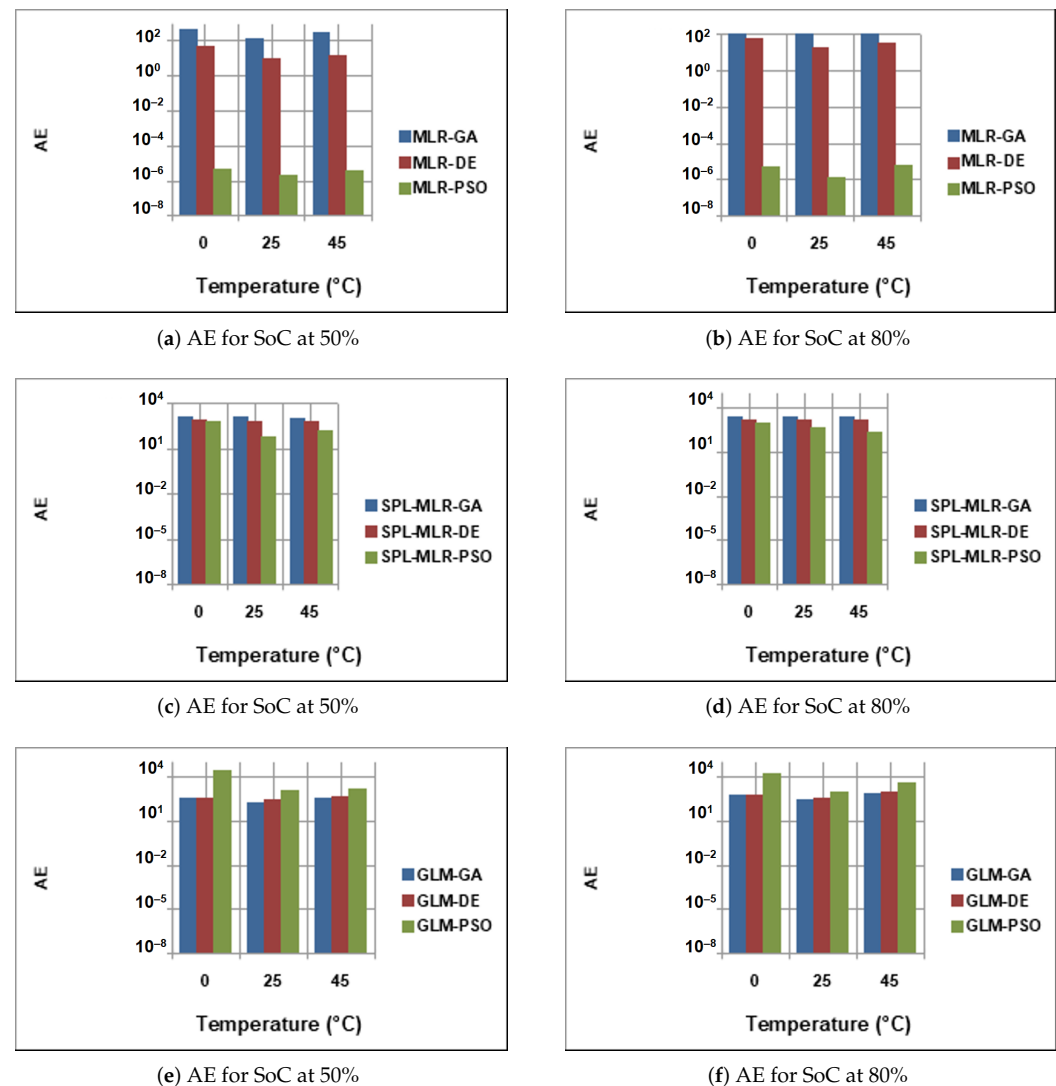


Figure 7. General comparison among models for the US06 cycle.

It is important to note that the AE metric was chosen for the graphical analysis, because it considers the magnitudes of accumulated errors, and also, the US06 cycle was emphasized for better representing the real driving conditions, with higher speeds and higher acceleration stages that represent a much more aggressive driving behavior. Considering the observations, it seems to be clear that the use of the MLR is more suitable for this specific problem.

Comparing the optimization search reached by the bio-inspired metaheuristics, in Table 3, it is clear that the adjustment provided by the PSO led to the smallest errors, presenting the best generalization considering all databases. Note that the errors are minor for the three metrics addressed for all the 24 cases.

In Table 4, there are just four cases in which the PSO was not the winner, three related to DST and one to FUDS. Regarding Table 5, we observe a distinct behavior since the GA stood out for most of the smallest errors, especially in 16 cases. Here, the PSO led to the best performances in four scenarios.

As widely known and resorting to the “no free lunch theorem”, there is no absolute method capable of solving all problems. It means that the performance of an algorithm is problem dependent. The evolutionary approaches (DE and GA) are generational, while the swarm-based method (PSO) is explanatory in the sense that the agents are moved along the search space. Evolutionary algorithms are better in exploration (global search)

while the swarm-based ones are better in exploitation (local search or fine-tuning) [86]. This is because the operators of each algorithm are distinct. Therefore, it seems to be clear that for this specific problems, exploitation is more needed than global search due to the characteristics of the cost function considering the best model: MLR-PSO. However, when the estimator is changed, the shape of the cost function is also modified. This is interesting, since the cost function may present a different shape, which can influence the search capability of the optimizers. In this case, the GA stood out for the GLM.

Table 3. Test results for all profiles at different SoC and temperatures, to MLR algorithm.

