IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received May 14, 2020, accepted June 10, 2020, date of publication June 19, 2020, date of current version July 1, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3003741

Analyzing the Nearly Optimal Solutions in a Multi-Objective Optimization Approach for the Multivariable Nonlinear Identification of a PEM Fuel Cell Cooling System

ALBERTO PAJARES, F. XAVIER BLASCO, (Member, IEEE), JUAN MANUEL HERRERO, AND JOSÉ VICENTE SALCEDO

Instituto Universitario de Automática e Informática Industrial, Universitat Politècnica de València, 46022 Valencia, Spain

Corresponding author: Alberto Pajares (alpafer1@upv.es)

This work was supported in part by the Ministerio de Ciencia, Innovación y Universidades, Spain, under Grant RTI2018-096904-B-I00, and in part by the Generalitat Valenciana Regional Government under Project AICO/2019/055.

ABSTRACT In this work, the parametric identification of a cooling system in a PEM (proton exchange membrane) fuel cell is carried out. This system is multivariable and nonlinear. In this type of system there are different objectives and the unmodeled dynamics cause conflicting objectives (prediction errors in each output). For this reason, resolution is proposed using a multi-objective optimization approach. Nearly optimal alternatives can exist in any optimization problem. Among them, the nearly optimal solutions that are significantly different (that we call nearly optimal solutions nondominated in their neighborhood) are potentially useful solutions. In identification problems, two situations arise for consideration: 1) aggregation in the design objectives (when considering the prediction error throughout the identification test). When an aggregation occurs in the design objectives, interesting non-neighboring (significantly different) multimodal and nearly optimal alternatives appear. These alternatives have different trade-offs in the aggregated objectives; 2) new objectives in decision making appear. Some models can, with similar performance in the design objectives, obtain a significant improvement in new objectives not included in the optimization phase. A typical case of additional objectives are the validation objectives. In these situations, nearly optimal solutions nondominated in their neighborhood play a key role. These alternatives allow the designer to make the final decision with more valuable information. Therefore, this work highlights, as a novelty, the relevance of considering nearly optimal models nondominated in their neighborhood in problems of parametric identification of multivariable nonlinear systems and shows an application in a complex problem.

INDEX TERMS Multi-objective, nearly optimal, multivariable nonlinear system identification, PEM fuel cell.

I. INTRODUCTION

Many engineering problems are usually solved using optimization techniques [1]–[4]. Often these problems have different and conflicting objectives. Therefore, it is useful to treat them as multi-objective optimization problems (MOP [5]–[8]). The solution to an MOP is a set of optimal solutions (Pareto front) where the designer can analyze the performance balance of each to decide the solution that he considers most adequate [7].

For the parametric identification of multivariable models, the objectives to be optimized are related to the errors on each of the outputs. These errors are obtained by comparing the model predictions with experimental data from the process. Typically, tests are carried out to identify and validate the model parameters [9]. The input signals for the process are planned (multistep, for example) to enable the system to reach the different desired operating points. Multivariable models have several outputs, and therefore different objectives naturally arise for simultaneous optimization. Due to unmodeled dynamics, these objectives typically conflict. Therefore, an MOP enables the designer to analyze the exchange of performance for each model for each design objective.

The associate editor coordinating the review of this manuscript and approving it for publication was Di He.

In an MOP, in addition to the optimal solutions, there are nearly optimal alternatives that can be useful for the designer (also called approximate or ϵ-efficient solutions [10]–[12]) but these alternatives are ignored in a classic MOP resolution. Among them, the solutions with similar performance to the optimal ones in the objective space, and that differ significantly in the parameter space, are the really useful alternatives for the designer [13]–[15]. Under our approach, these solutions are the nearly optimal solutions nondominated in their neighborhood (potentially useful alternatives [14]). These alternatives provide diversity to the set of solutions obtained without excessively increasing the number of alternatives to be analyzed in the decision stage. In [16], [17], the multi-objective approach when considering these alternatives has demonstrated its usefulness for control tuning problems. For example, in [16], the nearly optimal solutions nondominated in their neighborhood have different characteristics than the optimal ones (robustness to uncertainty in the model, performance in a new scenario, etc.) and, in this particular case, the designer preferred a nearly optimal solution rather than an optimal one. In any case, these alternatives provide valuable information for the designer. This greater diversity enables the designer to make the final decision wisely. In this work we propose to use this methodology in the field of system identification. Specifically, in the parametric identification of nonlinear multivariable dynamic systems.

From an optimization point of view, there are two situations where the nearly optimal solutions nondominated in their neighborhood are especially useful:

1) *When the design objectives are formed with the aggregation of partial objectives*. When this occurs, the appearance of multimodalities and nearly optimal non-neighboring solutions is common. This is due to the apparition of significantly different alternatives (shown by different performance in the partial objectives) that present the same or similar value for the design objectives. In fact, these alternatives could be optimal solutions if the partial objectives were optimized independently.
2) *When new objectives are incorporated in the decision-making phase*. There are usually different solutions with similar values for the design objectives. In this scenario, the designer can take into account new objectives, not contemplated in the optimization phase, to make a better informed final decision. In this context, the nearly optimal alternatives nondominated in their neighborhood can provide a significant improvement in these new objectives, and obtain a similar performance in the design objectives.

Therefore, in both cases, the nearly optimal solutions nondominated in their neighborhood can be potentially useful. Therefore, these solutions are worthy of study at the decision-making stage. These two situations also occur when we approach the identification of nonlinear multivariable systems using multi-objective optimization:

1) The phenomenon due to the aggregation of partial objectives appears naturally when, for example, the error is evaluated at different operating points in the same test. It is to be expected that there are always unmodelled dynamics and this may cause a model to fit well at one operating point, but not so well at another point. Therefore, there is an aggregation of the error on each different operating point. The model could nevertheless be an interesting solution but it may not be in the Pareto front solutions because of small differences in the objective functions.
2) New objectives appear naturally in the decision-making phase when the models obtained must be validated. In this phase, the prediction errors of the models can be analyzed using an experimental validation dataset and this can be considered as a new objective. There could be nearly optimal models nondominated in their neighborhood with a significant improvement in these non-optimized objectives, and obtaining a similar performance in the design objectives. Therefore, these models are especially relevant alternatives.

This work shows as a novelty the usefulness of the nearly optimal solutions nondominated in their neighborhood in the parametric identification of nonlinear multivariable systems. In this paper, we model a cooling system for a proton exchange membrane (PEM [18], [19]) fuel cell (also named PEMFC). The experimental data and nonlinear model described in [20] are employed. The PEMFC stack is part of a micro-combined heat and power (μ-CHP) system [21]–[24]. μ-CHP systems are cogeneration systems oriented to domestic use and produce thermal and electrical energy simultaneously. Fuel cells are a promising technology for such systems [25]. The durability, cost, reliability, and energy efficiency of the stack largely depend on the correct design of its cooling control system [26]–[29]. For an adequate design of the control system it is necessary to have an accurate nonlinear model [30]–[33]. By using the methodology presented in this paper, optimal and nearly optimal models that are nondominated in their neighborhood are obtained using the evolutionary nevMOGA [14]. These models provide the designer with significantly different but relevant alternatives and the designer can make a final decision using this additional valuable information. This methodology can deal with problems that have a large number of objectives and decision variables. In this context, it is necessary to define an appropriate configuration of nevMOGA so that the number of solutions to be obtained is not excessive, otherwise the computational cost increases considerably. This can be done, for example, by reducing the nearly optimal zone, or by increasing the neighborhood, among others. The MOP discussed in this work has a large number of decision variables.

