

Original articles

# Robust parameter estimation of a PEMFC via optimization based on probabilistic model building

Luis Blanco-Cocom<sup>a</sup>, Salvador Botello-Rionda<sup>a</sup>, L.C. Ordoñez<sup>b</sup>, S. Ivvan Valdez<sup>c,\*</sup>

<sup>a</sup> Centro de Investigación en Matemáticas, A.C., Jalisco S/N, Col. Valenciana CP: 36023, Guanajuato, Gto, México, Apartado Postal 402, CP 36000, Mexico

<sup>b</sup> Unidad de Energía Renovable, Centro de Investigación Científica de Yucatán. Parque Científico Tecnológico de Yucatán, Mérida, Yucatán, CP 97302, Mexico

<sup>c</sup> Cátedra-CONACYT Research Fellow, Centro de Investigación en Ciencias de Información Geoespacial, CENTROGEO, A.C., Contoy 137, Col. Lomas de Padierna, Delegación Tlalpan, CP. 14240, CDMX, Mexico

Received 2 January 2020; received in revised form 18 November 2020; accepted 20 December 2020

Available online 31 December 2020

## Abstract

In this work, we approximated a set of unknown physical parameters for a semi-empirical mathematical model of a PEMFC. We used an Estimation of Distribution Algorithm (EDA) known as  $UMDA^G$  to find the tuple that best reproduces the experimental polarization curve. We tackled non-derivable objective functions to perform robust parameter estimation. We compared the sum of the squared error with published results, and the sum and the median of the absolute error values were used to diminish or remove the effect of possible noise or outliers. Since the  $UMDA^G$  requires a single user-given parameter (the population size) and presents a natural reduction of the variance, it was possible to introduce a variance-based stopping criterion. The obtained results were compared with the most up-to-date evolutionary algorithms, demonstrating that this proposal is competitive. We used four previously reported experimental datasets to get the parameters or validate them. Two of them were used to test the method and to compare it with reported results of recent bio-inspired metaheuristics. Then, we used the identified parameters to simulate the cases of the remaining data sets validating the correct estimation. Finally, we introduced a posterior statistical analysis (hypothesis test), which provided further information about dependencies and the impact of each parameter on the cell performance.

© 2020 International Association for Mathematics and Computers in Simulation (IMACS). Published by Elsevier B.V. All rights reserved.

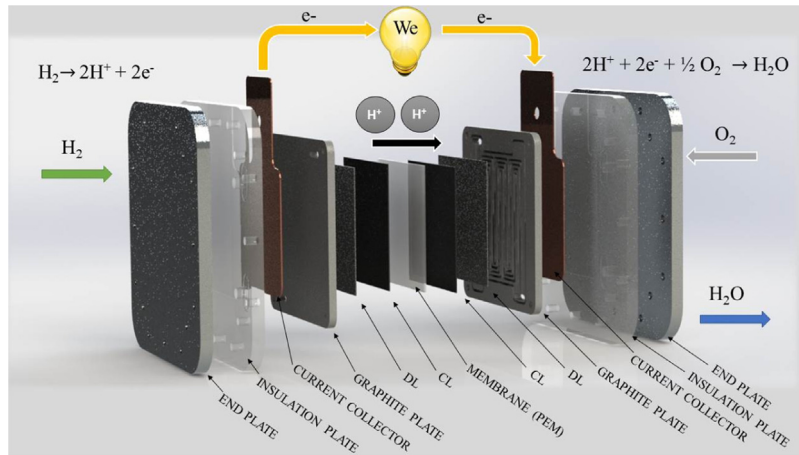
**Keywords:** Parameter estimation;  $UMDA^G$ ; PEMFC; Polarization curve; Probabilistic model building optimization; Semiempirical model

## 1. Introduction

A proton exchange membrane Fuel cell (PEMFC) is an electrochemical system that converts the chemical energy of hydrogen and oxygen directly into electricity by electrocatalytic reactions. Fig. 1 shows the basic structure of a PEMFC. It consists of an ionomer membrane in contact on either side with a catalytic layer, a diffusion layer, and a terminal plate with flow channels for the reactants. At the anode, hydrogen flows into the flow channels, and it is transported through the diffusion layer (DL) to the catalytic layer (CL) usually conformed by Pt/C catalyst. At Pt

\* Corresponding author.

E-mail address: [sergio.valdez@conacyt.mx](mailto:sergio.valdez@conacyt.mx) (S.I. Valdez).



**Fig. 1.** Basic structure of a proton exchange membrane fuel cell.

catalytic active sites,  $H_2$  is oxidized to produce protons and electrons. The former moves to the cathodic catalytic layer through the membrane, while the later moves to the cathode current collector via the external electric circuit. Meanwhile, oxygen flowing in the cathodic flow channels diffuses through the DL to the cathodic CL, usually based on Pt/C or PtRu/C catalysts, where it is reduced with the electrons and protons to produce water and heat (Reactions (1)–(3)) [40].

Anode reaction:



Cathode reaction:



Global reaction:



Mathematical models are useful for the study of a PEMFC performance and for improving its design. There are different studies related to the modeling and simulation of a PEMFC [26,40,46,48]. Nevertheless, every mathematical model requires a set of parameters that are difficult to obtain experimentally, and that needs to be estimated.

The estimation of parameters is posed as an optimization problem, in which the objective function is usually defined as an error minimization of the model output for fitting experimental data [16,39,41,56]. A recent review in [53], shows the most common objective functions used in estimating PEMFC parameters. A candidate solution to this estimation problem is a set of parameters. A derivative-based method, usually, models the problem as the sum of squared errors, to use a derivable smooth function. Nevertheless, estimation of distribution algorithms (EDAs) deals adequately with non-derivable functions such as the one approached in this research. Regarding the error function, a small perturbation of a solution in the parameter space leads to an abrupt change in the model output, due to the non-linearity of the optimization problem, additionally, it could present multiple minimums and non-smoothness. Considering the EDAs niche of application, we propose to minimize the sum of squared errors for comparing with previous studies, the sum of absolute error values, and the median for reducing or eliminating the effect of outliers.

Several numerical algorithms have been used for solving the parameter estimation problem in PEMFC, such as simulated annealing [40,41], genetic algorithm (GA) [35,39], RNA-genetic algorithm [56], particle swarm optimization (PSO) [47,54], artificial bee swarm in [3], harmonic search in [4], artificial immune system in [5], and differential evolution algorithm [10,50].

In [39], a GA is used for estimating the parameters of a fuel cell, this approach found adequate parameters; however, the author uses arbitrary execution parameters, and solutions from different executions present a large statistical variance. Thus, the confidence of the results is more related to the expertise of the final decision maker

than to the method for estimating the cell parameters. It is desirable that the GA returns similar results in several executions and that the GA stops when the best solution is found. These drawbacks are circumvented in [56], via new operators for the RNA-GA that improve the results of previous proposals. They also introduced a stopping criterion that automatically stopped the search when no further improvement is detected. However, the new operators make obscure the search process as well as the parameter settings for this algorithm. The Hybrid Artificial Bee Colony algorithm (HABC) [57] is one of the most competitive approaches for this problem, and it has an understandable behavior based on bees paths. However, the parameters settings are not intuitive, and the stopping criterion is arbitrary or depends on the expert criteria.

The covariance matrix adaptation evolution strategy (*CMAES*), belongs to derivative-free and stochastic methods for numerical optimization of non-linear or non-convex continuous optimization problems. The algorithm bases its heuristics on the iterative adaptation of the complete covariance matrix of a normal distribution of mutations, decomposes the total covariance matrix into a covariance matrix *C* and the global variance, called the step size control [19–21]. The *CMAES* learns all pair-wise dependencies between all parameters by updating a covariance matrix for the sample distribution [44].

Most of the mentioned algorithms lack a convergence criterion; hence, an expert must set the number of iterations and other parameters to ensure successful executions. Finally, the mentioned stochastic algorithms do not ensure a local optimum nor bounded computation time.

In the recent literature, modifications to swarm-intelligence and bio-inspired optimization algorithms have been introduced in order to improve their performance to estimate adequate parameters of a PEMFC, for instance, the Cuckoo Search Algorithm [9] is inspired in the Cuckoo parasite bird, uses a Levy random flight to evolve a set of nets and eggs, it applied to the optimization of parameters of the polarization curve obtained from [35]. Shark Smell Optimizer [43] is inspired in the shark predatory capacity based on following the odor of prey, it is devoted to engineering applications since its introduction when it was applied to a control problem. This algorithm is used for optimizing parameters of the polarization curves from the Ballard V 5 kW, and the SR-12 PEM 500 W. Teaching–Learning Based Optimization with Differential Evolution (TLBO-DE) [17], in the same vein than the Shark Smell algorithm, Teaching–Learning base Optimization Algorithm was proposed for improving a mechanical design based on emulating a teaching experience, in [50], it is combined with the widely-used differential evolution, the authors used this algorithm to set the parameters of the experimental polarization curve obtained from the 250 MW PEMFC. An adaptation of the Cuckoo Search Algorithm with Onlooker Bee Search (ObCS) is proposed in [59] for tuning the parameters for the nonlinear modeling approach of a PEMFC using type-2 fuzzy neural network (T2FNN), the objective function was the minimum of the mean of the square sum error. The Improve Fluid Search Optimization Algorithm (IFSO) [42], simulates Bernoulli's principle in fluid mechanics and is applied to three practical case studies including Horizon H-12 stacks, NedStack PS6, and Ballard Mark V 5 kW under different operating conditions with temperature variations between 30 °C and 55 °C and pressure variations between 1.0/1.0 Bar and 3.0/3.0 Bar. The Chaotic Binary Shark Smell Optimization is proposed in [18] as a variant of the Shark Smell Optimizer, this algorithm emulates the ability of the sharks in finding their preys. The binary operator permits to obtain the best response of the individuals for levels of forwarding movement, rotational movement, and chaotic improvement. To validate the algorithm the authors used the experimental data from a 250 MW PEMFC stack [35], the objective function is given by the minimum of the SSE. A version of the Coyote Optimization Algorithm was proposed in [55], the algorithm is based on the adaptation of the behavior of the coyote to the environment and also the coyote's experiences exchanging, it has a balance between exploration and exploitation, the objective function was the sum of the absolute error between the experimental data and simulations. For validating this algorithm the authors used two different PEMFC models: (a) 2 kW Nexa FC, and (b) 6 kW NedSstack PS6 FC. In [22], algorithms based on social behavior (Shuffled frog-leaping algorithm, Imperialist competitive algorithm and Firefly optimization algorithm) are applied to estimate parameters using the fitness function based on minimize the sum of squared errors (SSE) between the output voltage of each PEMFC stack and the estimated voltage by the semi-empirical model in three set of experimental data obtained from: (1) NedSstack PS6 PEMFC stack with the rated power of 6 kW, (2) BCS 500-W PEMFC stack, and (3) open cathode 500-W Horizon PEMFC.