			50% SoC			80% SoC		
			AE	MAE	MSE	AE	MAE	MSE
DST	0 °C	MLR-GA	5.44×10^2	1.02×10^{-1}	1.08×10^{-2}	8.69×10^3	9.35×10^{-1}	8.74×10^{-1}
		MLR-DE	6.63×10^1	1.24×10^{-2}	1.68×10^{-4}	5.49×10^2	5.90×10^{-2}	3.50×10^{-3}
		MLR-PSO	6.42×10^{-6}	1.20×10^{-9}	1.47×10^{-18}	1.06×10^{-4}	1.14×10^{-8}	1.30×10^{-16}
	25 °C	MLR-GA	5.38×10^1	1.02×10^{-2}	2.87×10^{-4}	8.98×10^3	9.52×10^{-1}	9.08×10^{-1}
		MLR-DE	1.30×10^1	2.46×10^{-3}	9.60×10^{-6}	5.32×10^2	5.64×10^{-2}	3.19×10^{-3}
		MLR-PSO	6.59×10^{-7}	1.25×10^{-10}	2.44×10^{-20}	1.09×10^{-4}	1.16×10^{-8}	1.34×10^{-16}
	45 °C	MLR-GA	5.04×10^3	9.40×10^{-1}	8.83×10^{-1}	4.39×10^2	4.66×10^{-2}	2.40×10^{-3}
		MLR-DE	3.23×10^2	6.01×10^{-2}	3.62×10^{-3}	2.76×10^1	2.93×10^{-3}	1.17×10^{-5}
		MLR-PSO	6.10×10^{-5}	1.14×10^{-8}	1.29×10^{-16}	5.51×10^{-6}	5.85×10^{-10}	3.73×10^{-19}
FUDS	0 °C	MLR-GA	6.42×10^2	1.13×10^{-1}	1.32×10^{-2}	9.02×10^2	9.49×10^{-2}	9.43×10^{-3}
		MLR-DE	7.52×10^1	1.32×10^{-2}	1.91×10^{-4}	9.75×10^1	1.03×10^{-2}	1.25×10^{-4}
		MLR-PSO	7.54×10^{-6}	1.33×10^{-9}	1.79×10^{-18}	1.10×10^{-5}	1.16×10^{-9}	1.38×10^{-18}
	25 °C	MLR-GA	3.14×10^3	5.58×10^{-1}	3.12×10^{-1}	1.45×10^2	1.49×10^{-2}	3.44×10^{-4}
		MLR-DE	1.96×10^2	3.49×10^{-2}	1.23×10^{-3}	2.29×10^1	2.35×10^{-3}	9.08×10^{-6}
		MLR-PSO	3.76×10^{-5}	6.68×10^{-9}	4.47×10^{-17}	1.65×10^{-6}	1.70×10^{-10}	4.25×10^{-20}
	45 °C	MLR-GA	3.07×10^2	5.49×10^{-2}	3.31×10^{-3}	5.53×10^2	5.69×10^{-2}	3.55×10^{-3}
		MLR-DE	1.41×10^1	2.52×10^{-3}	9.50×10^{-6}	3.33×10^1	3.42×10^{-3}	1.58×10^{-5}
		MLR-PSO	4.34×10^{-6}	7.75×10^{-10}	6.23×10^{-19}	7.20×10^{-6}	7.40×10^{-10}	5.81×10^{-19}
US06	0 °C	MLR-GA	4.06×10^2	7.90×10^{-2}	6.55×10^{-3}	4.23×10^2	4.68×10^{-2}	2.47×10^{-3}
		MLR-DE	5.24×10^1	1.02×10^{-2}	1.15×10^{-4}	5.78×10^1	6.40×10^{-3}	5.52×10^{-5}
		MLR-PSO	4.73×10^{-6}	9.21×10^{-10}	8.70×10^{-19}	5.37×10^{-6}	5.93×10^{-10}	3.74×10^{-19}
	25 °C	MLR-GA	1.33×10^2	2.58×10^{-2}	8.03×10^{-4}	1.11×10^2	1.22×10^{-2}	2.21×10^{-4}
		MLR-DE	9.08×10^0	1.76×10^{-3}	4.64×10^{-6}	1.73×10^1	1.90×10^{-3}	5.66×10^{-6}
		MLR-PSO	2.00×10^{-6}	3.87×10^{-10}	1.65×10^{-19}	1.31×10^{-6}	1.44×10^{-10}	3.10×10^{-20}
	45 °C	MLR-GA	2.86×10^2	5.59×10^{-2}	3.32×10^{-3}	5.13×10^2	5.66×10^{-2}	3.40×10^{-3}
		MLR-DE	1.37×10^1	2.67×10^{-3}	1.02×10^{-5}	3.29×10^1	3.62×10^{-3}	1.70×10^{-5}
		MLR-PSO	3.96×10^{-6}	7.74×10^{-10}	6.15×10^{-19}	6.44×10^{-6}	7.11×10^{-10}	5.31×10^{-19}
BJDST	0 °C	MLR-GA	3.00×10^2	5.53×10^{-2}	3.22×10^{-3}	7.29×10^2	7.73×10^{-2}	6.11×10^{-3}
		MLR-DE	4.25×10^1	7.83×10^{-3}	6.79×10^{-5}	6.66×10^1	7.06×10^{-3}	5.86×10^{-5}
		MLR-PSO	3.49×10^{-6}	6.43×10^{-10}	4.26×10^{-19}	9.26×10^{-6}	9.82×10^{-10}	9.81×10^{-19}
	25 °C	MLR-GA	1.57×10^2	2.93×10^{-2}	9.51×10^{-4}	2.95×10^2	3.10×10^{-2}	1.05×10^{-3}
		MLR-DE	7.92×10^0	1.48×10^{-3}	3.15×10^{-6}	2.22×10^1	2.34×10^{-3}	7.06×10^{-6}
		MLR-PSO	2.32×10^{-6}	4.33×10^{-10}	1.98×10^{-19}	3.53×10^{-6}	3.71×10^{-10}	1.60×10^{-19}
	45 °C	MLR-GA	3.07×10^2	5.74×10^{-2}	3.38×10^{-3}	5.51×10^2	5.84×10^{-2}	3.50×10^{-3}
		MLR-DE	1.58×10^1	2.95×10^{-3}	1.10×10^{-5}	3.81×10^1	4.03×10^{-3}	1.90×10^{-5}
		MLR-PSO	4.18×10^{-6}	7.81×10^{-10}	6.21×10^{-19}	6.70×10^{-6}	7.10×10^{-10}	5.25×10^{-19}

A final remark is a comparison with Zheng's work [81], who is the member of the research team that built the database. They compare two estimators based on the Kalman filter over three tests (FUDS, US06, and BJDST) in three different temperatures (0°, 25°, and 45°). They found the MAE evaluation parameter between 0.6% and 4.0% for the nine cases and an MAE close to 1% and 0.6% for the better cases (°C). Nonetheless, in our study, the MLR-PSO reached MAE values very close to zero for all the 24 tests. It is an introductory remark, since this work is a baseline for researchers in this field.

Table 4. Test results for all profiles at different SoC and temperatures for the SPL-MLR algorithm.