This work is structured as follows. In Section II some basic multi-objective backgrounds are presented. In Section III, the nevMOGA algorithm used in this work is described. The utility of the nearly optimal solutions nondominated in their neighborhood in some particular cases are demonstrated

in Section IV. In Section V, the parametric identification of nonlinear multivariable dynamic system as MOPs and the results obtained using nevMOGA are presented. Finally, the conclusions are presented in Section VI.

II. MULTI-OBJECTIVE BACKGROUND

A multi-objective optimization problem can be defined as follows:[1]

$$\min_{\boldsymbol{x}\in Q} \boldsymbol{f}(\boldsymbol{x}) \tag{1}$$

where $\boldsymbol{x} = [x_1, \ldots, x_k]$ is defined as a decision vector in the domain $Q \subset \Re^k$ and $\boldsymbol{f}: Q \to \Re^m$ is defined as the vector of objective functions $\boldsymbol{f}(\boldsymbol{x}) = [f_1(\boldsymbol{x}), \ldots, f_m(\boldsymbol{x})]$. The domain Q is defined by the set of constraints on $\boldsymbol{x}$. For instance (but not limited to[2]):

$$\underline{x_i} \leq x_i \leq \overline{x_i}, \quad i = [1, \ldots, k] \tag{2}$$

$\underline{x_i}$ and $\overline{x_i}$ are the lower and upper bounds of $\boldsymbol{x}$ components.

An MOP obtains, as a result, a set of optimal solutions (Pareto set, P_Q). P_Q contains solutions which are nondominated by any other.

Definition 1 (Dominance [34]): A decision vector $\boldsymbol{x^1}$ is dominated by any other decision vector $\boldsymbol{x^2}$ if $f_i(\boldsymbol{x^2}) \leq f_i(\boldsymbol{x^1})$ for all $i \in [1, \ldots, m]$ and $f_j(\boldsymbol{x^2}) < f_j(\boldsymbol{x^1})$ for at least one j, $j \in [1, \ldots, m]$. This is denoted as $\boldsymbol{x^2} \preceq \boldsymbol{x^1}$.

Definition 2 (Pareto Set P_Q): is the set of solutions in Q that is nondominated by another solution in Q:

$$P_Q := \{\boldsymbol{x} \in Q | \nexists \boldsymbol{x'} \in Q : \boldsymbol{x'} \preceq \boldsymbol{x}\}$$

Definition 3 (Pareto Front $\boldsymbol{f}(P_Q)$): given a set of Pareto optimal solutions P_Q, the Pareto front is defined as:

$$\boldsymbol{f}(P_Q) := \{\boldsymbol{f}(\boldsymbol{x}) | \boldsymbol{x} \in P_Q\}$$

In addition to Pareto set solutions, there are a set of nearly optimal solutions ($P_{Q,\epsilon}$ Definition 5) that might be interesting for the designer.

Definition 4 ($-\epsilon$-Dominance [35]): define $\boldsymbol{\epsilon} = [\epsilon_1, \ldots, \epsilon_m]$ as the maximum acceptable performance degradation. A decision vector $\boldsymbol{x^1}$ is $-\epsilon$-dominated by another decision vector $\boldsymbol{x^2}$ if $f_i(\boldsymbol{x^2}) + \epsilon_i \leq f_i(\boldsymbol{x^1})$ for all $i \in [1, \ldots, m]$ and $f_j(\boldsymbol{x^2}) + \epsilon_i < f_j(\boldsymbol{x^1})$ for at least one j, $j \in [1, \ldots, m]$. This is denoted by $\boldsymbol{x^2} \preceq_{-\epsilon} \boldsymbol{x^1}$.

Definition 5 (Set of Nearly Optimal Solutions, $P_{Q,\epsilon}$ [15]): is the set of solutions in Q which are not $-\epsilon$-dominated by another solution in Q:

$$P_{Q,\epsilon} := \{\boldsymbol{x} \in Q | \nexists \boldsymbol{x'} \in Q : \boldsymbol{x'} \preceq_{-\epsilon} \boldsymbol{x}\}$$

Nearly optimal solutions that differ significantly from the optimal (in the parameters space) are potentially useful alternatives. We define potentially useful alternatives as nearly optimal solutions nondominated in their neighborhood ($P_{Q,n}$ Definition 8). With them, the designer can obtain greater diversity which translates into better informed decision making.

Definition 6 (Neighborhood): define $\boldsymbol{n} = [n_1, \ldots, n_k]$ as the maximum distance between neighboring solutions. Two decision vectors $\boldsymbol{x^1}$ and $\boldsymbol{x^2}$ are neighboring solutions ($\boldsymbol{x^1} =_n \boldsymbol{x^2}$) if $|x_i^1 - x_i^2| < n_i$ for all $i \in [1, \ldots, k]$.

Definition 7 ($n-$Dominance): a decision vector $\boldsymbol{x^1}$ is $n-$dominated by another decision vector $\boldsymbol{x^2}$ if they are neighboring solutions (Definition 6) and $\boldsymbol{x^2} \preceq \boldsymbol{x^1}$. This is denoted by $\boldsymbol{x^2} \preceq_n \boldsymbol{x^1}$.

Definition 8 (Set of Nearly Optimal Solutions Nondominated in Their Neighborhood $P_{Q,n}$ [14]): is the set of solutions of $P_{Q,\epsilon}$ which are not $n-$dominated by another solution in $P_{Q,\epsilon}$:

$$P_{Q,n} := \{\boldsymbol{x} \in P_{Q,\epsilon} | \nexists \boldsymbol{x'} \in P_{Q,\epsilon} : \boldsymbol{x'} \preceq_n \boldsymbol{x}\}$$

Obtaining the defined sets (P_Q, $P_{Q,\epsilon}$ and $P_{Q,n}$) is often inaccessible computationally. This is because they can contain infinite solutions. Therefore, we usually look for discrete sets P_Q^*, $P_{Q,\epsilon}^*$ and $P_{Q,n}^*$ that adequately characterize P_Q, $P_{Q,\epsilon}$ and $P_{Q,n}$ respectively.

Figure 1 shows an example of an MOP. There is a set of optimal solutions (SET_1 and SET_3) located in $neighborhood_1$ and $neighborhood_2$. In addition, there is a set of nearly optimal solutions (gray area). Both sets form $P_{Q,\epsilon}$. Among them, as already mentioned, potentially useful solutions are those nondominated in their neighborhood. In this example, these alternatives are SET_1, SET_2, SET_3 and SET_4. It can be seen that there are nearly optimal alternatives that are significantly different from the optimal ones ($neighborhood_3$). A knowledge of this neighborhood gives the designer greater diversity and enables a better informed final decision to be made. In this article, for obtaining potentially useful solutions, we use the algorithm nevMOGA [14] which will be described in Section III.

III. nevMOGA ALGORITHM

nevMOGA [14] is an evolutionary algorithm that provides, in addition to optimal solutions, nearly optimal solutions nondominated in their neighborhood ($P_{Q,n}^*$). This algorithm is based on the ev-MOGA algorithm [36]. nevMOGA has four populations: $P(t)$; $Front(t)$; $SubFront(t)$; and $G(t)$. $P(t)$ is the main population and has $Nind_P$ individuals (user-defined). This population should converge towards $P_{Q,n}$ in a distributed manner. $Front(t)$ is the file where a discrete approximation of the Pareto set (P_Q^*). $SubFront(t)$ contains a discrete approximation of the nearly optimal solutions nondominated in their neighborhood ($P_{Qn-Q}^* = P_{Q,n}^* \setminus P_Q^*$). The size of these two populations is variable but linked to the number of boxes defined by the user ($\boldsymbol{n\_box}$). $G(t)$ is an auxiliary population and has $Nind_G$ individuals (user-defined taking into account that it must be a multiple of 4). This population stores the new individuals generated in each iteration.