The Harris Hawks Optimization (HHO) is presented in [2], this algorithm is a gradient-free optimization technique based on the nature of Harris' hawks, they can track and detect the prey by their eyes, occasionally the prey cannot be seen easily, then, the hawks wait, observe, and monitor the desert site to detect prey. This technique uses the idea of exploration (nature of the Harris' hawks) and exploitation (surprise pounce). The Chaotic Harris

Hawks Optimization (CHHO) is presented in [32], this modification uses the HHO with chaos maps to foretell actions by formulating a set of chaotic equations, and was applied to estimate parameters of the semi-empirical model, minimizing the square sum of the errors of the simulation respect to the data obtained from four different stacks: (a) 250 W stack, (b) BCS 500 W, (c) SR-12 PEM 500 W, and Temasek 1 kW.

In [52] a mathematics-based algorithm named JAYA-NM is proposed to estimate parameters of a semi-empirical mathematical model of a PEMFC, this algorithm consists of two stages: the coarse global search stage (JAYA) and the intensive local search stage (Nelder–Mead simplex search algorithm). The reported results were compared with other bio-inspired algorithms as Grasshopper Optimization Algorithm, GOA [45], and Salp Swarm Optimizer, SSO, [33]. This last set of experiments are the most recent and most competitive reported in the specialized literature, in the best of our knowledge, hence, they are used in this article for comparing our proposal.

The novelty of the paper is the application of estimation of distribution algorithms to the PEMFC parameter estimation, the comparison of different evolutionary and swarm intelligence algorithms, the use of a variance-based stopping criterion in order to avoid an arbitrary stopping criterion, such as number of generations or iterations, that are usual in applications of this kind of algorithms, the introduction of two objective functions that automatically diminish (the sum of absolute values) and remove (the median) the effect of the outliers. The sum of squared errors is the most widely used function and it is adequate when you have data that fits exactly the theoretical model, nevertheless, that is not the case for experimental data, noisy data is very common, hence one of the benefits of using derivative-free algorithms is the potential of using multimodal non-derivable objective functions, we take advantage of this characteristic, the results show that the sum of absolute values and the median reports similar estimated parameters than the sum of square errors, hence we validate that such functions in combination with the studied algorithms can successfully solve the problem.

## 2. The semi-empirical mathematical model

The semi-empirical mathematical model describes the characteristic polarization curve of a PEMFC. This model involves several parameters, which describe the overpotential in the anode and the cathode as a function of oxygen and hydrogen partial pressures [7,31]. The cell output voltage VFC is defined by Eq. (4).

$$V_{FC} = E_{Nernst} - \eta_{act} - \eta_{Ohmic} - \eta_{con}, \quad (4)$$

where,  $E_{Nernst}$  is the thermodynamic potential of the global reaction,  $\eta_{act}$ ,  $\eta_{Ohmic}$  and  $\eta_{con}$  are the activation, ohmic and concentration voltage drops, respectively. At lower currents, the main voltage drop is due to the activation losses. At higher currents, the voltage drop is caused by the concentration losses or resistance to the mass transfer processes; and at intermediate currents, the voltage drop occurs due to the ohmic losses [50]. Each term of Eq. (4) is described in Eqs. (5), (6), (8) and (11), according to [31]:

$$E_{Nernst} = 1.229 - 0.85 \times 10^{-3}(T - 298.15) + 4.308 \times 10^{-5}T \left[ \ln(P_{H_2}) + \frac{1}{2} \ln(P_{O_2}) \right], \quad (5)$$

where  $T$  represents the cell temperature (K),  $P_{H_2}$  and  $P_{O_2}$  are the partial pressures (atm) of hydrogen and oxygen, respectively (see Table 1).

The activation overpotential,  $\eta_{act}$ , occurs due to reactions on the active surface of the anode and cathode sides and it is defined by Eq. (6).

$$\eta_{act} = -[\xi_1 + \xi_2 T + \xi_3 T \ln(C_{O_2}) + \xi_4 T \ln(i)], \quad (6)$$

where,  $i$  is the cell current (A),  $\xi_i$  ( $i = 1, \dots, 4$ ), are the semi-empirical coefficients, [11], the oxygen concentration in the cathode interface,  $C_{O_2}$  (mol cm<sup>-3</sup>), it is defined in Eq. (7).

$$C_{O_2} = \frac{P_{O_2}}{5.08 \times 10^6 \times e^{-(498/T)}}, \quad (7)$$

The ohmic loss  $\eta_{Ohmic}$  results from the sum of the resistance of the proton conduction through the membrane and the resistance of electrons to the external electrical circuit, Eq. (8).

$$\eta_{Ohmic} = i(R_M + R_C), \quad (8)$$

**Table 1**

List of important variables of the semi-empirical mathematical model.

Symbol	Variable
$E_{Nernts}$	Open circuit voltage (V)
$\eta_{act}$	Activation voltage drop (V)
$\eta_{Ohmic}$	Ohmic voltage drop (V)
$\eta_{con}$	Concentration voltage drop (V)
$T$	Temperature (°C)
$P_{H_2}$	Partial pressure of hydrogen (atm)
$P_{O_2}$	Partial pressure of oxygen (atm)
$C_{O_2}$	Oxygen concentration (mol cm <sup>-3</sup> )
$V_{FC}$	Stack voltage (V)
$i$	Cell current (A)
$R_M$	Equivalent membrane resistance to proton conduction ( $\Omega$ )
$R_C$	Equivalent contact resistance to electron conduction ( $\Omega$ )
$\rho_M$	Specific membrane resistivity ( $\Omega$ cm).
$L_c$	Membrane thickness (cm)
$A$	Active cell area (cm <sup>2</sup> )
$J$	Current density (A cm <sup>2</sup> )
$J_{max}$	Maximum current density (A cm <sup>2</sup> )
$RH_c$	Relative humidity of vapor in the cathode
$RH_a$	Relative humidity of vapor in the anode
$\psi$	Parameter related to the water content of the membrane
$B$	Parametric coefficient
$n$	Number of cells connected in series
$P_{H_2O}^*$	Saturation pressure of water vapor (atm)
$P_{N_2}$	Nitrogen partial pressure at the cathode flow channel (atm)
$P_a$	Inlet pressures at the anode (atm)
$P_c$	Inlet pressures at the cathode (atm)
$\xi_i$	Semi-empirical coefficient ( $i = 1, \dots, 4$ )

where,  $R_M$  ( $\Omega$ ) is the equivalent membrane impedance, and  $R_C$  ( $\Omega$ ) is the contact resistances both between the membrane and electrodes as well as the electrodes and the bipolar plates.  $R_M$  is described by Eq. (9).

$$R_M = \frac{\rho_M \times L_c}{A}, \quad (9)$$

$$\rho_M = \frac{181.6[1 + 0.03 \cdot (i/A) + 0.062 \cdot (T/303)^2 \cdot (i/A)^2]}{[\psi - 0.634 - 3 \cdot (i/A)] \cdot \exp[4.18 \cdot (T - 303/T)]}, \quad (10)$$

where  $A$  is the active cell area (cm<sup>2</sup>),  $L_c$  is the membrane thickness (cm) and  $\rho_M$  ( $\Omega$  cm) is the specific membrane resistivity.  $\rho_M$  is described by Eq. (10) and  $\psi$  is the parameter related to the water content of the membrane which ranges between 13 and 24 [50]. This work considers the Nafion<sup>®</sup> membrane from Dupont, specifically, Nafion 115 ( $L_c = 127$   $\mu$ m).

The concentration overpotential,  $\eta_{con}$  is defined in Eq. (11).

$$\eta_{con} = -B \cdot \left(1 - \frac{J}{J_{max}}\right), \quad (11)$$

where  $B$  is the parametric coefficient,  $J$  is the current density (A cm<sup>-2</sup>),  $J_{max}$  is the maximum current density (A cm<sup>-2</sup>). The concentration voltage drop also occurs due to fuel crossover and internal currents.

The saturation pressure of water vapor  $P_{H_2O}^*$  is a function of the cell temperature  $T$  (K) as expressed in Eq. (12).

$$\log_{10}(P_{H_2O}^*) = 2.95 \times 10^{-2}(T - 273.15) - 9.18 \times 10^{-5}(T - 273.15)^2 + 1.44 \times 10^{-7}(T - 273.15)^3 - 2.18. \quad (12)$$

When the reactants are air and H<sub>2</sub>, their partial pressures are calculated by Eqs. (13) and (14).

$$P_{N_2} = \frac{0.79}{0.21} P_{O_2}, \quad (13)$$

$$P_{O_2} = P_c - (RH_c \times P_{H_2O}^*) - P_{N_2} \times \left(\frac{0.291(i/A)}{T^{0.832}}\right). \quad (14)$$

When the reactants are O<sub>2</sub> and H<sub>2</sub>, the partial pressures are computed according to Eqs. (15) and (16).

$$P_{O_2} = RH_c \times P_{H_2O}^* \left[ \left( \exp \left( \frac{4.192(i/A)}{T^{1.334}} \right) \times \frac{RH_c P_{H_2O}^*}{P_c} \right)^{-1} - 1 \right], \quad (15)$$

$$P_{H_2} = 0.5 RH_a \times P_{H_2O}^* \left[ \left( \exp \left( \frac{1.635(i/A)}{T^{1.334}} \right) \times \frac{RH_a P_{H_2O}^*}{P_a} \right)^{-1} - 1 \right], \quad (16)$$

where  $P_{H_2O}^*$  is the saturation pressure of water vapor (atm),  $P_{N_2}$  is the nitrogen partial pressure at the cathode flow channel (atm).  $P_a$  and  $P_c$  are the inlet pressures (atm) at the anode and cathode, respectively,  $RH_a$  and  $RH_c$  are the relative humidity of vapor in the anode and cathode, and  $A$  is the area of the membrane. The output voltage of the fuel cell stack is expressed in Eq. (17).

$$V_s = n \cdot V_{FC}, \quad (17)$$

where  $n$  is the number of cells connected in series.

We selected a set of parameters  $\vec{\theta} = \{\xi_1, \xi_2, \xi_3, \xi_4, R_c, B, \psi\}$  in order to estimate the operational values with our proposed algorithm [35,50].