			50% SoC			80% SoC		
			AE	MAE	MSE	AE	MAE	MSE
DST	0 °C	SPL-MLR-GA	1.30×10^3	2.40×10^{-1}	1.60×10^{-1}	5.25×10^3	5.60×10^{-1}	8.70×10^{-1}
		SPL-MLR-DE	8.97×10^2	1.70×10^{-1}	7.00×10^{-2}	5.40×10^3	5.80×10^{-1}	1.20×10^0
		SPL-MLR-PSO	8.40×10^2	1.60×10^{-1}	1.00×10^{-1}	1.01×10^4	1.08×10^0	4.69×10^0
	25 °C	SPL-MLR-GA	1.28×10^3	2.40×10^{-1}	1.60×10^{-1}	5.61×10^3	6.00×10^{-1}	9.80×10^{-1}
		SPL-MLR-DE	7.09×10^2	1.30×10^{-1}	5.00×10^{-2}	5.61×10^3	5.90×10^{-1}	1.26×10^0
		SPL-MLR-PSO	2.22×10^2	4.00×10^{-2}	1.00×10^{-2}	1.04×10^4	1.11×10^0	4.89×10^0
	45 °C	SPL-MLR-GA	2.92×10^3	5.40×10^{-1}	8.10×10^{-1}	2.36×10^3	2.50×10^{-1}	1.80×10^{-1}
		SPL-MLR-DE	3.05×10^3	5.70×10^{-1}	1.20×10^0	1.29×10^3	1.40×10^{-1}	6.00×10^{-2}
		SPL-MLR-PSO	5.70×10^3	1.06×10^0	4.51×10^0	3.15×10^2	3.00×10^{-2}	1.00×10^{-2}
FUDS	0 °C	SPL-MLR-GA	1.40×10^3	2.50×10^{-1}	1.70×10^{-1}	2.59×10^3	2.70×10^{-1}	2.10×10^{-1}
		SPL-MLR-DE	9.31×10^2	1.60×10^{-1}	8.00×10^{-2}	1.69×10^3	1.80×10^{-1}	9.00×10^{-2}
		SPL-MLR-PSO	1.06×10^3	1.90×10^{-1}	1.40×10^{-1}	1.83×10^3	1.90×10^{-1}	1.60×10^{-1}
	25 °C	SPL-MLR-GA	2.42×10^3	4.30×10^{-1}	4.50×10^{-1}	2.54×10^3	2.60×10^{-1}	2.00×10^{-1}
		SPL-MLR-DE	2.08×10^3	3.70×10^{-1}	5.00×10^{-1}	1.41×10^3	1.40×10^{-1}	6.00×10^{-2}
		SPL-MLR-PSO	3.75×10^3	6.70×10^{-1}	1.78×10^0	8.47×10^2	9.00×10^{-2}	4.00×10^{-2}
	45 °C	SPL-MLR-GA	1.23×10^3	2.20×10^{-1}	1.50×10^{-1}	2.42×10^3	2.50×10^{-1}	1.80×10^{-1}
		SPL-MLR-DE	6.74×10^2	1.20×10^{-1}	4.00×10^{-2}	1.32×10^3	1.40×10^{-1}	5.00×10^{-2}
		SPL-MLR-PSO	1.42×10^2	3.00×10^{-2}	3.40×10^{-3}	4.75×10^2	5.00×10^{-2}	1.00×10^{-2}
US06	0 °C	SPL-MLR-GA	1.22×10^3	2.40×10^{-1}	1.40×10^{-1}	2.36×10^3	2.60×10^{-1}	1.80×10^{-1}
		SPL-MLR-DE	7.78×10^2	1.50×10^{-1}	6.00×10^{-2}	1.48×10^3	1.60×10^{-1}	7.00×10^{-2}
		SPL-MLR-PSO	6.14×10^2	1.20×10^{-1}	6.00×10^{-2}	8.79×10^2	1.00×10^{-1}	4.00×10^{-2}
	25 °C	SPL-MLR-GA	1.22×10^3	2.40×10^{-1}	1.40×10^{-1}	2.45×10^3	2.70×10^{-1}	1.90×10^{-1}
		SPL-MLR-DE	6.62×10^2	1.30×10^{-1}	4.00×10^{-2}	1.36×10^3	1.50×10^{-1}	6.00×10^{-2}
		SPL-MLR-PSO	7.05×10^1	1.36×10^{-2}	1.00×10^{-3}	4.13×10^2	5.00×10^{-2}	1.00×10^{-2}
	45 °C	SPL-MLR-GA	1.16×10^3	2.30×10^{-1}	1.30×10^{-1}	2.31×10^3	2.50×10^{-1}	1.70×10^{-1}
		SPL-MLR-DE	6.37×10^2	1.20×10^{-1}	4.00×10^{-2}	1.28×10^3	1.40×10^{-1}	6.00×10^{-2}
		SPL-MLR-PSO	1.43×10^2	2.80×10^{-2}	4.00×10^{-3}	2.16×10^2	2.28×10^{-2}	3.10×10^{-3}
BJDST	0 °C	SPL-MLR-GA	1.23×10^3	2.30×10^{-1}	1.30×10^{-1}	2.65×10^3	2.80×10^{-1}	2.10×10^{-1}
		SPL-MLR-DE	7.79×10^2	1.40×10^{-1}	5.00×10^{-2}	1.70×10^3	1.80×10^{-1}	8.00×10^{-2}
		SPL-MLR-PSO	3.77×10^2	7.00×10^{-2}	2.00×10^{-2}	9.23×10^2	1.00×10^{-1}	4.00×10^{-2}
	25 °C	SPL-MLR-GA	1.20×10^3	2.20×10^{-1}	1.20×10^{-1}	2.53×10^3	2.70×10^{-1}	1.80×10^{-1}
		SPL-MLR-DE	6.56×10^2	1.20×10^{-1}	4.00×10^{-2}	1.41×10^3	1.50×10^{-1}	6.00×10^{-2}
		SPL-MLR-PSO	8.91×10^1	1.66×10^{-2}	1.16×10^{-3}	1.30×10^2	1.37×10^{-2}	9.15×10^{-4}
	45 °C	SPL-MLR-GA	1.17×10^3	2.20×10^{-1}	1.20×10^{-1}	2.44×10^3	2.60×10^{-1}	1.70×10^{-1}
		SPL-MLR-DE	6.44×10^2	1.20×10^{-1}	4.00×10^{-2}	1.38×10^3	1.50×10^{-1}	6.00×10^{-2}
		SPL-MLR-PSO	2.47×10^2	5.00×10^{-2}	1.00×10^{-2}	4.02×10^2	4.25×10^{-2}	7.37×10^{-3}

Table 5. Test results for all profiles at different SoC and temperatures for the GLM algorithm.