[1] A maximization problem can be converted into a minimization problem. For each objectives to be maximized $\max f_i(\boldsymbol{x}) = -min(-f_i(\boldsymbol{x}))$will be performed.

[2] Any other constraints could be introduced in a general MOP

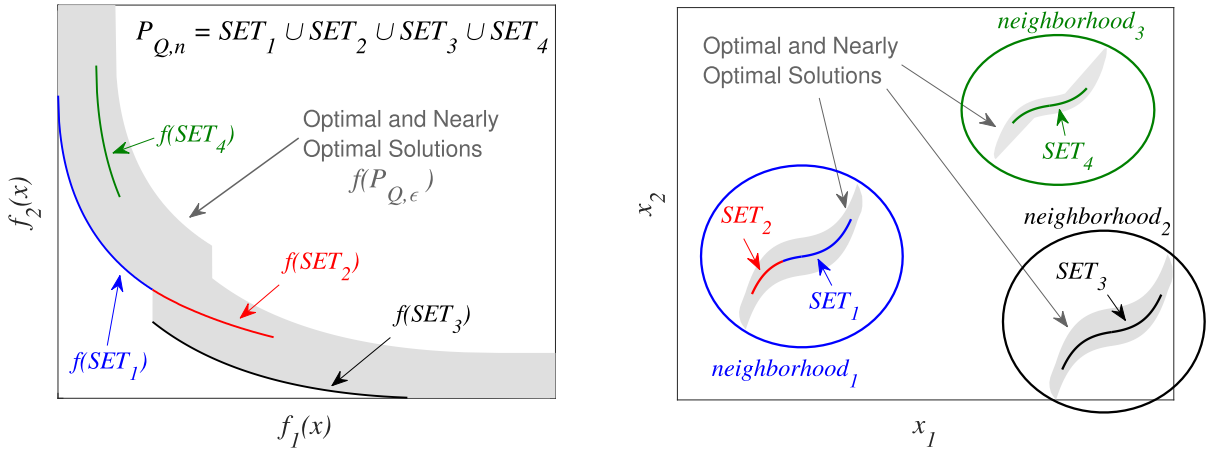


FIGURE 1. Visualization of an MOP in the objective space (on the left) and parameter space (on the right). The sets SET_1 and SET_3 are the optimal solutions. SET_2 and SET_4 are the nearly optimal solutions nondominated in their neighborhood.

Algorithm 1 shows the main pseudocode of nevMOGA (see all details in [14]). Initially, the empty $Front(t)$ and $SubFront(t)$ sets are created. Subsequently, the population $P(t)$ is created randomly (the user may establish all or part of this initial population). The inclusion of individuals of $P(t)$ in the $Front(t)$ and $SubFront(t)$ populations are then checked. In each iteration, the following steps are then performed. First, the population $G(t)$ is created. Individuals from this population are created by crossing and mutating individuals from $Front(t)$, $SubFront(t)$ and $P(t)$ populations. The individuals of $P(t)$ are chosen taking into account density (sparsely populated solutions are preferred). In this way, we encourage the evolution of less populated areas. Second, the inclusion of $G(t)$ individual in the $Front(t)$ and $SubFront(t)$ populations are checked. Finally, the population $P(t)$ is updated.

Algorithm 1 Main Pseudocode of nevMOGA

1: $t := 0$;
2: $Front(t) := \emptyset$; $SubFront(t) := \emptyset$;
3: Create initial population $P(t)$ randomly
4: Include individuals from $P(t)$ in $Front(t)$ if applicable
5: Include individuals from $P(t) \notin Front(t)$ in $SubFront(t)$ if applicable
6: **for** $t := 1$:Number of iterations **do**
7: Create population $G(t)$
8: Include individuals from $G(t)$ in $Front(t)$ if applicable
9: Include individuals from $G(t) \notin Front(t)$ in $SubFront(t)$ if applicable
10: Update $P(t)$ with individuals from $G(t)$
11: **end for**

For the use of nevMOGA it is necessary to define some parameters (see Table 1), and some have a default value in the implementation. Two of these parameters depend on problems and are especially important for a useful characterization of nearly optimal set nondominated in their neighborhood. First, parameter $\boldsymbol{\epsilon}$ is required to define the maximum acceptable objective degradation; second, parameter $\boldsymbol{n}$ defines the size of the neighbourhood. If the knowledge necessary for its definition is unavailable, there is a simple procedure for calculating this parameter from a reference solution [14]. Furthermore, it is recommended to define the population sizes $P(t)$ ($Nind_P$) and $G(t)$ ($Nind_{GA}$), as well as the number of generations to be carried out ($Generations$). If these values are not defined the nevMOGA will use default values. The parameter ***n_box*** defines the spread of the Pareto front in the objective space by dividing each dimension of the objective space into as many boxes as desired. Only one Pareto optimal solution can exist in each box. The definition of an initial population (***subpopIni***) is optional and randomly defined if not supplied.

TABLE 1. Parameters of nevMOGA.

Parameter	Description	Default
$\boldsymbol{\epsilon}$	Maximum degradation	–
$\boldsymbol{n}$	Neighborhood	Explained in [14]
$Nind_{GA}$	Population size $G(t)$	8
$Nind_P$	Population size $P(t)$	100
$Generations$	Number of generations	100
n_box	Division for each dimension	50
subpobIni	Initial population (optional)	–

IV. USEFULNESS OF THE $P_{Q,n}$ SET

As mentioned, nearly optimal alternatives in an MOP solution can be useful to the designer. However, its usefulness is even more relevant to problems where:

1) There are design objectives formed by the aggregation of partial objectives.
2) There are objectives not contemplated in the optimization phase that are incorporated in the decision phase (due to the designer's decision to reduce computational resources, or to help in the final decision).

In this section, we are going to carry out a detailed analysis of these mentioned problems, where they can appear in model

identification problems, and why nearly optimal solutions can supply useful information. For this analysis, the benchmark defined in [13] will be used because even if it is not derived from a model identification problem it helps show the potential of nearly optimal solutions. In the next section, a real identification problem is studied.

The benchmark is defined as:

$$\min_{\boldsymbol{x}} \boldsymbol{f}(\boldsymbol{x}) \tag{3}$$

$$\begin{aligned} \boldsymbol{f}(\boldsymbol{x}) &= [f_1(\boldsymbol{x}), f_2(\boldsymbol{x})] \\ f_1(\boldsymbol{x}) &= (x_1 - t_1(c+2a)+a)^2 + (x_2 - t_2 b)^2 + \delta_t \\ f_2(\boldsymbol{x}) &= (x_1 - t_1(c+2a)-a)^2 + (x_2 - t_2 b)^2 + \delta_t \end{aligned} \tag{4}$$

where

$$t_1 = sgn(x_1) min\left(\left\lceil \frac{|x_1| - a - c/2}{2a+c} \right\rceil, 1\right)$$

$$t_2 = sgn(x_2) min\left(\left\lceil \frac{|x_2| - b/2}{b} \right\rceil, 1\right)$$

and

$$\delta_t = \begin{cases} 0 & for\ t_1 = 0\ and\ t_2 = 0 \\ 0.1 & else \end{cases}$$

subject to:

$$\begin{aligned} \underline{\boldsymbol{x}} &= [-8\ \ -8] \\ \overline{\boldsymbol{x}} &= [8\ 8] \end{aligned}$$

using $a = 0.5$, $b = 5$ and $c = 5$.