### 2.1. Objective functions

For parameter estimation problems in PEMFC there are several works in which different objective functions are used, more information see [53]. In this work, we use three objective functions to estimate the set of unknown parameters,  $\vec{\theta}$ , for the semi-empirical model described in Section 2. The first objective function,  $SSE : \Theta \rightarrow \mathbb{R}_+$ , Eq. (18), is the sum of the squared error between the output of the model and experimental data obtained from a polarization curve. The main goal of using this objective function is comparing with reported results in the literature. When using squared errors a single point which does not fit with the model could completely change the fitted polarization curve, hence to diminish the effect of such points, the second objective function,  $SAE : \Theta \rightarrow \mathbb{R}_+$ , is the sum of the absolute error, Eq. (19). Finally, often, the semi-empirical mathematical model reproduces the polarization curve only under certain conditions, or for some range of values, or the data is noisy or contains errors. For these cases, it is not straight-forward to determine which part of the polarization curve can be reproduced, for this end, we propose to use the median of the absolute error, this objective function  $MAE : \Theta \rightarrow \mathbb{R}_+$ , automatically fits most of the experimental data removing the effect of the points that cannot be adequately fitted.

$$SSE(\theta) = \sum_{i=1}^m \left| V_{FC,i} - V(z_i; \vec{\theta}) \right|^2, \quad (18)$$

$$SAE(\theta) = \sum_{i=1}^m \left| V_{FC,i} - V(z_i; \vec{\theta}) \right|, \quad (19)$$

$$MAE(\theta) = \text{median} \left\{ \left| V_{FC,i} - V(z_i; \vec{\theta}) \right| \right\}_{i=1}^m, \quad (20)$$

where  $m$  is the number of the experimental data,  $V_{FC,i}$  is the experimental data, and  $V(z_i; \vec{\theta})$  is the voltage output of the semi-empirical model. The problem of parameter estimation consists in determining on the search space  $\vec{\theta} \in \Theta$  such that the objective function returns a minimal value in  $\vec{\theta}$ .

#### 2.1.1. Discussion about objective functions

The use of three objective functions has a double purpose: (a) Since SSE, Eq. (18), is the most widely used function in parameter estimation, we use it to obtain results comparable to those previously reported; (b) direct search methods, such as Evolutionary Algorithms and EDAs, only depend on the objective function evaluation; thus, there is no need for proposing derivable or smooth functions. This fact is used in the SAE, Eq. (19), which has the advantage of being more robust than SSE. It is well known that SSE is affected by outliers, which means that a single wrong measure in the experimental data provokes a significant change in the fitted model. Additionally, SAE proportionally weights the error of each point while SSE assigns larger weights, consequently, they delivered different visual results. Besides, minimizing the median of the absolute values (MAE) fits most of the data removing



the effect of the worst fitted points. Thus, the last two functions deliver a *robust* parameter estimation, furthermore, regarding that the semiempirical model, like any other model, is not valid for all inputs of the PEMFC, the median minimization detects the region of the polarization curve which can be reproduced with the semiempirical model.

### 3. Estimation of Distribution Algorithms

Estimation of Distribution Algorithms (EDAs) constitute a kind of metaheuristic algorithm [8,27,28,37] derived from the probabilistic modeling of Genetic Algorithms [14,15]. They were introduced in the field of evolutionary computation by Mühlenbein and Paaß in 1996 [37,58].

EDAs are based on populations in which the descendants of a generation are obtained by sampling from a probabilistic model, estimated from the parent set. Hence, EDAs incorporate learning by estimating a parametric probability distribution from the selected set, intending to associate the highest probability to most promising regions. This is different from other genetic algorithms and other evolutionary algorithms in which the evolution of the populations is carried out by crossover and mutation operators [6,14,15,23–25,30].

#### 3.1. The Univariate Marginal Distribution Algorithm with Gaussian models

In this work, we use an EDA named The Univariate Marginal Distribution Algorithm with Gaussian models ( $UMDA^G$ ) [29], this EDA is the real-coded counterpart of the binary algorithm introduced by Mühlenbein [36], is a generalization of previous works from Syswerda (1993), Eshelman and Schaffer (1993) and Mühlenbein and Voigt (1996) [13,38,49].

The  $UMDA^G$  assumes that the  $l$ -dimensional joint probability function, computed at each iteration, is factored as a product of  $l$  univariate and independent probability functions, as it is shown in Eq. (21).

$$\mathbf{P}_k(\vec{\theta}_{k,h}|S_{\theta_k}) = \mathbf{P}_k(\theta_{k,h}^1, \theta_{k,h}^2, \dots, \theta_{k,h}^l|S_{\theta_k}) = \prod_{i=1}^l \mathcal{P}_{k,i}(\theta_{k,h}^i|S_{\theta_k}). \quad (21)$$

The probability functions  $\mathcal{P}_{k,i}(\theta_{k,h}^i)$  are Normal probability functions with mean  $\mu_{k,i}$  and variance  $\sigma_{k,i}^2$ , for  $0 < i \leq l$ . These parameters are computed as follows:

$$\mu_{k,i}(S_{\theta_k}) = \sum_{s=1}^{N_{ind}} \theta_{k,s}^i p_{k,i}(\theta_{k,s}^i), \quad \sigma_{k,i}^2(S_{\theta_k}) = \sum_{s=1}^{N_{ind}} (\mu_{k,i} - \theta_{k,s}^i)^2 p_{k,i}(\theta_{k,s}^i), \quad (22)$$

where  $p_{k,i}(\theta_{k,s}^i|S_{\theta_k})$  are weights associated with the  $N_{ind}$  individuals.

The weights  $p_{k,i}(\theta_{k,s}^i|S_{\theta_k})$  in Eq. (22) are computed in each iteration  $k$  using the selected set from the truncation method [51], that is, given a threshold  $\nu$  and the subset  $S_{\theta_k} \subset \theta_k$ , the assigned probability to each  $\theta_{k,s}^i$  in the iteration  $k$ , for  $0 < i \leq l$  and  $\vec{\theta}_{k,s} \in \theta_k$ , is given by,

$$p_{k,i}(\theta_{k,s}^i|S_{\theta_k}) = \begin{cases} \frac{1}{\nu}, & \text{if } \vec{\theta}_{k,s} \in S_{\theta_k}, \\ 0, & \text{if } \vec{\theta}_{k,s} \notin S_{\theta_k}. \end{cases} \quad (23)$$

An additional contribution of this work is the stop criterion based on a variance norm. The stop criterion is given by comparing the norm of the normalized variance in Eq. (24) of the parameters with the tolerance  $tol^2$ , (see Algorithm 1). This criterion is less arbitrary of giving a number of generations, and it actually detects when the exploration of the search space is stopped and the best solution found is unlikely improved.

$$norm_k = \left\| \left\{ \frac{var(\theta_i)}{U\_Bound(\theta_i) - L\_Bound(\theta_i)} \right\}_{i=1}^l \right\|. \quad (24)$$

In this research, the  $UMDA^G$  is applied to estimate the parameters  $\vec{\theta} = \{\xi_1, \xi_2, \xi_3, \xi_4, R_c, B, \psi\}$ .

#### 3.2. Parameter settings

The execution parameters of the  $UMDA^G$  are  $N_{ind} = 100$ , the number of selected individuals is  $\nu = 50$ , the tolerance for stopping the algorithm is set to  $tol = 1e^{-10}$  and the search limits, given by the upper and lower bounds for each parameter are shown in Table 2.

**Algorithm 1**  $UMDA^G$ 


---

```

1: Input:  $N_{ind}$ ,  $N_{gen}$ ,  $tol$  and  $v$ .
2: Generate an initial population  $\theta_0$  with  $N_{ind}$  individuals.
3:  $k = 0$ 
4: while  $k < N_{gen}$  and  $norm_k > tol^2$  do
5:   Evaluate  $\theta_k$  on the objective function.
6:   Select the  $v$  best individuals,  $S_{\theta_k} \subset \theta_k$ 
7:   Compute normalized norm  $norm_k$  of the variance of the  $S_{\theta_k}$ .
8:   if  $norm_k < tol^2$  then
9:     Break;
10:  else
11:    Estimate the probability distribution function  $P(x|S_{\theta_k})$ , see Section 3.1.
12:    Sample  $\theta_{k+1}$  with  $N_{ind} - 1$  new individuals from  $P(x|S_{\theta_k})$ .
13:    Replace  $\theta_k$  with  $\theta_{k+1}$ , preserving the best solution of the  $\theta_k$  (elitism).
14:     $k = k + 1$ .
15:  end if
16: end while
17: Return the best solution as the solution of the minimization problem.

```

---

**Table 2**

Upper and lower bound for searching the PEMFC model parameters [35,39].

		$\xi_1$	$\xi_2$	$\xi_3$	$\xi_4$	$R_C$	B	$\psi$
Case 1	L-Bound	−0.952	0.001	7.04e−5	−1.98e−4	1e−4	0.016	14
	U-Bound	−0.944	0.005	7.80e−5	−1.88e−4	8e−4	0.5	23
Case 2	L-Bound	−1.19969	0.001	3.60e−5	−2.60e−4	1e−4	0.0136	10
	U-Bound	−0.8532	0.005	9.80e−5	−9.54e−5	8e−4	0.5	24

These parameters are straight forward to set, in contrast with other algorithms, for example, the tolerance is basically a precision criterion, so, the tolerance is set to the same order of magnitude than the desired precision of the variables. Certainly, it must be below to the physically possible precision, in this case, if the physical parameters can be set with a precision of  $1e^{-4}$ , then we use a precision below this number that is to say  $1e^{-5}$ . Then, the square of this number is  $1e^{-10}$ , considering that the variance is a square of a distance. We recommend setting the number of generations sufficiently large to avoid stopping the algorithm by this condition and use this value only as a safety criterion.

Finally, for finding the adequate population size, it is set to 10 times the number of variables, then it is increased until the result does not improve. Lower population sizes could lead to premature convergence, and suboptimal results, and greater populations will deliver an adequate optimum approximation but using more computational resources than those needed. The threshold for truncating the population is set to half of the population size, reducing this value could slow-down the convergence and could reduce the exploitation. In our experience, for this problem and those reported in the literature [27,29], preserving the half of the population size delivers optimum approximations with high precision. The objective function value of the worst preserved solution is an informative indicator about this parameter, if the worst preserved solution is improved in most of the generations, then the size of the selected set is adequate, if it is not, the size of the selected set must be reduced.

The  $UMDA^G$  is executed using the three objective functions discussed in Section 2.1. The label  $UMDA^G$ -SSE is used to identify the SSE as the objective function,  $UMDA^G$ -SAE for the SAE, and  $UMDA^G$ -MAE for the MAE objective function.