			50% SoC			80% SoC		
			AE	MAE	MSE	AE	MAE	MSE
DST	0 °C	GLM-GA	4.00×10^2	7.00×10^{-2}	3.10×10^0	4.03×10^3	4.30×10^{-1}	1.92×10^1
		GLM-DE	3.96×10^2	7.00×10^{-2}	3.20×10^0	4.02×10^3	4.30×10^{-1}	1.87×10^1
		GLM-PSO	6.78×10^4	1.27×10^1	4.67×10^0	6.70×10^5	7.21×10^4	1.63×10^2
	25 °C	GLM-GA	1.71×10^2	3.00×10^{-2}	2.38×10^0	4.06×10^3	4.30×10^{-1}	2.06×10^1
		GLM-DE	1.73×10^2	3.00×10^{-2}	2.27×10^0	4.04×10^3	4.30×10^{-1}	1.96×10^1
		GLM-PSO	4.19×10^2	8.00×10^{-2}	2.09×10^0	3.14×10^5	3.33×10^4	1.77×10^2
	45 °C	GLM-GA	1.54×10^3	2.90×10^{-1}	2.15×10^1	4.95×10^2	5.00×10^{-2}	1.55×10^0
		GLM-DE	1.53×10^3	2.90×10^{-1}	2.07×10^1	7.22×10^2	8.00×10^{-2}	1.49×10^0
		GLM-PSO	4.00×10^5	7.46×10^4	1.66×10^2	3.62×10^3	3.80×10^{-1}	1.16×10^1
FUDS	0 °C	GLM-GA	4.73×10^2	8.00×10^{-2}	3.24×10^0	1.07×10^3	1.10×10^{-1}	2.56×10^0
		GLM-DE	4.68×10^2	8.00×10^{-2}	3.34×10^0	1.07×10^3	1.10×10^{-1}	2.65×10^0
		GLM-PSO	1.19×10^5	2.09×10^1	6.98×10^0	1.54×10^5	1.62×10^1	4.72×10^0
	25 °C	GLM-GA	1.38×10^3	2.50×10^{-1}	1.15×10^1	2.78×10^2	3.00×10^{-2}	1.89×10^0
		GLM-DE	1.36×10^3	2.40×10^{-1}	1.10×10^1	3.81×10^2	4.00×10^{-2}	1.83×10^0
		GLM-PSO	1.91×10^5	3.39×10^4	5.69×10^2	1.20×10^3	1.20×10^{-1}	2.73×10^0
	45 °C	GLM-GA	3.43×10^2	6.00×10^{-2}	1.90×10^0	6.78×10^2	7.00×10^{-2}	1.48×10^0
		GLM-DE	4.37×10^2	8.00×10^{-2}	1.81×10^0	8.96×10^2	9.00×10^{-2}	1.42×10^0
		GLM-PSO	1.47×10^3	2.60×10^{-1}	1.56×10^1	3.95×10^3	4.10×10^{-1}	1.53×10^1
US06	0 °C	GLM-GA	3.22×10^2	6.00×10^{-2}	2.82×10^0	5.52×10^2	6.00×10^{-2}	2.08×10^0
		GLM-DE	3.09×10^2	6.00×10^{-2}	2.90×10^0	5.25×10^2	6.00×10^{-2}	2.14×10^0
		GLM-PSO	2.35×10^4	4.58×10^0	1.52×10^0	1.51×10^4	1.67×10^0	2.60×10^{-1}
	25 °C	GLM-GA	1.52×10^2	3.00×10^{-2}	2.12×10^0	2.69×10^2	3.00×10^{-2}	1.89×10^0
		GLM-DE	2.41×10^2	5.00×10^{-2}	2.02×10^0	3.20×10^2	4.00×10^{-2}	1.80×10^0
		GLM-PSO	9.66×10^2	1.90×10^{-1}	6.57×10^0	8.22×10^2	9.00×10^{-2}	2.24×10^0
	45 °C	GLM-GA	3.07×10^2	6.00×10^{-2}	1.86×10^0	6.17×10^2	7.00×10^{-2}	1.47×10^0
		GLM-DE	3.97×10^2	8.00×10^{-2}	1.75×10^0	8.31×10^2	9.00×10^{-2}	1.39×10^0
		GLM-PSO	1.38×10^3	2.70×10^{-1}	1.50×10^1	3.67×10^3	4.00×10^{-1}	1.40×10^1
BJDST	0 °C	GLM-GA	2.83×10^2	5.00×10^{-2}	2.60×10^0	9.19×10^2	1.00×10^{-1}	2.39×10^0
		GLM-DE	2.62×10^2	5.00×10^{-2}	2.62×10^0	8.54×10^2	9.00×10^{-2}	2.38×10^0
		GLM-PSO	9.65×10^3	1.78×10^0	2.40×10^{-1}	9.50×10^4	1.01×10^1	3.15×10^0
	25 °C	GLM-GA	1.69×10^2	3.00×10^{-2}	2.10×10^0	2.74×10^2	3.00×10^{-2}	1.67×10^0
		GLM-DE	2.59×10^2	5.00×10^{-2}	1.96×10^0	4.98×10^2	5.00×10^{-2}	1.56×10^0
		GLM-PSO	1.05×10^3	2.00×10^{-1}	7.01×10^0	2.88×10^3	3.00×10^{-1}	6.29×10^0
	45 °C	GLM-GA	3.23×10^2	6.00×10^{-2}	1.86×10^0	6.39×10^2	7.00×10^{-2}	1.47×10^0
		GLM-DE	4.28×10^2	8.00×10^{-2}	1.72×10^0	8.92×10^2	9.00×10^{-2}	1.35×10^0
		GLM-PSO	1.43×10^3	2.70×10^{-1}	1.45×10^1	3.80×10^3	4.00×10^{-1}	1.34×10^1

6. Conclusions

Battery charge status prediction is a crucial step for the new generation of electric vehicles. However, the literature points out several approaches to accomplish this task, presenting high error and low reliability and computational techniques with high cost.

In this scenario, this study proposed a model to perform the SoC estimation in lithium-ion batteries for electric vehicles at low computational cost, simple implemen-

tation, and high assertiveness. It used the Multiple Linear Regression without (MLR) and with spline interpolation (SPL-MLR) and the Generalized Linear Model (GLM). It also applied three bio-inspired optimization metaheuristics to calculate the free coefficients of those models: Genetic Algorithm (GA), Differential Evolution (DE), and Particle Swarm Optimization (PSO).

The computational results obtained for both training and test sets indicate that the MLR-PSO achieved the best general results, being the best predictor for all case studies. It is also noteworthy that this technique performed better than the methods presented in the literature by the database authors. It is an important finding since the MLR-PSO model requires a relatively small computational effort, being a competitive candidate for real applications such as embedded systems in electric vehicles.