This MOP contains one global Pareto set:

$$P_{0,0} = [-0.5, 0.5]\text{x}\{0\} = P_Q \tag{5}$$

as well as the following eight local Pareto sets:

$$\left.\begin{aligned} P_{-1,-1} &= [-6.5, -5.5]\text{x}\{-5\} \\ P_{0,-1} &= [-0.5, 0.5]\text{x}\{-5\} \\ P_{1,-1} &= [5.5, 6.5]\text{x}\{-5\} \\ P_{-1,0} &= [-6.5, -5.5]\text{x}\{0\} \\ P_{1,0} &= [5.5, 6.5]\text{x}\{0\} \\ P_{-1,1} &= [-6.5, -5.5]\text{x}\{5\} \\ P_{0,1} &= [-0.5, 0.5]\text{x}\{5\} \\ P_{1,1} &= [5.5, 6.5]\text{x}\{5\} \end{aligned}\right\} P_{Qn-Q} \tag{6}$$

Once the benchmark has been defined, it is necessary to define the parameters of nevMOGA $\boldsymbol{\epsilon}$ (maximum acceptable degradation) and $\boldsymbol{n}$ (neighborhood). By knowing the global and local Pareto set, the definition of these parameters is easier. Since the design objectives lack physical meaning, we define $\boldsymbol{\epsilon} = [0.15\ 0.15]$ so that the eight local Pareto sets in the objective space ($f(P_{Qn-Q})$) are found in the nearly optimal set. Secondly, we define $\boldsymbol{n}$ applying the process described in [14] with the defined $\boldsymbol{\epsilon}$ parameter and the reference solution $x^R = [0\ \ 0]$. Thus, $\boldsymbol{n} = [0.13\ \ 0.38]$ is defined. With these parameters, the set $P_{Q,n}$ (set of nearly optimal solutions nondominated in their neighborhood) is defined as:

$$\begin{aligned} P_{Q,n} := P_Q \cup P_{Qn-Q} := P_{0,0} \cup P_{-1,-1} \cup P_{0,-1} \cup \\ P_{1,-1} \cup P_{-1,0} \cup P_{1,0} \cup P_{-1,1} \cup P_{0,1} \cup P_{1,1} \end{aligned} \tag{7}$$

A. AGGREGATION OF OBJECTIVES

In systems identification problems, an experimental test is performed to obtain the dynamic behaviour of the process. In nonlinear process modelling, this test is usually made up of multi-step input signals to drive the process through the various operation points that the model must predict.

Design objectives usually measure the output prediction error (differences between model behavior and experimental test) throughout the defined identification test. Thus, each design objectives is actually formed by an aggregation of the errors at the different operation points covered by the test. Especially, in nonlinear systems where the complexity of the dynamic behavior means that the mathematical models have unmodeled dynamics, and the models with a good fit for a given operation point obtain a poorer behavior on different operation points. This problem is even worse in multivariable systems, the problem is increased by the number of variables.

Suppose we want to identify a system that has two outputs and the designer has defined an identification test that covers three different operation points. A simplified MOP where two design objectives to minimize are defined to quantify the error on each of the two outputs of the system. In this scenario, an aggregation occurs in the design objectives. Each one, in reality, is formed by the aggregation of the error on each of the three operation points. To avoid this aggregation, each operating point and variable should be evaluated separately with dedicated objectives and test. However, this approach increase the MOP size and complexity.

It is usual to obtain significantly different models with similar values in design objectives. For example, a model with excellent performance on the first operation point could have a worse performance on the third operation point, another model with poor performance on the first operation point could have an excellent performance on the second operation point and so on. The main problem is that in the simplified MOP (obtained by aggregation of objectives) most of these alternatives could disappear from the Pareto set because of small differences in the design objectives. Many of these solutions could have been members of the Pareto set if the MOP had not been simplified.

From the point of view of optimization, the aggregation of partial objectives generates multimodalities and nearly optimal non-neighboring solutions. Revealing some of these hidden solutions can provide potentially useful alternatives for the designer at the decision stage.

Let's analyze the case of aggregated partial objectives through the *benchmark* defined previously. The $P_{Q,n}$ defined in the equation (7) is the set of optimal and nearly optimal solutions nondominated in their neighborhood. To analyze the aggregated partial objectives, we now define the problem with a single design objective $f(\boldsymbol{x})$, where the two objectives previously defined (now playing the role of partial objectives) in the *benchmark* are added:

$$\begin{aligned} &\min_{\boldsymbol{x}} f(\boldsymbol{x}) \\ &f(\boldsymbol{x}) = f_1(\boldsymbol{x}) + f_2(\boldsymbol{x}) \end{aligned} \tag{8}$$

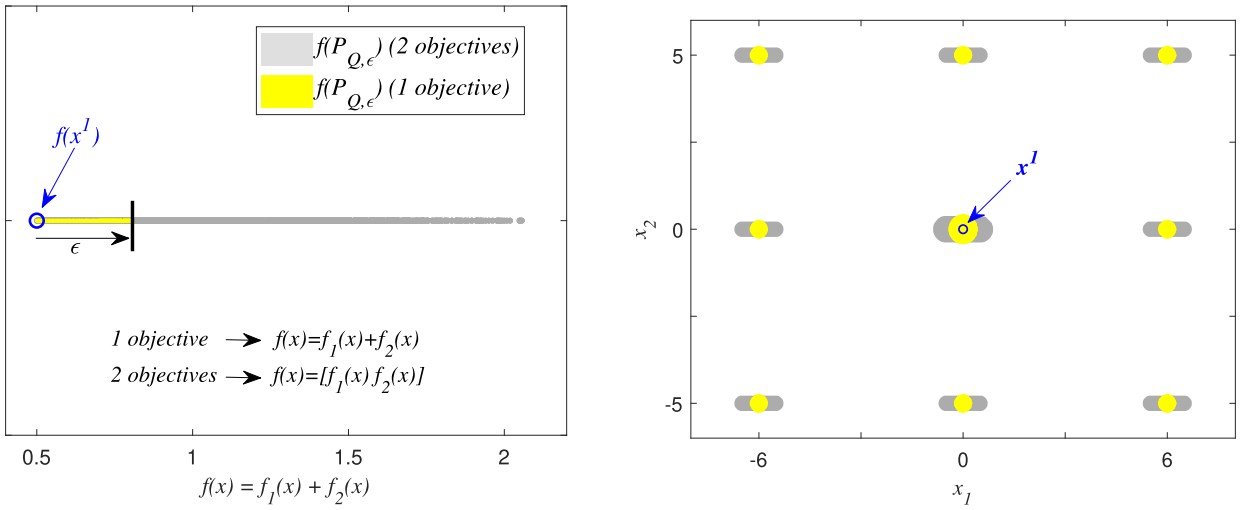


FIGURE 2. *Benchmark* solved as a single objective problem where $f(x) = f_1(x) + f_2(x)$. x^1 is the optimal solution to the problem and the yellow area is the nearly optimal area $P_{Q,\epsilon}$ with the new approach. The gray area corresponds to the projection of the nearly optimal solutions $P_{Q,\epsilon}$ of the original MOP of two objectives, on the objective $f(x)$.