**4. Results**

This section presents results for 4 cases. For test cases we estimate the configuration parameters:  $\{\xi_1, \xi_2, \xi_3, \xi_4, R_C, B, \psi\}$  for validation cases we change the operating conditions of the estimated configuration to demonstrate that the configuration is valid for two different operating conditions. We explain the methodology below.



**Table 3**  
Operational conditions [35].

		Anode pressure (bar)	Cathode pressure (bar)	Temperature (°C)
Test	Case 1	3	5	80
	Case 2	1	1	70
Validation	Case 3	2.5	3	70
	Case 4	1.5	1.5	70

#### 4.1. Definition of test and validation cases

The experimental data were obtained from [35] using the Engauge Digitizer software [34].

We use 4 polarization curves, named as *Case 1*, *Case 2*, *Case 3* and *Case 4*, delivered by 2 PEMFC configurations with different operating conditions. A configuration is a set of values of the parameters:  $\{\xi_1, \xi_2, \xi_3, \xi_4, R_c, B, \psi\}$  while operating conditions are the anode pressure  $P_a$ , the cathode pressure  $P_c$  and temperature  $T$ , see Table 3. In the case studies presented in this article, *Cases 1* and *3* use the same configuration but different operating conditions, and *Cases 2* and *Case 4* use another configuration with different operating conditions. In consequence, if the estimated configuration parameters for Case 1 are correct, they should also reproduce the polarization curve of Case 3, applying the corresponding operating conditions. In all cases, we consider a stack with  $n = 24$  connected PEMFC, with values of  $A = 27 \text{ cm}^2$ ,  $L_c = 127 \text{ }\mu\text{m}$ ,  $J_{max} = 860 \text{ mA cm}^2$ ,  $RH_a = 1$  and  $RH_c = 1$ .

In this regard, we distinguish two kinds of cases named *Test* and *Validation*. On one hand, *Test cases* are used to estimate the configuration parameters  $\{\xi_1, \xi_2, \xi_3, \xi_4, R_c, B, \psi\}$  for the numerical simulation that best reproduces the experimental polarization curve, these are the Cases 1 and 2. On the other hand, *Validation cases* use the estimated configuration for evaluating different operating conditions, to show that a single set of configuration parameters reproduces two polarization curves with different operating conditions. In other words, an optimization execution is used for estimating  $\{\xi_1, \xi_2, \xi_3, \xi_4, R_c, B, \psi\}$  to approximate the polarization curve of *Case 1*. The resulting estimated parameters are used for a numerical simulation with the operating conditions of *Case 3* to demonstrate that the same configuration reproduces polarization curves for both operating conditions.

Hence, Cases 1 and 2 are *Test cases* and Cases 3 and 4 are *Validation cases*, respectively. The estimated parameters, as well as the error, are compared with previously reported methods [3,10,39,52,56,57], with the Genetic Algorithm, Particle Swarm Optimization from the Matlab Optimization Toolbox, and the Covariance Matrix Adaptation Evolution Strategy from [19].

#### 4.2. Numerical results

##### 4.2.1. Comparison with estimated values reported in literature of different EAs

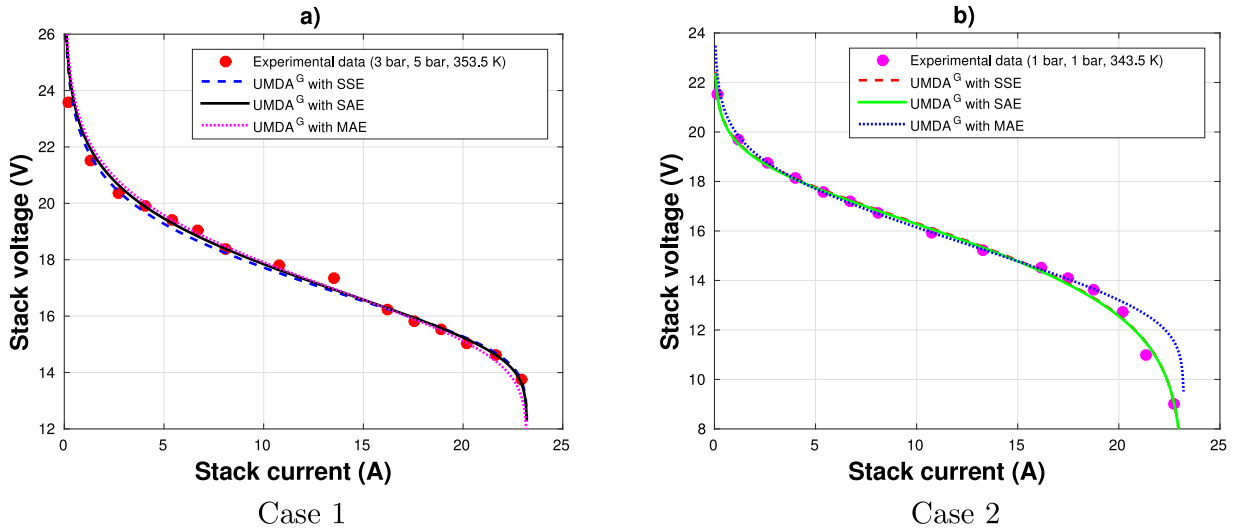
We compare the configuration parameters found by different evolutionary algorithms (EAs) reported in the specialized literature with those found by our  $UMDA^G$  version, we carried out executions using the  $GA_M$  and  $PSO_M$  from the Matlab Optimization Toolbox, and the  $CMAES_M$  code in Matlab obtained from the study in [19].

The results for *Case 1* are shown in Table 4, for instance, the first 6 columns show estimations of configuration parameters reported in the specialized literature by the following bio-inspired algorithms: ARNA-GA [56], SGA [39], RGA [39], MPSO [3], GOA [45,52], SSO [33,52], they include the most recent proposals in the topic. For the sake of completeness, we execute the Matlab versions of the GA and PSO, because they are well-tested implementations. We compare all results with the  $CMAES$  due to this technique have shown competitive results in complex applications [20,21,44]. The main purpose of this table is to show that the  $UMDA^G$  version, introduced here, delivers similar configuration parameters than those found independently by different researchers, using different methodologies and software implementations. Notice that the  $UMDA^G - SSE$  reports the lowest SSE value, and the second and third best are the PSO and GA, respectively.

The best executions with the SAE and MAE objective functions are also reported. The SAE reduces the outlier effect, while the MAE removes completely the outliers because it only uses the median of the error. Notice that even if the MAE uses the error of a single datum, it reports values similar to those of all other methodologies which use the whole data set.

**Table 4**  
Comparison of the best results reported in the literature with those of the  $UMDA^G$ ,  $GA$ ,  $PSO$  and  $CMAES$  for Case 1 with  $P_a = 3$  bar,  $P_c = 5$  bar,  $T = 353.15$  K.

Parameters	ARNA-GA [56]	SGA [39]	RGA [39]	MPSO [3]	SSO [33,52]	GOA [45,52]	$GA_M$	$PSO_M$	$CMAES_M$	$UMDA^G$ -SSE	$UMDA^G$ -SAE	$UMDA^G$ -MAE
$\xi_1$	−0.9470	−0.9473	−0.9506	−0.9479	−0.95078	−0.95024	−0.9492	−0.9442	−0.9510	−0.9519	−0.9489	−0.9479
$\xi_2 (\times 10^{-3})$	3.0586	3.0641	3.0842	3.0857	3.0700	3.0400	2.9671	3.2017	2.9694	3.0092	3.0308	3.0130
$\xi_3 (\times 10^{-5})$	7.6059	7.7134	7.7524	7.799	7.0000	8.0000	7.5744	7.9725	7.3055	7.7882	7.7573	7.5011
$\xi_4 (\times 10^{-4})$	−1.8800	−1.9390	−1.8803	−1.8800	−1.9000	−1.9000	−1.8800	−1.8800	−1.8800	−1.8800	−1.8800	−1.9403
$R_C (\times 10^{-4})$	1.1026	2.7197	1.0027	2.8666	3.1000	1.1000	1.0000	1.0000	1.0000	1.0000	3.966	4.2979
$B (\times 10^{-2})$	3.2898	2.3981	3.2900	2.9682	3.1510	3.2720	1.9314	1.9317	1.9048	1.9406	1.8884	1.8907
$\psi$	23.000	19.765	22.9906	20.7624	23.000	23.000	22.999	23.000	23.000	22.999	20.255	19.4105
SSE	6.17593	4.7483	5.7332	5.5665	3.9818	5.8379	2.003167	2.003191	1.712817	2.003161	–	–
SAE	–	–	–	–	–	–	–	–	–	–	3.53391	–
MAE	–	–	–	–	–	–	–	–	–	–	–	0.077499



**Fig. 2.** Polarization curve fitted with data from [35] for the Test Cases 1 and 2 using  $UMDA^G$ -SSE (sum of squared error),  $UMDA^G$ -SAE (sum of absolute error), and  $UMDA^G$ -MAE (median of absolute error).

For Case 2 in Table 5, the values of the estimated parameters present a larger variance than those of the Case 1, nevertheless, the  $UMDA^G$  reports the best objective function values compared with the reported parameter values obtained by HABC [57], ARNA-GA [56], RGA [39], CHHO [32], GOA [45,52] and SSO [33,52], however,  $CMAES_M$  gets a better SEE value than  $UMDA^G$ .

Fig. 2, shows the experimental polarization curves and those calculated with  $UMDA^G$ -SSE,  $UMDA^G$ -SAE, and  $UMDA^G$ -MAE for Cases 1 and 2. It shows the SSE fitting with a blue dotted line, the SAE fitting with a black line, and the MAE fitting with a magenta dashed line, Fig. 2-(a). For Case 1, we can observe that the SAE fitting is visually closer to the points, this is the effect of using absolute values instead of squared errors, in the Case 2, some points have a greater effect on the fitting, Fig. 2-(b). For the MAE case, the fitting is worse in most of the points, because it only uses the median error, so we are pointing out that this estimation can be performed with noisy data or with less data than the other, and even though it uses a single error value, the estimated parameters are quite close to those estimated with SAE or SSE.

Fig. 2-(a), for Case 1, in the region of ohmic losses, shows a datum at 17 V and 14 A. outside the trend, which forced an intersection between SSE and SAE. As expected, SAE and MAE are more robust and less impacted by this type of data, whose variation could be due to different physical phenomena within the fuel cell such as the resistance to the transfer of protons through the membrane by the effects on the water content in it. Fig. 2-(b) shows the polarization curves computed with the parameters estimated by the  $UMDA^G$ -SSE,  $UMDA^G$ -SAE, and  $UMDA^G$ -MAE methods. The three polarization curves closely fit the experimental data, however, notice that the MAE estimation is slightly different, because this estimation ignores the largest errors and focuses on the median, the main advantage of this estimation is that one can fit an experimental curve even if there are atypical data or noise, because this estimation is the minimization of a single error measurement, nevertheless, one cannot know a priori which such measurement is. Also, notice that a derivative-based algorithm could never approach this objective function.