Finally, for future work, it is desired to carry out tests with different battery models and also in others conditions, with different C rates. For this, new databases are necessary. The existing noises on the measurement sensors also must be evaluated as must the ability of the computational models tested in this case. Still, it is required to consider the SoH in future studies to achieve a faithful representation of the SoC behavior in the long term. Regarding the computational intelligence models, the literature presents more than one hundred other algorithms and many variations of those used in this work. A further investigation should be developed in order to increase the knowledge about this topic.

Author Contributions: Conceptualization, F.C.C. and H.V.S.; methodology, D.C., F.C.C. and H.V.S.; software, D.C.; validation, J.E., L.S. and D.C.; formal analysis, D.C., F.C.C., H.V.S. and Y.d.S.T.; investigation, L.S.; resources, S.L.S.J. and D.C.; data curation, D.C., T.A.A. and L.S.; writing—original draft preparation, D.C., M.G. and J.E.; writing—review and editing, J.E., S.L.S.J., Y.d.S.T., H.V.S. and F.C.C.; visualization, H.V.S. and F.C.C.; supervision, F.C.C. All authors have read and agreed to the published version of the manuscript.

Funding: The authors thank the Federal University of Technology—Parana, University of Campinas (UNICAMP), and Brazilian agencies: Coordination for the Improvement of Higher Education Personnel (CAPES)—Financing Code 001, Brazilian National Council for Scientific and Technological Development (CNPq), processes number 40558/2018-5, 315298/2020-0, and Araucaria Foundation, process number 51497, and State of São Paulo Research Foundation (FAPESP) for their financial support.

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

References

1. Holjevac, N.; Cheli, F.; Gobbi, M. Multi-objective vehicle optimization: Comparison of combustion engine, hybrid and electric powertrains. *Proc. Inst. Mech. Eng. Part D J. Autom. Eng.* **2020**, *234*, 469–487. [\[CrossRef\]](#)
2. Eckert, J.J.; Barbosa, T.P.; Silva, F.L.; Roso, V.R.; Silva, L.C.; da Silva, L.A.R. Optimum fuzzy logic controller applied to a hybrid hydraulic vehicle to minimize fuel consumption and emissions. *Expert Syst. Appl.* **2022**, 117903. [\[CrossRef\]](#)
3. Eckert, J.J.; de Alkmin e Silva, L.C.; dos Santos Costa, E.; Santiciolli, F.M.; Corrêa, F.C.; Dedini, F.G. Optimization of electric propulsion system for a hybridized vehicle. *Mech. Based Des. Struct. Mach.* **2019**, *47*, 175–200. [\[CrossRef\]](#)
4. Corno, M.; Pozzato, G. Active adaptive battery aging management for electric vehicles. *IEEE Trans. Veh. Technol.* **2019**, *69*, 258–269. [\[CrossRef\]](#)
5. Li, Y.; Wang, S.; Duan, X.; Liu, S.; Liu, J.; Hu, S. Multi-objective energy management for Atkinson cycle engine and series hybrid electric vehicle based on evolutionary NSGA-II algorithm using digital twins. *Energy Convers. Manag.* **2021**, *230*, 113788. [\[CrossRef\]](#)
6. Xun, Q.; Liu, Y. Evaluation of fluctuating voltage topology with fuel cells and supercapacitors for automotive applications. *Int. J. Energy Res.* **2019**, *43*, 4807–4819. [\[CrossRef\]](#)
7. Hazra, S.; Reddy, J.K. An Aspect of Noise, Vibration, and Harshness Issues in Electric Vehicles. *SAE Int. J. Veh. Dyn. Stab. NVH* **2021**, *6*, 23–33. [\[CrossRef\]](#)
8. Padavala, P.; Inavolu, N.; Thaveedu, J.R.; Mediseti, J.R. Challenges in Noise Refinement of a Pure Electric Passenger Vehicle. *SAE Int. J. Veh. Dyn. Stab. NVH* **2021**, *5*, 45–64. [\[CrossRef\]](#)
9. Hu, J.; Niu, X.; Jiang, X.; Zu, G. Energy management strategy based on driving pattern recognition for a dual-motor battery electric vehicle. *Int. J. Energy Res.* **2019**, *43*, 3346–3364. [\[CrossRef\]](#)
10. Zha, Y.; Quan, X.; Ma, F.; Liu, G.; Zheng, X.; Yu, M. Stability Control for a Four-Wheel-Independent-Drive Electric Vehicle Based on Model Predictive Control. *SAE Int. J. Veh. Dyn. Stab. NVH* **2021**, *5*. [\[CrossRef\]](#)