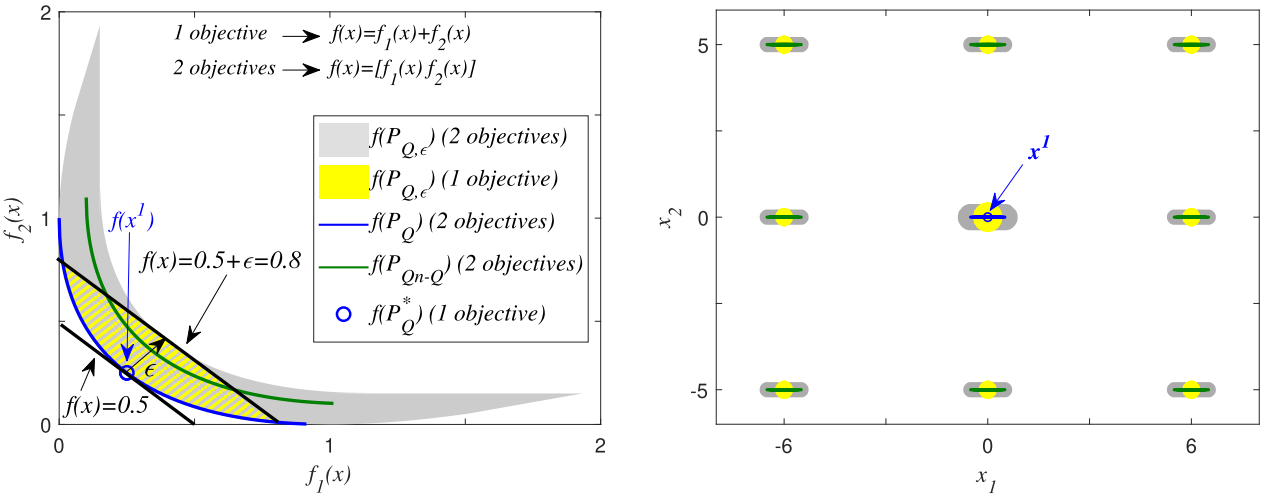


FIGURE 3. Comparison of the set $P_{Q,\epsilon}$ defined on the problem with a single objective ($f(x) = f_1(x) + f_2(x)$) and multi-objective ($f(x) = [f_1(x)\ f_2(x)]$).

In this problem $\boldsymbol{x^1}$ is the optimal solution. Figure 2 shows this optimal solution and some additional information. Many solutions from the original Pareto front (MOP with two design objectives) become nearly optimal solutions in the new scenario. For instance, with $\epsilon = 0.3$ the nearly optimal set ($P_{Q,\epsilon}$) is the yellow area. Additionally, in gray, the projection of $P_{Q,\epsilon}$ of the original 2D MOP is shown. Therefore, if only the optimal set is obtained (in this case $\boldsymbol{x^1}$), many potentially useful alternatives would be lost. By obtaining the nearly optimal solutions nondominated in their neighborhood some of this missed information can be recovered.

Figure 3 shows the zone of nearly optimal solutions $P_{Q,\epsilon}$ for the scenario of one objetive – but on the plane (f_1, f_2). This area in the objective space is the zone defined between the lines $f_1(\boldsymbol{x}) + f_2(\boldsymbol{x}) = f(\boldsymbol{x^1})$ and $f_1(\boldsymbol{x}) + f_2(\boldsymbol{x}) = f(\boldsymbol{x^1}) + \epsilon$ with $\epsilon = 0.3$ (ϵ chosen to respect the area $P_{Q,\epsilon}$ defined for the original MOP gray area).

The neighborhood of the problem is the same as in the original MOP ($\boldsymbol{n} = [0.13\ \ 0.38]$). The neighborhood of $\boldsymbol{x^1}$ is established (see Figure 4). Following our approach, the solution neighbors to $\boldsymbol{x^1}$ are dispensable solutions (similar to $\boldsymbol{x^1}$ but with a worse objective value). Of the remaining nearly optimal solutions, the $\boldsymbol{x^2}$ and $\boldsymbol{x^3}$ solutions are the best solutions (solutions with the lowest value of $f(\boldsymbol{x})$, see Figure 4).

This iterative process enables obtaining our set of interest $P^*_{Q,n}$ for the proposed MOP (see Figure 5). As can be seen, the set $P^*_{Q,n}$ obtained for the MOP with a single design objective ($f(\boldsymbol{x}) = f_1(\boldsymbol{x}) + f_2(\boldsymbol{x})$), has managed to characterize to a great extent the set $P_{Q,n}$ of the original MOP

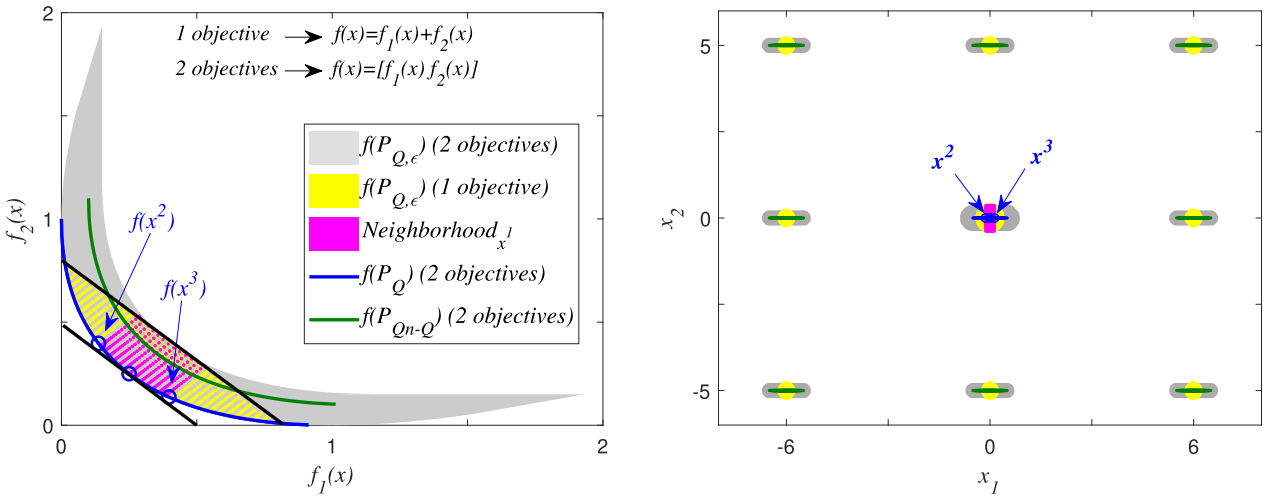


FIGURE 4. Procedure for obtaining the discrete set $P^*_{Q,n}$.

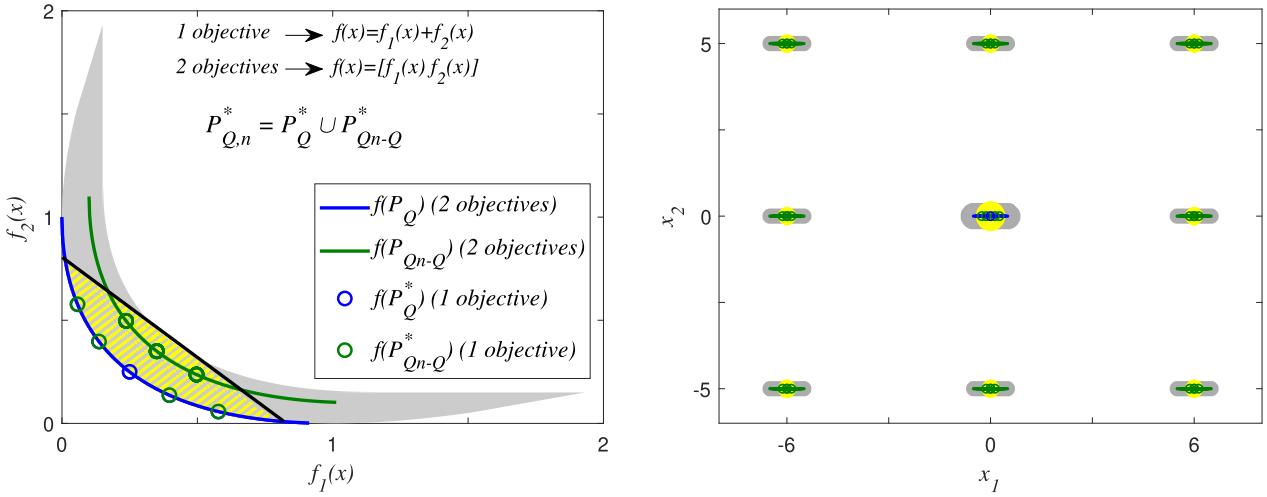


FIGURE 5. Discrete set $P^*_{Q,n} = P^*_Q \cup P^*_{Qn-Q}$ with $\epsilon = 0.3$ and $n = [0.13\ 0.38]$ obtained for the *benchmark* with a aggregated partial objectives. Despite the aggregation, the ideal set $P_{Q,n} = P_Q \cup P_{Qn-Q}$ defined for the problem with two objectives, is largely characterized.