#### 4.3. Comparison with the most recently reported estimations using a standard dataset

We compare the  $UMDA^G$  respect to the competitive Adaptive Differential Evolution algorithm (ADE) presented in [10] in 2014, and the JAYA and JAYA-NM [52] that are two competitive algorithms published in 2019. Table 6 shows the best solution reported for all the comparing algorithms. In the previous comparison, the data is extracted from the plot of the polarization curve using specialized software, on the contrary, in this case, all the algorithms use exactly the same dataset as it is provided by [52]. The  $UMDA^G$  version used here delivers the best results, recall that the difference between the results in Table 4 and those in Table 6 is due to the small differences in the

**Table 5**  
Comparison of the best results reported in the literature with those of the  $UMDA^G$ ,  $GA$ ,  $PSO$  and  $CMAES$  for Case 2 with  $P_a = 1$  bar,  $P_c = 1$  bar,  $T = 343.15$  K.

Parameters	HABC [57]	RGA [39]	ARNA-GA [56]	CHHO [32]	GOA [45]	SSO [33]	$GA_M$	$PSO_M$	$CMAES_M$	$UMDA^G$ -SSE	$UMDA^G$ -SAE	$UMDA^G$ -MAE
$\xi_1$	−0.8540	−1.1568	−0.8806	−0.85328	−1.17858	−0.89452	−1.0663	−1.1857	−1.0359	−1.0893	−1.1044	−1.0657
$\xi_2 (\times 10^{-3})$	2.8498	3.4243	2.9451	2.27606	3.5100	2.8600	3.0617	3.3152	3.1628	3.3966	3.3444	3.3147
$\xi_3 (\times 10^{-5})$	8.3371	6.4161	8.4438	4.66449	7.0000	8.0000	6.2655	8.4690	4.6929	7.9959	7.3925	7.6165
$\xi_4 (\times 10^{-4})$	−1.294	−1.1544	−1.2883	−1.34608	−1.2000	−1.2000	−0.9756	−1.0429	−1.0433	−1.0338	−1.0314	−1.3870
$R_C (\times 10^{-4})$	1.0000	1.4504	1.0068	3.38141	7.3000	4.3000	1.6186	1.0000	1.0000	1.0379	1.3415	1.9328
$B (\times 10^{-2})$	3.4000	3.4300	3.1675	5.13636	3.4280	3.1470	6.6184	7.1573	7.1590	7.2029	7.1176	2.5227
$\psi$	14.287	12.899	13.486	17.4648	14.78972	12.9415	18.547	24.000	24.000	23.998	23.989	20.896
SSE	11.32189	9.73294	11.17143	6.97008	4.2359	4.5936	0.71834	0.62445	0.62444	0.62588	–	–
SAE	–	–	–	–	–	–	–	–	–	–	2.32169	–
SAE	–	–	–	–	–	–	–	–	–	–	–	0.0209799

**Table 6**

Comparison of  $UMDA^G$ -SSE results using the experimental polarization curve reported in [52]. Case 1 with  $P_a = 3$  bar,  $P_c = 5$  bar,  $T = 353.15$  K.

Parameters	JAYA	JAYA-NM	ADE	$UMDA^G$ -SSE
$\xi_1$	−0.95200	−0.95200	−0.95200	−0.951992
$\xi_2 (\times 10^{-3})$	3.1000	3.1000	3.08643	3.03879
$\xi_3 (\times 10^{-5})$	8.0000	8.0000	7.74271	7.74924
$\xi_4 (\times 10^{-4})$	−1.9000	−1.9000	−1.8800	−1.88096
$R_C (\times 10^{-4})$	1.0000	1.0000	1.0000	1.04613
$B (\times 10^{-2})$	3.2710	3.2700	3.01892	2.57188
$\psi$	23.000	23.000	23.000	22.6666
SSE	3.5054	3.5075	4.4380	1.73619

**Table 7**

Comparison of  $UMDA^G$ -SSE results using the experimental polarization curve reported in [52]. Case 2 ( $P_a = 1$  bar,  $P_c = 1$  bar,  $T = 343.15$  K).

Parameters	JAYA [52]	JAYA-NM [52]	$UMDA^G$ -SSE
$\xi_1$	−1.19966	−1.19966	−1.04147
$\xi_2 (\times 10^{-3})$	3.4800	3.5500	3.25381
$\xi_3 (\times 10^{-5})$	6.0000	6.0000	7.88666
$\xi_4 (\times 10^{-4})$	−1.4000	−1.2000	−1.19061
$R_C (\times 10^{-4})$	2.2000	1.0000	2.90570
$B (\times 10^{-2})$	3.3700	3.3340	6.12714
$\psi$	15.21097	13.2287	22.7246
SSE	4.0410	3.0948	0.62525

datasets. Similarly, Table 7 presents the comparative results of the estimated parameters using the experimental data presented in [52]. Our results show a competitive value of SSE using the  $UMDA^G$  compared to the JAYA and JAYA-NM algorithms. For the sake of completeness, the comparing algorithms for Case 1 and Case 2 are different because not all the authors report both cases.

#### 4.4. Validation cases

Test cases used for estimating the configuration parameters  $\{\xi_1, \xi_2, \xi_3, \xi_4, R_C, B, \psi\}$ , and validation cases use the same configuration but changing the operating conditions that are the anode pressure  $P_a$ , the cathode pressure  $P_c$  and temperature  $T$ . The operational conditions for validation Cases 3 and 4 are in Table 3. Fig. 3, shows the polarization curve for the validation cases, if the estimated parameters for Case 1 are correct, then they must reproduce also the polarization curve of Case 3, because is the same configuration with different operational conditions. For this case, the minimum values of each objective function using by the  $UMDA^G$  were  $SSE = 5.72271$ ,  $SAE = 5.97704$  and  $MAE = 0.24353$ , respectively, see Fig. 3-(a). In the same vein, if the estimated parameters for Case 2 are correct, they must reproduce also the polarization curve of Case 4, thus, Fig. 3-(b), shows that the estimations are correct and the respective minimum values of objective functions were  $SSE = 1.45053$ ,  $SAE = 3.07005$  and  $MAE = 0.16672$ .

We can observe that the SSE values of the validation cases are consistent with the SSE values of the estimation cases, this indicates that the results of the estimates can reproduce different operating conditions and with comparable values of the objective function with respect to the results of the parameter estimation cases.

#### 4.5. Robustness and statistical analysis of 30 independent executions

Bio-inspired algorithms such as evolutionary and swarm-intelligence algorithms are stochastic methods that delivers a different solution each time they are executed. Hence, quite often, they are compared using statistical measures and hypothesis tests, using a series of independent executions. The previous comparisons show the competitiveness of this proposal with reported solutions in the specialized literature. These comparisons are made using the best solutions from each algorithm. According to the results in Tables 4 and 5 the most competitive

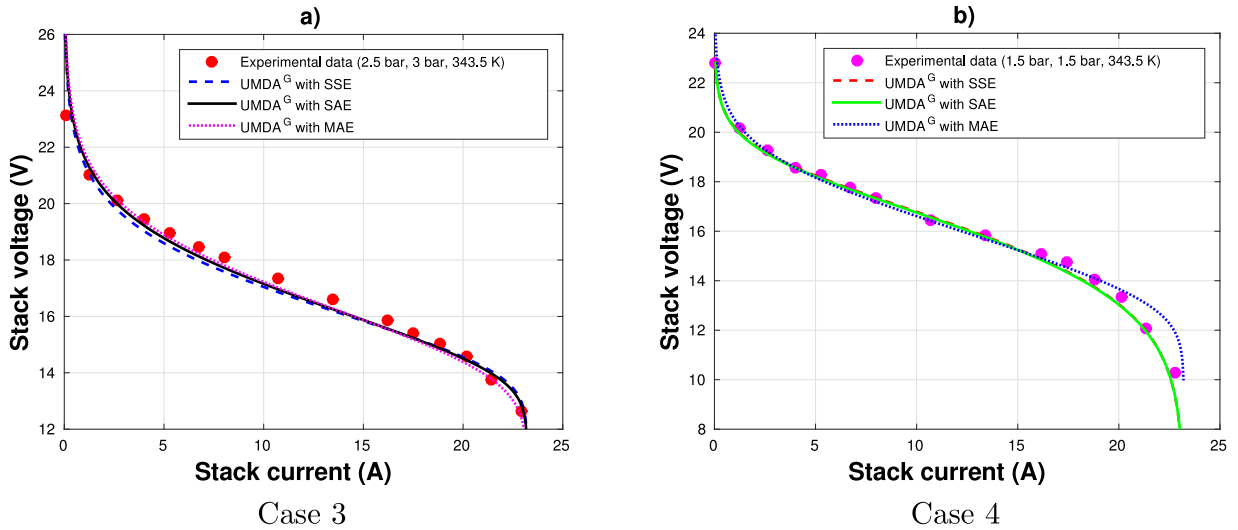


Fig. 3. Polarization curve fitting with data in [35] for the validation cases of  $UMDA^G$ -SSE,  $UMDA^G$ -SAE and  $UMDA^G$ -MAE.

algorithms are the  $UMDA^G$ ,  $PSO$ ,  $GA$ , and  $CMAES$ , thus we perform statistical comparisons and hypothesis test among them.

The columns of Table 8, summarizes the results from 30 independent executions of  $UMDA^G$ ,  $GA_M$ ,  $PSO_M$ , and  $CMAES_M$ . Notice that the configuration parameters delivered by the algorithms present a low standard deviation, and in most of the cases, the lowest is that of the  $UMDA^G$ . The standard deviations are smaller than the mean and the search limits, which means that the  $UMDA^G$  reports a consistent estimation in all the executions, for instance,  $\xi_1$  reports a standard deviation of three orders of magnitude below the mean. In most of the solutions, except one, the parameters of the best solution are inside the region of plus/minus the standard deviation. Thus, it is inferred that the  $UMDA^G$  behaves similarly in all executions. This is a desirable characteristic for an optimization algorithm that is going to be used to evaluate the behavior of PEMFC at different experimental conditions, this shows that it is consistently delivering similar parameters in each execution. In the SSE row, it is shown that  $UMDA^G$  has the minimum mean of the objective function as well as standard deviation, confirming that it is consistent in the estimated parameters and objective value. Hence, the  $UMDA^G$  presents the best performance for this sample of executions, then, we test if this performance is significantly different from that of the other algorithms.