11. Hazra, S.; Reddy, J.K. A Review Paper on Recent Research of Noise and Vibration in Electric Vehicle Powertrain Mounting System. *SAE Int. J. Veh. Dyn. Stab. NVH* **2021**, *6*, 3–22. [\[CrossRef\]](#)
12. Demircali, A.; Koroglu, S. Jaya algorithm-based energy management system for battery-and ultracapacitor-powered ultralight electric vehicle. *Int. J. Energy Res.* **2020**, *44*, 4977–4985. [\[CrossRef\]](#)
13. Barbosa, T.P.; Eckert, J.J.; Roso, V.R.; Pujatti, F.J.P.; da Silva, L.A.R.; Gutiérrez, J.C.H. Fuel saving and lower pollutants emissions using an ethanol-fueled engine in a hydraulic hybrid passengers vehicle. *Energy* **2021**, *235*, 121361. [\[CrossRef\]](#)
14. Feng, S.; Magee, C.L. Technological development of key domains in electric vehicles: Improvement rates, technology trajectories and key assignees. *Appl. Energy* **2020**, *260*, 114264. [\[CrossRef\]](#)
15. Li, Y.; Huang, X.; Liu, D.; Wang, M.; Xu, J. Hybrid energy storage system and energy distribution strategy for four-wheel independent-drive electric vehicles. *J. Clean. Prod.* **2019**, *220*, 756–770. [\[CrossRef\]](#)
16. Veneri, O.; Capasso, C.; Patalano, S. Experimental investigation into the effectiveness of a super-capacitor based hybrid energy storage system for urban commercial vehicles. *Appl. Energy* **2018**, *227*, 312–323. [\[CrossRef\]](#)
17. Sarvaiya, S.; Ganesh, S.; Xu, B. Comparative analysis of hybrid vehicle energy management strategies with optimization of fuel economy and battery life. *Energy* **2021**, *228*, 120604. [\[CrossRef\]](#)
18. Castaings, A.; Lhomme, W.; Trigui, R.; Bouscayrol, A. Energy management of a multi-source vehicle by λ -control. *Appl. Sci.* **2020**, *10*, 6541. [\[CrossRef\]](#)
19. da Silva, S.F.; Eckert, J.J.; Silva, F.L.; Silva, L.C.; Dedini, F.G. Multi-objective optimization design and control of plug-in hybrid electric vehicle powertrain for minimization of energy consumption, exhaust emissions and battery degradation. *Energy Convers. Manag.* **2021**, *234*, 113909. [\[CrossRef\]](#)
20. Eckert, J.J.; da Silva, S.F.; de Menezes Lourenço, M.A.; Corrêa, F.C.; Silva, L.C.; Dedini, F.G. Energy management and gear shifting control for a hybridized vehicle to minimize gas emissions, energy consumption and battery aging. *Energy Convers. Manag.* **2021**, *240*, 114222. [\[CrossRef\]](#)
21. Hannan, M.A.; Lipu, M.; Hussain, A.; Ker, P.J.; Mahlia, T.I.; Mansor, M.; Ayob, A.; Saad, M.H.; Dong, Z. Toward enhanced state of charge estimation of lithium-ion batteries using optimized machine learning techniques. *Sci. Rep.* **2020**, *10*, 4687. [\[CrossRef\]](#)
22. Remes, C.L.; Oliveira, S.V.G. Modelagem, Simulação e Estimação de Carga de Baterias de Lítio com Implementação de um Carregador de Baterias. Master's Thesis, Federal University of Parana, Curitiba, Brazil, 2016.
23. Bercibar, M.; Gandiaga, I.; Villarreal, I.; Omar, N.; Van Mierlo, J.; Van den Bossche, P. Critical review of state of health estimation methods of Li-ion batteries for real applications. *Renew. Sustain. Energy Rev.* **2016**, *56*, 572–587. [\[CrossRef\]](#)
24. Lipu, M.H.; Hannan, M.; Karim, T.F.; Hussain, A.; Saad, M.H.M.; Ayob, A.; Miah, M.S.; Mahlia, T.I. Intelligent algorithms and control strategies for battery management system in electric vehicles: Progress, challenges and future outlook. *J. Clean. Prod.* **2021**, *292*, 126044. [\[CrossRef\]](#)
25. Rahman, M.A.; Anwar, S.; Izadian, A. Electrochemical model parameter identification of a lithium-ion battery using particle swarm optimization method. *J. Power Sources* **2016**, *307*, 86–97. [\[CrossRef\]](#)
26. Wang, L.; Zhao, X.; Liu, L.; Pan, C. State of health estimation of battery modules via differential voltage analysis with local data symmetry method. *Electrochim. Acta* **2017**, *256*, 81–89. [\[CrossRef\]](#)
27. Miranda, M.H.; Silva, F.L.; Lourenço, M.A.; Eckert, J.J.; Silva, L.C. Electric vehicle powertrain and fuzzy controller optimization using a planar dynamics simulation based on a real-world driving cycle. *Energy* **2022**, *238*, 121979. [\[CrossRef\]](#)
28. Li, H.; Ravey, A.; N'Diaye, A.; Djerdir, A. Online adaptive equivalent consumption minimization strategy for fuel cell hybrid electric vehicle considering power sources degradation. *Energy Convers. Manag.* **2019**, *192*, 133–149. [\[CrossRef\]](#)
29. Omar, N.; Monem, M.A.; Firouz, Y.; Salminen, J.; Smekens, J.; Hegazy, O.; Gaulous, H.; Mulder, G.; Van den Bossche, P.; Coosemans, T.; et al. Lithium iron phosphate based battery-assessment of the aging parameters and development of cycle life model. *Appl. Energy* **2014**, *113*, 1575–1585. [\[CrossRef\]](#)
30. Wang, B.; Hu, Q.; Wang, Z. Improving Power Output of Battery and Mode Switching Frequency Based on Real-Time Average Power Method for Multi-Mode Hybrid Energy Storage System in Electric Vehicles. *IEEE Access* **2020**, *8*, 34654–34663. [\[CrossRef\]](#)
31. Bai, Y.; He, H.; Li, J.; Li, S.; Xiong Wang, Y.; Yang, Q. Battery anti-aging control for a plug-in hybrid electric vehicle with a hierarchical optimization energy management strategy. *J. Clean. Prod.* **2019**, *237*, 117841. [\[CrossRef\]](#)
32. Li, J.; Greye, B.; Buchholz, M.; Danzer, M.A. Interval method for an efficient state of charge and capacity estimation of multicell batteries. *J. Energy Storage* **2017**, *13*, 1–9. [\[CrossRef\]](#)
33. Li, G.; Zhang, J.; He, H. Battery SOC constraint comparison for predictive energy management of plug-in hybrid electric bus. *Appl. Energy* **2017**, *194*, 578–587. [\[CrossRef\]](#)
34. Eckert, J.J.; Silva, L.C.A.; Santicioli, F.M.; Costa, E.S.; Corrêa, F.C.; Dedini, F.G. Energy storage and control optimization for an electric vehicle. *Int. J. Energy Res.* **2018**, *44*, 3506–3523. [\[CrossRef\]](#)
35. Eckert, J.J.; Silva, L.C.; Dedini, F.G.; Correa, F.C. Electric Vehicle Powertrain and Fuzzy Control Multi-objective Optimization, considering Dual Hybrid Energy Storage Systems. *IEEE Trans. Veh. Technol.* **2020**, *69*, 3773–3782. [\[CrossRef\]](#)
36. Hou, H.; Xue, M.; Xu, Y.; Xiao, Z.; Deng, X.; Xu, T.; Liu, P.; Cui, R. Multi-objective economic dispatch of a microgrid considering electric vehicle and transferable load. *Appl. Energy* **2020**, *262*, 114489. [\[CrossRef\]](#)
37. Xiong, R.; Cao, J.; Yu, Q.; He, H.; Sun, F. Critical review on the battery state of charge estimation methods for electric vehicles. *IEEE Access* **2017**, *6*, 1832–1843. [\[CrossRef\]](#)