$(f(\boldsymbol{x}) = [f_1(\boldsymbol{x})\, f_2(\boldsymbol{x})])$. In addition, this characterization could have a greater number of solutions with a smaller neighborhood. In this case, the weight of both partial objectives is the same $1 \cdot f_1$ and $1 \cdot f_2$, and therefore both have the same importance for the designer. However, the yellow area ($P_{Q,\epsilon}$ for the MOP with a single objective) can be balanced in one sense or another, modifying the weights used to add the partial objectives according to the designer's preferences (see Section IV-B for particular case of aggregation with weights $1 \cdot f_1$ and $0 \cdot f_2$).

Therefore, in an MOP when aggregating partial objectives, useful information from the original MOP (without aggregation) is lost. However, some of this information is contained in the set $P_{Q,n}$, and can be retrieved by characterizing this set.

B. EXCLUSION OF DESIGN OBJECTIVES

When carrying out the final decision of an MOP it is common to study characteristics not included in the design objectives used for optimization. These new features or objectives are not included in the optimization phase due to:

1) Designer decision: it is common to have objectives/characteristics that you do not want to influence the optimization process. This is the case, for example, of the objectives related to a validation test in a model identification problem.
2) Resource limitation: there are objectives that cannot be included in the optimization phase due to resource, time, or economic limitations. For instance, checking robustness in model identification of nonlinear systems, or validating a real plant. These new objectives can be computationally or economically very expensive.
3) Help decision making: an MOP could have a large number of objectives to consider. This fact, increases the number of optimal solutions in the Pareto front and makes it difficult to analyze. In this scenario, some may be of secondary or unknown importance.

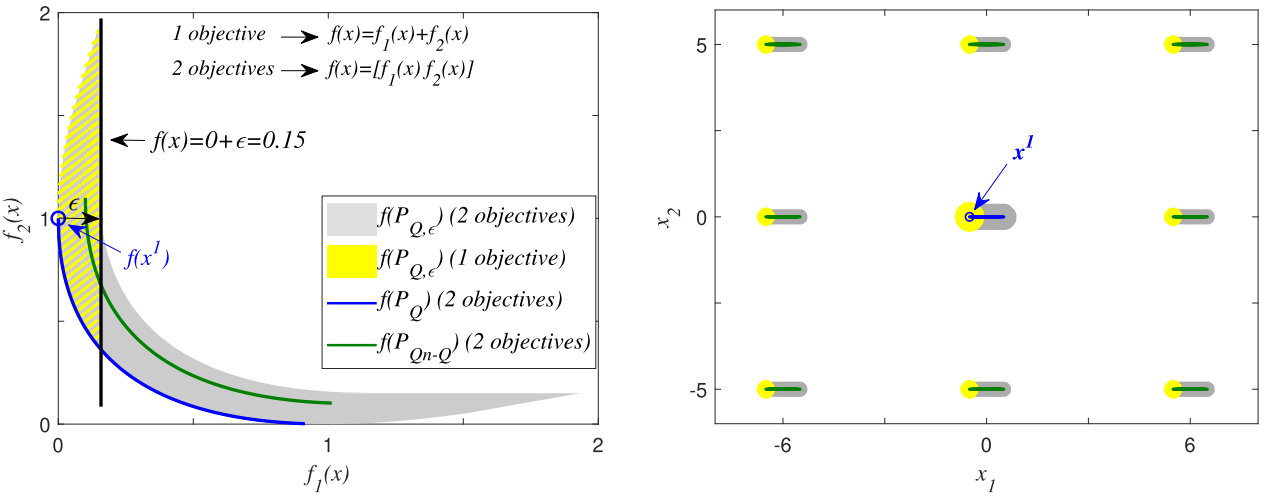


FIGURE 6. Comparison of the sets $P_{Q,\epsilon}$ on the problem with a single objective ($f(x) = f_1(x)$) and multi-objective ($f(x) = [f_1(x)\ f_2(x)]$).

In all these scenarios, analyzing these objectives in the decision phase can provide relevant information for the designer. Between two alternatives (for example, one optimal and one nearly optimal) with very similar performances in the design objectives, an improvement (which can be quite significant) in some characteristic not included in the design objectives can decide the final choice depending on designer preferences.

Suppose we have again defined the *benchmark* in Section IV. Now, a single design objective $f(\boldsymbol{x}) = f_1(\boldsymbol{x})$ is defined. The objective $f_2(\boldsymbol{x})$ does not take part in the optimization process, and its evaluation will only be considered after the optimization in the decision-making process:

$$\begin{aligned} &\min_{\boldsymbol{x}} f(\boldsymbol{x}) \\ &f(\boldsymbol{x}) = f_1(\boldsymbol{x}) \end{aligned} \tag{9}$$

Again, in this scenario we only obtain a single optimal solution ($\boldsymbol{x^1}$, see Figure 6). Therefore, if we only obtain the optimal solution we would lose a lot of useful information. In fact, there are nine neighborhoods (in the parameter space) that produce similar values of $f_1(\boldsymbol{x})$ (yellow areas, $P_{Q,\epsilon}$). All these solutions could be useful and could produce good values of the objective not included in the optimization phase ($f_2(\boldsymbol{x})$).

For comparison with the original 2D MOP, Figure 6 shows $P_{Q,\epsilon}$ and $f(P_{Q,\epsilon})$ represented on the 2D objectives space (yellow areas). This set, in the objective space, is delimited by the optimal solution $f(\boldsymbol{x^1})$ and the line $f_1(\boldsymbol{x}) = f(\boldsymbol{x^1}) + \epsilon$ with $\epsilon = 0.15$ (ϵ chosen to respect the zone $P_{Q,\epsilon}$ defined for the original MOP).

We define the same neighborhood as in the multi-objective problem ($\boldsymbol{n} = [0.13\ \ 0.38]$), and so the neighborhood of $\boldsymbol{x^1}$ is established around $\boldsymbol{x^1}$. These solutions are similar to $\boldsymbol{x^1}$ in their parameters, and have a worse objective value $f(\boldsymbol{x})$ and are therefore discarded. The solutions with the best performance from among the rest of the nearly optimal solutions are chosen. This iterative process (analogous to the process carried out in the previous section) enables obtaining the discrete set $P^*_{Q,n}$. This set can characterize, broadly, the set $P_{Q,n}$ of the original MOP with two design objectives (see Figure 7).