The Bootstrap hypothesis test is a non-parametric test that does not assume anything about the shape of the distributions of the data [12], this is the exact case for these experiments in which we have not a basis to assume a parametric distribution of the results. The hypothesis test is applied over the mean of the objective function,  $SSE$ , for comparing  $GA$ ,  $PSO$ ,  $CMAES$  and  $UMDA^G$ . For Case 1,  $UMDA^G$  is better than  $GA_M$  with a  $p$ -value of  $8.4e^{-4}$ ,  $UMDA^G$  is better than  $PSO_M$  with a  $p$ -value of 0.0210, and  $UMDA^G$  is better than  $CMAES_M$  with a  $p$ -value of  $5.5e^{-5}$ . For Case 2,  $UMDA^G$  is better than  $GA_M$  with a  $p$ -value of 0.000, and it is better than  $PSO_M$  with a  $p$ -value of 0.01689, finally, the  $CMAES_M$  is better than the  $UMDA^G$  with  $p$ -value of 0.0593.

In order to demonstrate that the algorithm is consistent in the estimated configuration parameters even if the objective function is changed, columns four and five show the results from 30 independent executions of the algorithm with the sum of absolute error values (SAE), and the median of the absolute error (MAE), in addition, notice that the maximum change in the means of the parameters estimated with the SSE, SAE and MAE functions is the  $R_c$  parameters, while the main visual change in the polarization curves of Figs. 2 and 3 is the curvature, hence, seemingly there is observed a correlation between the curvature of the polarization curve and the  $R_c$  value, note that this observation comes directly from the execution data without using knowledge of the model.

#### 4.6. Statistical analysis of parameter correlations

This section introduces a statistical analysis using the output data from the 30 independent executions. They are used to compute a correlation matrix to infer the most crucial parameters for the fuel cell performance, if the interaction between the variables is known, we can understand better the operation of a PEMFC.



**Table 8**Mean  $\pm$  standard deviation of parameters and objective functions from 30 independent executions of different optimization algorithms.

Results	Case 1					
Param.	$GA_M$	$PSO_M$	$CMAES_M$	$UMDA^G$ -SSE	$UMDA^G$ -SAE	$UMDA^G$ -MAE
$\xi_1$	$-9.5103e-1 \pm 1.6535e-3$	$-9.4755e-1 \pm 2.6344e-3$	$-9.4745e-1 \pm 3.3867e-3$	$-9.5118e-1 \pm 7.6883e-4$	$-9.5027e-1 \pm 8.9439e-4$	$-9.4974e-1 \pm 9.5197e-4$
$\xi_2$	$3.2992e-3 \pm 4.6253e-5$	$3.1572e-3 \pm 4.3851e-5$	$3.0050e-3 \pm 5.1639e-5$	$3.0008e-3 \pm 9.6906e-6$	$3.0254e-3 \pm 1.2617e-5$	$3.0435e-3 \pm 1.2427e-5$
$\xi_3$	$7.5907e-5 \pm 2.3379e-6$	$7.4658e-5 \pm 2.7633e-6$	$7.4486e-5 \pm 3.0098e-6$	$7.7036e-5 \pm 6.6502e-7$	$7.6801e-5 \pm 7.1080e-7$	$7.6558e-5 \pm 7.5557e-7$
$\xi_4$	$-1.8801e-4 \pm 1.6483e-8$	$-1.8800e-4 \pm 1.5394e-11$	$-1.8805e-4 \pm 1.1870e-7$	$-1.8800e-4 \pm 2.6503e-8$	$-1.8877e-4 \pm 7.8800e-7$	$-1.9251e-4 \pm 9.2184e-7$
$R_C$	$1.5924e-4 \pm 1.7577e-4$	$1.0000e-4 \pm 7.5160e-11$	$1.6255e-4 \pm 1.4067e-4$	$1.1113e-4 \pm 2.2762e-5$	$3.4445e-4 \pm 7.8876e-5$	$2.9185e-4 \pm 7.6196e-5$
$B$ (V)	$1.9211e-2 \pm 1.0672e-3$	$1.8435e-2 \pm 1.4937e-3$	$1.6639e-2 \pm 1.0432e-3$	$1.9526e-2 \pm 4.7134e-4$	$1.9912e-2 \pm 9.7800e-4$	$2.5804e-2 \pm 2.8562e-3$
$\psi$	$2.2207e+1 \pm 1.7390e+0$	$2.2468e+1 \pm 8.9720e-1$	$1.8369e+1 \pm 3.4946e+0$	$2.2906e+1 \pm 1.4555e-1$	$2.0797e+1 \pm 5.8818e-1$	$2.1217e+1 \pm 1.0486$
$SSE$	$2.1912e+0 \pm 5.0726e-1$	$2.0272e+0 \pm 4.0618e-2$	$2.5614 \pm 0.8652$	$2.0128e+0 \pm 7.9793e-3$		
$SAE$					$3.6002e+0 \pm 3.1620e-2$	
$MAE$						$8.8051e-2 \pm 2.8562e-3$
Param.	Case 2					
	$GA_M$	$PSO_M$	$CMAES_M$	$UMDA^G$ -SSE	$UMDA^G$ -SAE	$UMDA^G$ -MED
$\xi_1$	$-1.0492e+0 \pm 9.8643e-2$	$-1.0600e+0 \pm 1.0199e-1$	$-1.0575e+0 \pm 1.2466e-1$	$-1.0726e+0 \pm 2.2366e-2$	$-1.0718e+0 \pm 3.0424e-2$	$-1.0706e+0 \pm 2.8523e-2$
$\xi_2$	$3.1150e-3 \pm 4.0220e-4$	$3.0070e-3 \pm 3.8254e-4$	$3.1929e-3 \pm 5.6167e-4$	$3.2896e-3 \pm 8.4091e-5$	$3.2941e-3 \pm 1.2296e-4$	$3.2847e-3 \pm 8.2161e-5$
$\xi_3$	$6.9043e-5 \pm 1.9248e-5$	$6.0434e-5 \pm 1.7888e-5$	$7.2809e-5 \pm 2.0207e-5$	$7.6164e-5 \pm 5.1028e-6$	$7.6865e-5 \pm 5.7401e-6$	$7.3648e-5 \pm 4.3256e-6$
$\xi_4$	$-9.6049e-5 \pm 1.1258e-6$	$-1.0196e-4 \pm 3.4118e-6$	$-1.0391e-4 \pm 1.7658e-6$	$-1.0374e-4 \pm 1.3579e-6$	$-1.0353e-4 \pm 9.2311e-7$	$-1.3380e-4 \pm 8.2798e-6$
$R_C$	$2.2783e-4 \pm 3.7371e-4$	$2.0182e-4 \pm 1.6551e-4$	$1.0389e-4 \pm 2.0065e-5$	$1.5965e-4 \pm 5.5751e-5$	$2.1284e-4 \pm 5.5504e-5$	$1.9381e-3 \pm 4.7488e-4$
$B$ (V)	$6.5006e-2 \pm 1.5983e-3$	$7.0679e-2 \pm 1.0345e-3$	$7.1508e-2 \pm 3.9289e-4$	$7.0655e-2 \pm 1.1037e-3$	$7.0004e-2 \pm 7.2960e-4$	$2.5077e-2 \pm 1.3578e-3$
$\psi$	$1.7782e+1 \pm 2.2909e+0$	$2.3160e+1 \pm 1.2587e+0$	$2.3799e+1 \pm 8.9478e-1$	$2.3407e+1 \pm 5.0261e-1$	$2.3374e+1 \pm 3.3991e-1$	$1.9850e+1 \pm 1.1214$
$SE$	$7.6147e-1 \pm 4.1220e-2$	$6.4849e-1 \pm 2.9919e-2$	$6.2931e-1 \pm 2.4323e-2$	$6.3744e-1 \pm 7.0570e-3$		
$SAE$					$2.3447e+0 \pm 1.2713e-2$	
$MAE$						$2.6324e-2 \pm 4.4035e-3$

In Fig. 4, the lower triangular matrix shows the correlation values between the row and column parameters, while the upper triangular matrix shows an ellipse if the two parameters are considered correlated. To know if two variables are correlated we applied a non-parametric hypothesis test with a 0.9 confidence level, using the `cor.test` function from the R language [1]. The obtained value was used to capture any possible relation between two variables. Correlations and a hypothesis test are computed, using the Kendall formula, to capture non-linear correlations. *If a variable is highly correlated with the objective function, then, we elucidate that a perturbation in such a variable significantly perturbs the cell performance. Correspondingly, if two or more variables are correlated, one elucidates that such variables must be perturbed together to significantly modify the cell performance.* Fig. 4 shows that in all cases, the most impacting variable in the objective function is the water content in the membrane,  $\psi$ , due to all four correlation matrices present thin ellipses in the position of  $\psi$  row and objective function column. The purpose of showing matrices from SEE and SAE is to show the variables that are consistently correlated in both estimations, thus, it is more plausible that these correlations are correct, notice that  $\psi$  is present in all matrices, which additionally is consistent with the physics of the cell. Other less impacting correlations with the objective functions are those of  $\xi_4$  and  $\xi_2$ . Notice that  $\xi_2$  shows a high correlation for the first case but not for the second. The semi-empirical coefficients  $\xi$  cannot be directly tuned in a real-world fuel cell, because they are a function of other physical parameters which, for the sake of completeness, are explicitly defined in Ref. [11]. On the other hand, the water content in the membrane,  $\psi$ , is the most important correlation detected in this analysis, considering that it could be directly adjusted in a real fuel cell and that it is the most impacting parameter of the cell performance. Finally, notice that  $\psi_2$  and  $\psi_4$  present the second-largest correlation with the objective function for Case 1 and Case 2, respectively.

#### 4.7. Discussion

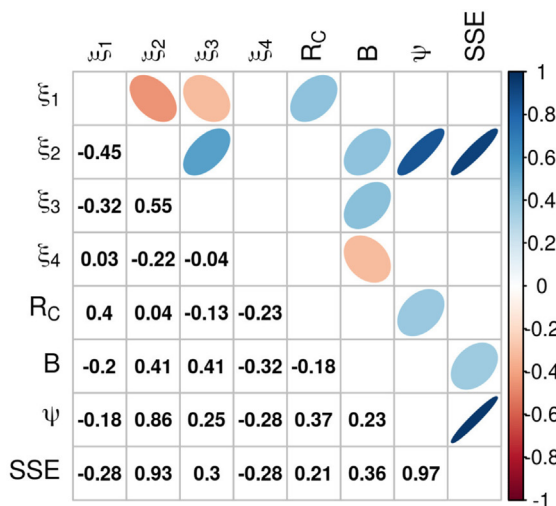
There exist different mathematical models, each of them allows the study and analysis of different processes involved in the operation and performance of a fuel cell. The main advantage of the semi-empirical mathematical model is that it reproduces the whole polarization curve in contrast with the macro-homogeneous [26] and the agglomerate models [46], which partially reproduce the curve. On the opposite side, it obfuscates details of the internal processes and the impact of the fuel-cell geometry.