38. Lipu, M.H.; Hannan, M.; Hussain, A.; Ayob, A.; Saad, M.H.; Karim, T.F.; How, D.N. Data-driven state of charge estimation of lithium-ion batteries: Algorithms, implementation factors, limitations and future trends. *J. Clean. Prod.* **2020**, *277*, 124110. [\[CrossRef\]](#)
39. Shepherd, C.M. Design of primary and secondary cells: II. An equation describing battery discharge. *J. Electrochem. Soc.* **1965**, *112*, 657. [\[CrossRef\]](#)
40. Saw, L.; Somasundaram, K.; Ye, Y.; Tay, A. Electro-thermal analysis of Lithium Iron Phosphate battery for electric vehicles. *J. Power Sources* **2014**, *249*, 231–238. [\[CrossRef\]](#)
41. Eckert, J.J.; Barbosa, T.P.; da Silva, S.F.; Silva, F.L.; Silva, L.C.; Dedini, F.G. Electric hydraulic hybrid vehicle powertrain design and optimization-based power distribution control to extend driving range and battery life cycle. *Energy Convers. Manag.* **2022**, *252*, 115094. [\[CrossRef\]](#)
42. da Silva, S.F.; Eckert, J.J.; Corrêa, F.C.; Silva, F.L.; Silva, L.C.; Dedini, F.G. Dual HESS electric vehicle powertrain design and fuzzy control based on multi-objective optimization to increase driving range and battery life cycle. *Appl. Energy* **2022**, *324*, 119723. [\[CrossRef\]](#)
43. Delfino, F.; Ferro, G.; Minciardi, R.; Robba, M.; Rossi, M.; Rossi, M. Identification and optimal control of an electrical storage system for microgrids with renewables. *Sustain. Energy Grids Netw.* **2019**, *17*, 100183. [\[CrossRef\]](#)
44. Hannan, M.; Lipu, M.; Hussain, A.; Mohamed, A. A review of lithium-ion battery state of charge estimation and management system in electric vehicle applications: Challenges and recommendations. *Renew. Sustain. Energy Rev.* **2017**, *78*, 834–854. [\[CrossRef\]](#)
45. Zhang, C.W.; Chen, S.R.; Gao, H.B.; Xu, K.J.; Yang, M.Y. State of charge estimation of power battery using improved back propagation neural network. *Batteries* **2018**, *4*, 69. [\[CrossRef\]](#)
46. Ali, M.U.; Zafar, A.; Nengroo, S.H.; Hussain, S.; Junaid Alvi, M.; Kim, H.J. Towards a smarter battery management system for electric vehicle applications: A critical review of lithium-ion battery state of charge estimation. *Energies* **2019**, *12*, 446. [\[CrossRef\]](#)
47. Vidal, C.; Malysz, P.; Kollmeyer, P.; Emadi, A. Machine learning applied to electrified vehicle battery state of charge and state of health estimation: State-of-the-art. *IEEE Access* **2020**, *8*, 52796–52814. [\[CrossRef\]](#)
48. Lu, C.; Tao, L.; Fan, H. Li-ion battery capacity estimation: A geometrical approach. *J. Power Sources* **2014**, *261*, 141–147. [\[CrossRef\]](#)
49. Yang, D.; Zhang, X.; Pan, R.; Wang, Y.; Chen, Z. A novel Gaussian process regression model for state-of-health estimation of lithium-ion battery using charging curve. *J. Power Sources* **2018**, *384*, 387–395. [\[CrossRef\]](#)
50. Li, Y.; Zou, C.; Berecibar, M.; Nanini-Maury, E.; Chan, J.C.W.; van den Bossche, P.; Van Mierlo, J.; Omar, N. Random forest regression for online capacity estimation of lithium-ion batteries. *Appl. Energy* **2018**, *232*, 197–210. [\[CrossRef\]](#)
51. Stroe, D.I.; Schartz, E. Lithium-Ion Battery State-of-Health Estimation Using the Incremental Capacity Analysis Technique. *IEEE Trans. Ind. Appl.* **2020**, *56*, 678–685. [\[CrossRef\]](#)
52. Li, Y.; Sheng, H.; Cheng, Y.; Stroe, D.I.; Teodorescu, R. State-of-health estimation of lithium-ion batteries based on semi-supervised transfer component analysis. *Appl. Energy* **2020**, *277*, 115504. [\[CrossRef\]](#)
53. Shu, X.; Li, G.; Zhang, Y.; Shen, J.; Chen, Z.; Liu, Y. Online diagnosis of state of health for lithium-ion batteries based on short-term charging profiles. *J. Power Sources* **2020**, *471*, 228478. [\[CrossRef\]](#)
54. Li, Y.; Li, K.; Liu, X.; Wang, Y.; Zhang, L. Lithium-ion battery capacity estimation—A pruned convolutional neural network approach assisted with transfer learning. *Appl. Energy* **2021**, *285*, 116410. [\[CrossRef\]](#)
55. Li, Y.; Li, K.; Liu, X.; Zhang, L. Fast battery capacity estimation using convolutional neural networks. *Trans. Inst. Meas. Control* **2020**, 0142331220966425. [\[CrossRef\]](#)
56. Cheng, Y.; Song, D.; Wang, Z.; Lu, C.; Zerhouni, N. An ensemble prognostic method for lithium-ion battery capacity estimation based on time-varying weight allocation. *Appl. Energy* **2020**, *266*, 114817. [\[CrossRef\]](#)
57. Li, X.; Yuan, C.; Li, X.; Wang, Z. State of health estimation for Li-Ion battery using incremental capacity analysis and Gaussian process regression. *Energy* **2020**, *190*, 116467. [\[CrossRef\]](#)
58. Shen, S.; Sadoughi, M.; Chen, X.; Hong, M.; Hu, C. A deep learning method for online capacity estimation of lithium-ion batteries. *J. Energy Storage* **2019**, *25*, 100817. [\[CrossRef\]](#)
59. Zhao, D.; Zhou, Z.; Tang, S.; Cao, Y.; Wang, J.; Zhang, P.; Zhang, Y. Online estimation of satellite lithium-ion battery capacity based on approximate belief rule base and hidden Markov model. *Energy* **2022**, *256*, 124632. [\[CrossRef\]](#)
60. Hu, Y.; Yurkovich, S.; Guezennec, Y.; Yurkovich, B. Electro-thermal battery model identification for automotive applications. *J. Power Sources* **2011**, *196*, 449–457. [\[CrossRef\]](#)
61. Cervelló-Royo, R.; Guijarro, F. Forecasting stock market trend: A comparison of machine learning algorithms. *Financ. Mark. Valuat.* **2020**, *6*, 37–49. [\[CrossRef\]](#)
62. Álvarez Antón, J.C.; García Nieto, P.J.; García Gonzalo, E.; Viera Pérez, J.C.; González Vega, M.; Blanco Viejo, C. A New Predictive Model for the State-of-Charge of a High-Power Lithium-Ion Cell Based on a PSO-Optimized Multivariate Adaptive Regression Spline Approach. *IEEE Trans. Veh. Technol.* **2016**, *65*, 4197–4208. [\[CrossRef\]](#)
63. Belotti, J.T.; Castanho, D.S.; Araujo, L.N.; da Silva, L.V.; Alves, T.A.; Tadano, Y.S.; Stevan, S.L., Jr.; Correa, F.C.; Siqueira, H.V. Air pollution epidemiology: A simplified Generalized Linear Model approach optimized by bio-inspired metaheuristics. *Environ. Res.* **2020**, *191*, 110106. [\[CrossRef\]](#) [\[PubMed\]](#)
64. Ahmadzadeh, E.; Lee, J.; Moon, I. Optimized Neural Network Weights and Biases Using Particle Swarm Optimization Algorithm for Prediction Applications. *J. Korea Multimed. Soc.* **2017**, *20*, 1406–1420.