In $P^*_{Q,n}$ there are solutions that are worse for the design objective, but better than $\boldsymbol{x^1}$ with respect to the objective not included in the optimization phase f_2, for example $\boldsymbol{x^4}$ and $\boldsymbol{x^5}$. In fact, $\boldsymbol{x^4}$ achieves a significant improvement over f_2 in exchange for a slight loss over f_1. $\boldsymbol{x^5}$ is very different from $\boldsymbol{x^1}$ and $\boldsymbol{x^4}$ in the parameter space, and yet has a similar performance to both. Therefore, these solutions ($\boldsymbol{x^4}$ and $\boldsymbol{x^5}$) are significantly different from each other and are worthy of study as they have nearly optimal behavior on the design objective and achieve a better performance (even significantly better) for the objective not optimized $f_2(\boldsymbol{x})$.

V. IDENTIFICATION OF THE COOLING SUBSYSTEM OF A μ-CHP SYSTEM BASED ON A PEMFC

In this section, the cooling system of a PEMFC will be identified. The dynamic and nonlinear model to be identified is based on first principles. The model is completely described in [20].[3] The PEMFC cooling system to be identified is part of a real process that simulates a μ-CHP system [37], [38]. The model parameters will be adjusted and validated with experimental data.

The model inputs are (represented in red in Figure 8):

- F_a: PEMFC air flow, m^3/s.
- T_{amb}: Ambient temperature, $°C$.
- $T_{a_{in}}$: PEMFC air inlet temperature, $°C$.
- v: Voltage supplied by the PEMFC, V.
- i: Current supplied by the PEMFC, A.
- F_{w_1}: Primary circuit flow rate, m^3/s.
- F_{w_2}: Secondary circuit flow rate, m^3/s.
- R: Radiator, on/off.

[3] Available in https://riunet.upv.es/handle/10251/118336

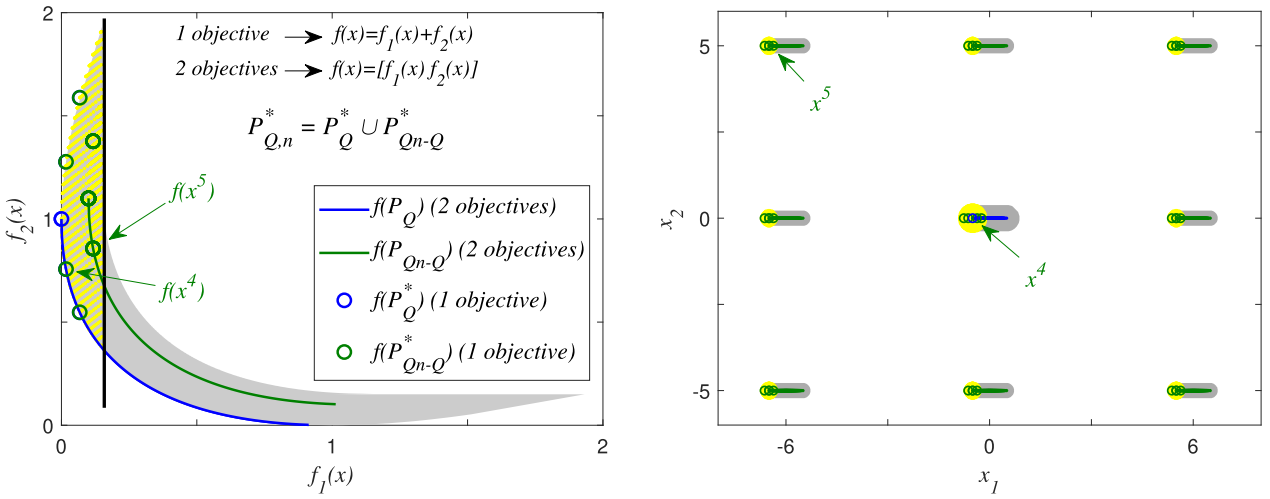


FIGURE 7. Discrete set $P^*_{Q,n} = P^*_Q \cup P^*_{Qn-Q}$ with $\epsilon = 0.15$ and $n = [0.13\ 0.38]$ with a single objective. Despite the exclusion of the optimization phase of f_2, the ideal set $P_{Q,n} = P_Q \cup P_{Qn-Q}$ defined for the problem with two objectives, is largely characterized.

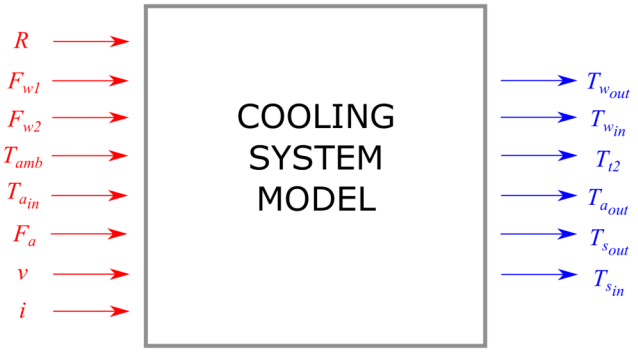


FIGURE 8. Inputs and outputs of the cooling system of a μ-CHP system.

The outputs of the model are (represented in blue in the Figure 8):

- $T_{w_{out}}$: PEMFC water outlet temperature, $^\circ C$.
- $T_{w_{in}}$: PEMFC water inlet temperature, $^\circ C$.
- T_{t2}: Temperature inside the tank 2, $^\circ C$,
- $T_{a_{out}}$: PEMFC outlet air temperature, $^\circ C$.
- $T_{s_{in}}$: Shell exchanger inlet water temperature, $^\circ C$.
- $T_{s_{out}}$: Shell exchanger outlet water temperature, $^\circ C$.

The model has 30 parameters to estimate (see Table 2, parameters described in detail in [20]). To adjust the model parameters, an identification test (approximately 2.5 h long) that was carried out on the real plant will be used (see Figure 9). In this test, steps are introduced at different inputs (electricity demand, the flows of the primary and secondary circuits of the cooling system, and demand for thermal energy). The MOP is defined as follows:

$$\min_{\boldsymbol{x}} \boldsymbol{f}(\boldsymbol{x}) = [f_1(\boldsymbol{x})\, f_2(\boldsymbol{x})\, f_3(\boldsymbol{x})] \tag{10}$$

subject to:

$$\underline{\boldsymbol{x}} \leq \boldsymbol{x} \leq \bar{\boldsymbol{x}}$$

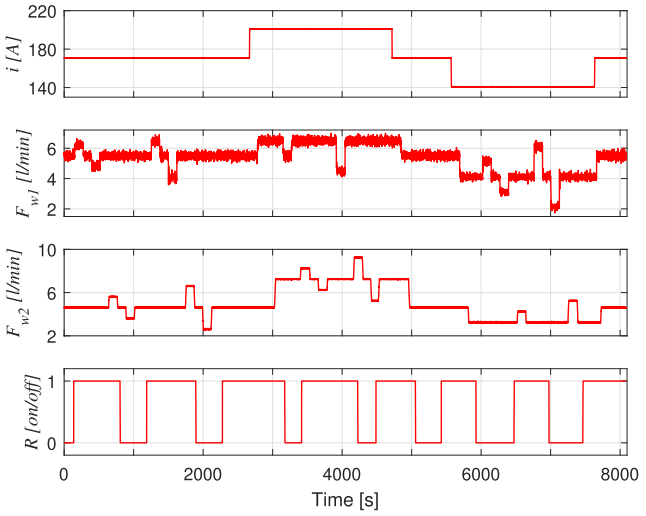


FIGURE 9. Identification test of the cooling process of a μ-CHP system.

where:

$$f_1 = \frac{1}{T}\int_0^T \left|\hat{T}_{w_{out}}(t) - T_{w_{out}}(t)\right| dt \tag{11}$$

$$f_2 = \frac{1}{T}\int_0^T \left|\hat{T}_{w_{in}}(t) - T_{w_{in}}(t)\right| dt \tag{12}$$

$$f_3 = \frac{1}{T}\int_0^T \left|\hat{T}_{t2}(t) - T_{t2}(t)\right| dt \tag{13}$$