Hence, we recommend this model for a first-approach design, then, a more detailed design can be produced and simulated via the macro-homogeneous and the agglomerate models, in this sense, the different mathematical models are complementary. The semi-empirical model permits us to understand the effect of the semi-empirical parameters  $\xi_1, \xi_2, \xi_3, \xi_4$  on the performance, measured via the polarization curve.

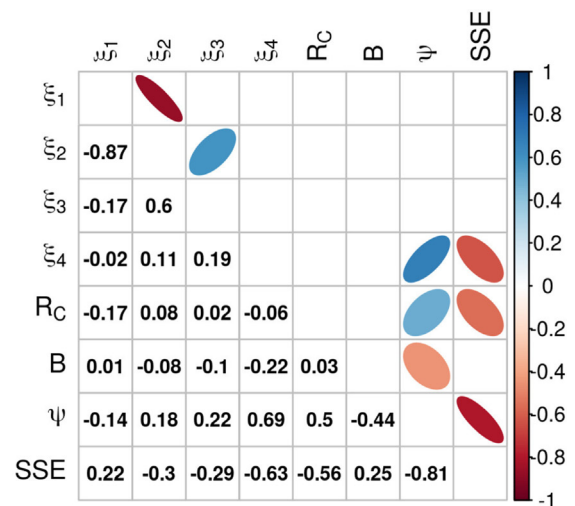
The application of the  $UMDA^G$  in this article shows its feasibility as a tool for estimating parameters in the semi-empirical model for PEMFC, in addition to finding a solution with less error than all the other proposals in the literature. The results of the  $UMDA^G$  are comparable with the results of the  $GA$ , the  $PSO$ , and the  $CMAES$ . Table 8, shows that, for Case 1, the  $UMDA^G$  presented a smaller standard deviation for SSE than the other algorithms, the  $CMAES$  presented a higher standard deviation for SSE compared to  $GA$ ,  $PSO$ , and  $UMDA$ , and the bootstrap hypothesis test indicates that exists more variability in its solutions. Only for Case 2, the  $CMAES$  outperformed all algorithms in SSE values. Our proposal presents two important benefits: it requires of a single user-given parameter, that is the population size, which is straight forward set by increasing an initial guess until there is no improvement in the optimum approximation, besides, in contrast with other evolutionary algorithms, this proposal does not require of an arbitrary or expert-set stopping criterion, the convergence is detected via the norm of the variance of the current population and the algorithm is automatically stopped.

## 5. Conclusions

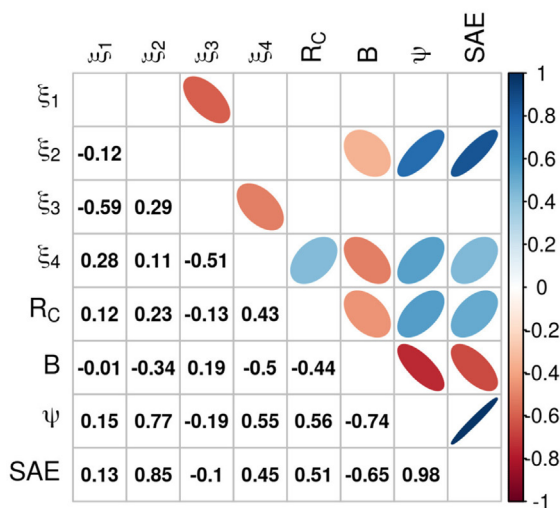
In this work, we present a statistical comparison of four evolutionary algorithms for the parameter estimation of a PEMFC, they are the  $UMGA^G$ ,  $GA$ ,  $PSO$ , and  $CMAES$ , in addition, they are compared with reported results in the literature. We found advantageous to use the  $UMDA^G$  in contrast with other methods, besides its simplicity, low computational cost in operations and memory, and its performance to estimate the parameters. Then, we propose to use the convergence, measured via the variance of the search distribution, as a stopping criterion for the  $UMDA^G$ . The convergence property, which is not the rule in Evolutionary Algorithms, is advantageous for reducing the number of execution parameters and avoiding arbitrary stopping criteria such as the number of



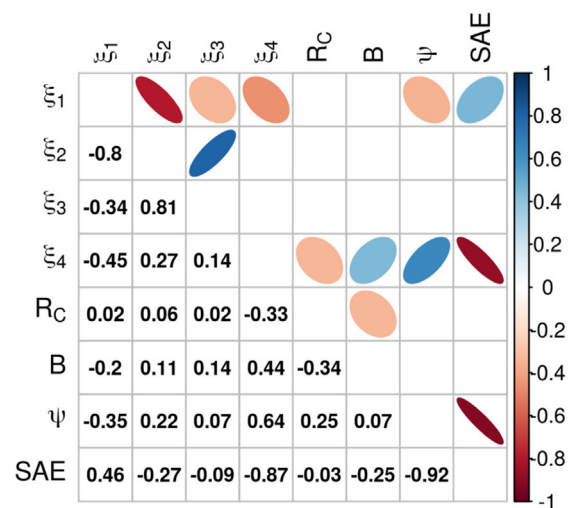
Case 1 with SSE



Case 2 with SSE



Case 1 with SAE



Case 2 with SAE

**Fig. 4.** Correlation matrix of 30 independent executions of cases 1 and 2 with SSE and SAE. The lower side of the correlation matrix shows numeric values, while the upper shows an ellipse representation, the narrower an ellipse is the higher the correlation.

iterations/generations, which provokes suboptimal solutions, wasted computations, or requires of expert tuning. The *UMDA<sup>G</sup>* improves the confidence of the results because it converges to a small region in the variable space and the objective space. This ensures two crucial features of the solution: (1) robustness, considering that the parameters estimated in 30 independent executions belong to a closed neighborhood and share similar objective values, that is to say, any of the 30 executions of our proposal delivers an adequate estimation, (2) the solution is a closed approximation to a local optimum because the algorithm intensively samples in a very small neighborhood of the best solution at the last iterations.

The  $UMDA^G$  is suited for handling non-derivable and multimodal objective functions, we use this advantage for approaching the minimum of three objective functions, which solution is consistent with previously reported results, but presents a lower error, besides we approach robust estimations. In this regard, the SSE is quite sensitive to outliers, nevertheless, it is the most widely used in literature due to its differentiability. The objective function that uses the sum of the absolute values of the error, SAE, is less biased by atypical data, while MAE practically avoids any outlier.

The information from several executions is used to determine relations among variables and the objective function, in this regard, we infer that the water content in the membrane,  $\psi$ , is the most critical parameter in the PEMFCs used in this study.

The results showed that the  $UMDA^G$  is a competitive algorithm compared with the state of the art and the most recent bio-inspired algorithms in the literature.

Future work contemplates the use of parameters estimated for the semi-empirical model to feed the agglomerate and macro-homogeneous models, to define consistent simulations of the same fuel cell through all of them.

### CRedit authorship contribution statement

**Luis Blanco-Cocom:** Software, Data curation, Writing - original draft, Visualization, Investigation. **Salvador Botello-Rionda:** Project administration, Supervision, Conceptualization, Methodology, Resources. **L.C. Ordoñez:** Validation, Writing - review & editing. **S. Ivvan Valdez:** Formal analysis, Writing - original draft, Writing - review & editing, Conceptualization, Methodology, Supervision, Visualization.

### Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

### Acknowledgments

This article was supported by CONACYT, Mexico under grant 100236.

S. Ivvan Valdez is supported by Grant Cátedras CONACYT, Mexico number 7795.

### References

- [1] R Core Team, R: A Language and Environment for Statistical Computing, R Foundation for Statistical Computing, Vienna, Austria, 2018.
- [2] A. Asghar Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, H. Chen, Harris hawks optimization: Algorithm and applications, *Future Gener. Comput. Syst.* 97 (2019) 849–872.
- [3] A. Askarzadeh, A. Rezazadeh, Optimization of pemfc model parameters with a modified particle swarm optimization, *Int. J. Energy Res.* 35 (2011a) 1258–1265.
- [4] A. Askarzadeh, A. Rezazadeh, A grouping-based global harmony search algorithm for modeling of proton exchange membrane fuel cell, *Int. J. Hydrogen Energy* 36 (2011b) 5047–5053.
- [5] A. Askarzadeh, A. Rezazadeh, Artificial immune system-based parameter extraction of proton exchange membrane fuel cell, *Int. J. Electr. Power Energy Syst.* 33 (2011c) 933–938.
- [6] S. Bandyopadhyay, H. Kargupta, G. Wang, Revisiting the gemga: Scalable evolutionary optimization through linkage learning, in: *Proceedings of the 1998 IEEE International Conference on Evolutionary Computation*, IEEE Press, 1998, pp. 603–608.
- [7] C. Berger, *Handbook of Fuel Cell Technology*, Prentice Hall, Engelwood Cliffs, NJ, 1968.
- [8] P.A.N. Bosman, D. Thierens, Linkage information processing in distribution estimation algorithms, in: *Proceedings of the Genetic and Evolutionary Computation Conference GECCO-99* 1, 1999, pp. 60–67.
- [9] Y. Chen, N. Wang, Cuckoo search algorithm with explosion operator for modeling proton exchange membrane fuel cells, *Int. J. Hydrogen Energy* 44 (2019) 3075–3087.
- [10] J. Cheng, G. Zhang, Parameter fitting of pemfc models based on adaptive differential evolution, *Int. J. Electr. Power Energy Syst.* 62 (2014) 189–198.
- [11] J. Corrêa, F. Farret, L. Canha, M. Simoes, An electrochemical-based fuel-cell model suitable for electrical engineering automation approach, *IEEE Trans. Ind. Electron.* 51 (5) (2004) 1103–1112.
- [12] B. Efron, *The Jackknife, the Bootstrap, and Other Resampling Plans*, Vol. 38, Siam, 1982.
- [13] L.J. Eshelman, J.D. Schaffer, Productive recombination and propagating and preserving schemata, *Foundations of Genetic Algorithms*, Vol. 3, 1993, pp. 299–314.
- [14] D.E. Goldberg, K. Deb, H. Kargupta, G. Harik, Rapid, accurate optimization of difficult problems using fast messy genetic algorithms, in: S. Forrest (Ed.), *Proceedings of the Fifth International Conference on Genetic Algorithms*, 1, Morgan Kaufman, 1993, pp. 56–64.