65. Santos, P.; Macedo, M.; Figueiredo, E.; Santana, C.J.; Soares, F.; Siqueira, H.; Maciel, A.; Gokhale, A.; Bastos-Filho, C.J. Application of PSO-based clustering algorithms on educational databases. In Proceedings of the 2017 IEEE Latin American Conference on Computational Intelligence (LA-CCI), Arequipa, Peru, 8–10 November 2017; pp. 1–6.
66. Puchta, E.D.P.; Siqueira, H.V.; dos Santos Kaster, M. Optimization tools based on metaheuristics for performance enhancement in a Gaussian adaptive PID controller. *IEEE Trans. Cybern.* **2019**, *50*, 1185–1194. [\[CrossRef\]](#)
67. Siqueira, H.; Macedo, M.; de Souza Tadano, Y.; Alves, T.A.; Stevan, S.L., Jr.; Oliveira, D.S., Jr.; Marinho, M.H.; de Mattos Neto, P.S.; de Oliveira, J.F.; Luna, I.; et al. Selection of temporal lags for predicting riverflow series from hydroelectric plants using variable selection methods. *Energies* **2020**, *13*, 4236. [\[CrossRef\]](#)
68. Campos, D.S.; de Souza Tadano, Y.; Alves, T.A.; Siqueira, H.V.; de Nóbrega Marinho, M.H. Unorganized machines and linear multivariate regression model applied to atmospheric pollutant forecasting. *Acta Scientiarum. Technol.* **2020**, *42*, e48203. [\[CrossRef\]](#)
69. Lazzarin, L.A. Método ensemble baseado em redes neurais artificiais para estimação de internações por doenças respiratórias. Master's Thesis, Universidade Tecnológica Federal do Paraná, Curitiba, Brazil, 2019.
70. Neuhaus, J.; McCulloch, C. Generalized linear models. *Wiley Interdiscip. Rev. Comput. Stat.* **2011**, *3*, 407–413. [\[CrossRef\]](#)
71. Holland, J.H. Genetic Algorithms. *Sci. Am.* **1992**, *267*, 66–73. [\[CrossRef\]](#)
72. Figueiredo, E.; Macedo, M.; Siqueira, H.V.; Santana, C.J., Jr.; Gokhale, A.; Bastos-Filho, C.J. Swarm intelligence for clustering—A systematic review with new perspectives on data mining. *Eng. Appl. Artif. Intell.* **2019**, *82*, 313–329. [\[CrossRef\]](#)
73. Puchta, E.D.; Lucas, R.; Ferreira, F.R.; Siqueira, H.V.; Kaster, M.S. Gaussian adaptive PID control optimized via genetic algorithm applied to a step-down DC-DC converter. In Proceedings of the 2016 12th IEEE International Conference on Industry Applications (INDUSCON), Curitiba, PR, Brazil, 20–23 November 2016; pp. 1–6. [\[CrossRef\]](#)
74. Michalewicz, Z. Evolution Strategies and Other Methods. In *Genetic Algorithms + Data Structures = Evolution Programs*; Michalewicz, Z., Ed.; Springer: Berlin/Heidelberg, Germany, 1996; pp. 159–177.
75. De Castro, L.N. *Fundamentals of Natural Computing*; Chapman and Hall/CRC: Boca Raton, FL, USA, 2006; p. 696.
76. Goldberg, D.E. *The Design of Innovation: Lessons from and for Competent Genetic Algorithms*; Springer: Berlin/Heidelberg, Germany, 2002; Volume 1.
77. Storn, R.; Price, K. Differential evolution a simple and efficient heuristic for global optimization over continuous spaces. *J. Glob. Optim.* **1997**, *11*, 341–359. [\[CrossRef\]](#)
78. Eberhart, R.; Kennedy, J. Particle swarm optimization. In Proceedings of the IEEE International Conference on Neural Networks, Perth, Australia, 27 November–1 December 1995; Volume 4, pp. 1942–1948. [\[CrossRef\]](#)
79. Clerc, M.; Kennedy, J. The particle swarm-explosion, stability, and convergence in a multidimensional complex space. *IEEE Trans. Evol. Comput.* **2002**, *6*, 58–73. [\[CrossRef\]](#)
80. Pecht, M. CALCE—Center for Advanced Life Cycle Engineering—Battery Research Group—University of Maryland. Available online: <https://web.calce.umd.edu/batteries/data.htm> (accessed on 4 July 2021).
81. Zheng, F.; Xing, Y.; Jiang, J.; Sun, B.; Kim, J.; Pecht, M. Influence of different open circuit voltage tests on state of charge online estimation for lithium-ion batteries. *Appl. Energy* **2016**, *183*, 513–525. [\[CrossRef\]](#)
82. *USABC Electric Vehicle Battery Test Procedures Manual. Revision 2*; Lockheed Idaho Technologies Co.: Idaho Falls, ID, USA, 1996. [\[CrossRef\]](#)
83. Hunt, G. *USABC Electric Vehicle Battery Test Procedures Manual*; United States Department of Energy: Washington, DC, USA, 1996.
84. Berry, I.M. The Effects of Driving Style and Vehicle Performance on the Real-World Fuel Consumption of US Light-Duty Vehicles. Ph.D. Thesis, Massachusetts Institute of Technology, Cambridge, MA, USA, 2010.
85. Box, G.E.; Jenkins, G.M.; Reinsel, G.C.; Ljung, G.M. *Time Series Analysis: Forecasting and Control*; John Wiley & Sons: Hoboken, NJ, USA, 2015.
86. Eberhart, R.C.; Shi, Y. Comparison between genetic algorithms and particle swarm optimization. In *International Conference on Evolutionary Programming*; Springer: Berlin/Heidelberg, Germany, 1998; pp. 611–616.