- [15] D.E. Goldberg, B. Korb, K. Deb, Messy genetic algorithms: Motivation, analysis and first results, *Complex Syst.* 3 (5) (1989) 493–530.
- [16] Z. Gong, C. Liu, K.L. Teo, J. Sun, Distributionally robust parameter identification of a time-delay dynamical system with stochastic measurements, *Appl. Math. Model.* 69 (2019) 685–695.
- [17] C. Guo, J. Lu, ZhongTian, W. Guo, A. Darvishan, Optimization of critical parameters of pem fuel cell using tlbo-de based on elman neural network, *Energy Convers. Manage.* 183 (2019) 149–158.
- [18] W. Han, D. Li, D. Yu, H. Ebrahimian, Optimal parameters of pem fuel cells using chaotic binary shark smell optimizer, *Energy Sources Part A: Recovery Util. Env. Eff.* (2019) 1–15.
- [19] N. Hansen, S. Kern, Evaluating the cma evolution strategy on multimodal test functions, in: *Parallel Problem Solving from Nature - PPSN VIII*, Springer Berlin Heidelberg, Berlin, Heidelberg, 2004, pp. 282–291.
- [20] N. Hansen, S.D. Müller, P. Koumoutsakos, Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (CMA-ES), *Evol. Comput.* 11 (2003) 1–18.
- [21] N. Hansen, A. Ostermeier, Completely derandomized self-adaptation in evolution strategies, *Evol. Comput.* 9 (2001) 159–195.
- [22] M. Kandidayeni, A. Macias, A. Khalatbarisoltani, L. Boulon, S. Kelouwani, Benchmark of proton exchange membrane fuel cell parameters extraction with metaheuristic optimization algorithms, *Energy* 183 (2019) 912–925.
- [23] H. Kargupta, The gene expression messy genetic algorithm, In *Proceedings of the 1996 IEEE International Conference on Evolutionary Computation*, 1996, pp. 631–636.
- [24] H. Kargupta, D.E. Goldberg, Search, blackbox optimization, and sample complexity, in: R.W. Belew, M. Vose (Eds.), *Foundations of Genetic Algorithms 4*, Morgan Kaufmann, 1997.
- [25] C.H.M. van Kemenade, Building block filtering and mixing, in: *Proceedings of the 1998 International Conference on Evolutionary Computation*, IEEE Press, 1998.
- [26] N. Khajeh-Hosseini-Dalasm, M. Kermani, D.G. Moghaddam, J. Stockie, A parametric study of cathode catalyst layer structural parameters on the performance of a pem fuel cell, *Int. J. Hydrogen Energy* 35 (2010) 2417–2427.
- [27] P. Larrañaga, R. Etxeberria, J.A. Lozano, J.M. Peña, Optimization in continuous domains by learning and simulation of gaussian networks, in: A.S. Wu (Ed.), *Proc. of the Genetic and Evolutionary Computation Conference, GECCO-2000, Workshop Program 1 (2000b)* 201–204.
- [28] P. Larrañaga, R. Etxeberria, J.A. Lozano, J.M. Peña, Combinatorial optimization by learning and simulation of bayesian networks, in: C. Boutilier, M. Goldszmidt (Eds.), *Uncertainty in Artificial Intelligence, UAI-2000 1*, 2000a, pp. 343–352.
- [29] P. Larrañaga, J.A. Lozano, *Estimation of Distribution Algorithms: A New Tool for Evolutionary Computation*, 1, Kluwer Academic Pub., 2002.
- [30] F.G. Lobo, K. Deb, D.E. Goldberg, G.R. Harik, L. Wang, Compressed introns in a linkage learning genetic algorithm, in: *Genetic Programming 1998: Proceedings of the Third Annual Conference*, Morgan Kaufman, 1998, pp. 551–558.
- [31] R.F. Mann, J.C. Amphlett, M.A. Hooper, H.M. Jensen, B.A. Peppley, P.R. Roberge, Development and application of a generalised steady-state electrochemical model for a pem fuel cell, *J. Power Sources* 86 (2000) 173–180.
- [32] A.S. Menesy, H.M. Sultan, A. Selim, M.G. Ashmawy, S. Kamel, Developing and applying chaotic harris hawks optimization technique for extracting parameters of several proton exchange membrane fuel cell stacks, *IEEE Access* 8 (2020) 1146–1159.
- [33] S. Mirjalili, A.H. Gandomi, S.Z. Mirjalili, S. Saremi, H. Faris, S.M. Mirjalili, Salp swarm algorithm: A bio-inspired optimizer for engineering design problems, *Adv. Eng. Softw.* 114 (2017) 163–191.
- [34] M. Mitchell, B. Muftakhidinov, T. Winchen, Z. Jedrzejewski-Szmek, T. Badger, Engauge digitizer, 2020, URL <http://markummittchell.github.io/engauge-digitizer>, 20-08-2020.
- [35] Z.J. Mo, X.J. Zhu, L.Y. Wei, G.Y. Cao, Parameter optimization for a pemfc model with a hybrid genetic algorithm, *Int. J. Energy Res.* 20 (2006) 585–597.
- [36] H. Mühlenbein, The equation for response to selection and its use for prediction, *Evol. Comput.* 5 (1998) 303–346.
- [37] H. Mühlenbein, G. Paaß, From recombination of genes to the estimation of distributions i. binary parameters, in: H.-M. Voigt, W. Ebeling, I. Rechenberg, H.-P. Schwefel (Eds.), *Parallel Problem Solving from Nature — PPSN IV*, Springer Berlin Heidelberg, Berlin, Heidelberg, 1996, pp. 178–187.
- [38] H. Mühlenbein, H.M. Voigt, Gene pool recombination in genetic algorithms, in: *Metaheuristics: Theory and Applications*, 1996, pp. 53–62.
- [39] M. Ohenoja, K. Leiviskä, Validation of genetic algorithm results in a fuel cell model, *Int. J. Hydrogen Energy* 35 (2010) 12618–12625.
- [40] M. Outeiro, R. Chibante, A. Carvalho, A. de Almeida, A parameter optimized model of a proton exchange membrane fuel cell including temperature effects, *J. Power Sources* 185 (2008) 952–960.
- [41] M.T. Outeiro, R. Chibante, A.S. Carvalho, A.T. de Almeida, A new parameter extraction method for accurate modeling of pem fuel cells, *Int. J. Energy Res.* 33 (2009) 978–988.
- [42] F. Qin, P. Liu, H. Niu, H. Song, N. Yousefi, Parameter estimation of pemfc based on improved fluid search optimization algorithm, *Energy Rep.* 6 (2020) 1224–1232.
- [43] Y. Rao, Z. Shao, A.H. Ahangarnejad, E. Gholamalazadeh, B. Sobhani, Shark smell optimizer applied to identify the optimal parameters of the proton exchange membrane fuel cell model, *Energy Convers. Manage.* 182 (2019) 1–8.
- [44] R. Ros, N. Hansen, A simple modification in cma-es achieving linear time and space complexity, in: *Parallel Problem Solving from Nature – PPSN X*, Springer Berlin Heidelberg, Berlin, Heidelberg, 2008, pp. 296–305.
- [45] S. Saremi, S. Mirjalili, A. Lewis, Grasshopper optimisation algorithm: Theory and application, *Adv. Eng. Softw.* 105 (2017) 30–47.
- [46] M. Secanell, K. Karan, A. Suleman, N. Djilali, Multi-variable optimization of pemfc cathodes using an agglomerate model, *Electrochim. Acta* 52 (2007) 6318–6337.
- [47] B. Shen, C. Liu, J. Ye, E. Feng, Z. Xiu, Parameter identification and optimization algorithm in microbial continuous culture, *Appl. Math. Model.* 36 (2012) 585–595.

- [48] D. Song, Q. Wang, Z. Liu, T. Navessin, M. Eikerling, S. Holdcroft, Numerical optimization study of the catalyst layer of PEM fuel cell cathode, *J. Power Sources* 126 (2004) 104–111.
- [49] G. Syswerda, Simulated crossover in genetic algorithms, *Foundations of Genetic Algorithms*, Vol. 2, 1993, pp. 239–255.
- [50] O.E. Turgut, M.T. Coban, Optimal proton exchange membrane fuel cell modelling based on hybrid teaching learning based optimization-differential evolution algorithm, *Ain Shams Eng. J.* 7 (1) (2016) 347–360.
- [51] S.I. Valdez, A. Hernández, S. Botello, Efficient estimation of distribution algorithms by using the empirical selection distribution, in: Peter Korosec (Ed.), *New Achievements in Evolutionary Computation*, InTech, ISBN: 978-953-307-053-7, 2010.
- [52] S. Xu, Y. Wang, Z. Wang, Parameter estimation of proton exchange membrane fuel cells using eagle strategy based on jaya algorithm and nelder-mead simplex method, *Energy* 173 (2019) 457–467.
- [53] B. Yang, J. Wang, L. Yu, H. Shu, T. Yu, X. Zhang, W. Yao, L. Sun, A critical survey on proton exchange membrane fuel cell parameter estimation using meta-heuristic algorithms, *J. Cleaner Prod.* 265 (2020) 121660.
- [54] M. Ye, X. Wang, Y. Xu, Parameter identification for proton exchange membrane fuel cell model using particle swarm optimization, *Int. J. Hydrogen Energy* 34 (2009) 981–989.
- [55] Z. Yuan, W. Wang, H. Wang, A. Yildizbasi, Developed coyote optimization algorithm and its application to optimal parameters estimation of pemfc model, *Energy Rep.* 6 (2020) 1106–1117.
- [56] L. Zhang, N. Wang, An adaptive rna genetic algorithm for modeling of proton exchange membrane fuel cells, *Int. J. Hydrogen Energy* 38 (2013) 219–228.
- [57] W. Zhang, N. Wang, S. Yang, Hybrid artificial bee colony algorithm for parameter estimation of proton exchange membrane fuel cell, *Int. J. Hydrogen Energy* 38 (2013) 5796–5806.
- [58] A.A. Zhigljavsky, *Theory of Global Random Search*, Kluwer Academic Publishers, 1991.
- [59] X. Zhu, N. Wang, Cuckoo search algorithm with onlooker bee search for modeling pemfcs using t2fnn, *Eng. Appl. Artif. Intell.* 85 (2019) 740–753.