$T = 8087s$ is the duration of the identification test, variables with circumflex accent are process outputs, variables without circumflex accent are the model outputs, $\boldsymbol{x}$ the parameter vector:

$$\begin{aligned}\boldsymbol{x} = [&V_{t1}\ h_{t_{min}}\ h_{t_{max}}\ h_{s_{min}}\ h_{s_{max}}\ V_{p1}\ h_{p1_{loss}}\ V_w\ V_a\ k_a\\ &h_{fc_{2max}}\ h_{fc_{2min}}\ h_{fc_{loss}}\ h_{a_{max}} h_{a_{min}}\ h_{w_{max}}\ h_{w_{min}}\ h_{aw}\\ &cal_{T_{w_{out}}}\ cal_{T_{a_{out}}}\ V_{p4}\ h_{p4_{loss}}\ cal_{T_{p4_{out}}}\ h_{r_{OFFmin}}\ h_{r_{OFFmax}}\\ &h_{r_{ONmin}}\ h_{r_{ONmax}}\ V_{t2}\ V_r\ T_{amb_r}]\end{aligned} \tag{14}$$

TABLE 2. Parameters of the model to estimate.

Parameter	Description
V_{t1}	Volume of water in tank 1, m^3
$h_{t_{min}}$	Minimum value of h_t (convection heat transfer parameter in the tube side of heat exchanger), W/K
$h_{t_{max}}$	Maximum value of h_t, W/K
$h_{s_{min}}$	Minimum value of h_s (convection heat transfer parameter in the tube-shell of heat exchanger), W/K
$h_{s_{max}}$	Maximum value of h_t, W/K
V_{p1}	Volume of pipe 1, m^3
$h_{p1_{loss}}$	Water-ambient heat transfer parameter in pipe 1, W/K
V_w	Volume of water in the stack, m^3
V_a	Volume of air in the stack, m^3
k_a	Product of air density and specific heat of air, $J/(m^3 \cdot K)$
$h_{fc_{2max}}$	Maximum value of h_{fc_2} (water-housing heat transfer parameter in the stack), W/K
$h_{fc_{2min}}$	Minimum value of h_{fc_2}, W/K
$h_{fc_{loss}}$	Stack-ambient heat transfer parameter, W/K
$h_{a_{max}}$	Maximum value of h_a (air side heat transfer parameter in the stack), W/K
$h_{a_{min}}$	Minimum value of h_a, W/K
$h_{w_{max}}$	Maximum value of h_w (water side heat transfer parameter in the stack), W/K
$h_{w_{min}}$	Minimum value of h_w, W/K
h_{aw}	Water-air interface heat transfer parameter in the stack, W/K
$cal_{T_{w_{out}}}$	Correction coefficient of the calibration error of $T_{w_{out}}$, K
$cal_{T_{a_{out}}}$	Correction coefficient of the calibration error of $T_{a_{out}}$, K
V_{p4}	Volume of pipe 4, m^3
$h_{p4_{loss}}$	Water-ambient heat transfer parameter in pipe 4, W/K
$cal_{T_{p4_{out}}}$	Correction coefficient of the calibration error of $T_{p4_{out}}$, K
$h_{r_{OFFmin}}$	Minimum value of $h_{r_{OFF}}$ (water-ambient heat transfer parameter in the radiator, when the radiator is off), W/K
$h_{r_{OFFmax}}$	Maximum value of $h_{r_{OFF}}$, W/K
$h_{r_{ONmin}}$	Minimum value of $h_{r_{ON}}$ (water-ambient heat transfer parameter in the radiator, when the radiator is on), W/K
$h_{r_{ONmax}}$	Maximum value of $h_{r_{ON}}$
V_{t2}	Volume of water in tank 2, m^3
V_r	Volume of water in the radiator, m^3
T_{amb_r}	Ambient temperature for the radiator, K

TABLE 3. Lower ($\underline{x}$) and upper ($\bar{x}$) limits of the cooling system parameters.

Parameter	$\underline{x}$	$\bar{x}$	Parameter	$\underline{x}$	$\bar{x}$
V_{t1}	0.001	0.003	$h_{w_{max}}$	1	200
$h_{t_{min}}$	1	100	$h_{w_{min}}$	1	150
$h_{t_{max}}$	1	200	h_{aw}	1	100
$h_{s_{min}}$	1	150	$cal_{T_{w_{out}}}$	-2	2
$h_{s_{max}}$	1	300	$cal_{T_{a_{out}}}$	-2	2
V_{p1}	0.0001	0.001	V_{p4}	0.001	0.002
$h_{p1_{loss}}$	1	15	$h_{p4_{loss}}$	1	14
V_w	0.001	0.004	$cal_{T_{p4_{out}}}$	-2	2
V_a	0.001	0.005	$h_{r_{OFFmin}}$	1	40
k_a	1000	8000	$h_{r_{OFFmax}}$	1	40
$h_{fc_{2max}}$	1	100	$h_{r_{ONmin}}$	1	100
$h_{fc_{2min}}$	1	100	$h_{r_{ONmax}}$	1	200
$h_{fc_{loss}}$	1	15	V_{t2}	0.015	0.035
$h_{a_{max}}$	1	150	V_r	0.0005	0.005
$h_{a_{min}}$	1	100	T_{amb_r}	15	35

and $\underline{\boldsymbol{x}}$ and $\bar{\boldsymbol{x}}$ (see Table 3) the lower and upper limits of the parameter vector $\boldsymbol{x}$.

The model have six outputs. An MOP with six objectives (based on the error throughout of the identification test on each of the six outputs) complicates the optimization and decision stages because of the large number of Pareto optimal models. However, analyzing the system, there are three outputs considered to be essential. These outputs are the water temperatures: inlet and outlet of the stack ($T_{w_{in}}$ and $T_{w_{out}}$ respectively), and tank 2 (T_{t2}). For this reason, we consider three design objectives to measure the error on the identification test for these outputs. However, the error on the secondary outputs (the outlet air temperature of the stack $T_{a_{out}}$, and inlet $T_{s_{in}}$ and outlet temperatures of the heat exchanger shell $T_{s_{out}}$) can also be interesting when faced with models with similar performance in the design objectives. Therefore, the errors in these outputs are calculated in the objectives f_4, f_5 and f_6 (see Equation 15). In this way, they can be analyzed in the decision phase although they are not included in the optimization stage.

$$
\begin{aligned}
f_4 &= \frac{1}{T}\int_0^T \left|\hat{T}_{a_{out}}(t) - T_{a_{out}}(t)\right| dt \\
f_5 &= \frac{1}{T}\int_0^T \left|\hat{T}_{s_{out}}(t) - T_{s_{out}}(t)\right| dt \\
f_6 &= \frac{1}{T}\int_0^T \left|\hat{T}_{s_{in}}(t) - T_{s_{in}}(t)\right| dt
\end{aligned} \tag{15}
$$

To optimize the defined MOP, nevMOGA with the following configuration is used:

- $Nind_{GA} = 4$ (size population G)
- $Nind_P = 250$ (size population P)
- $Generations = 1000$
- $\boldsymbol{\epsilon}$ = [0.01 0.01 0.01] (objective degradation accepted)
- $\boldsymbol{n}$ = [0.0005 10 20 10 40 0.0003 4 0.0005 0.0001 1000 15 15 3 15 10 15 15 15 0.5 0.5 0.0005 4 0.2 5 7 7 10 0.002 0.0004 3] (neighborhood definition)

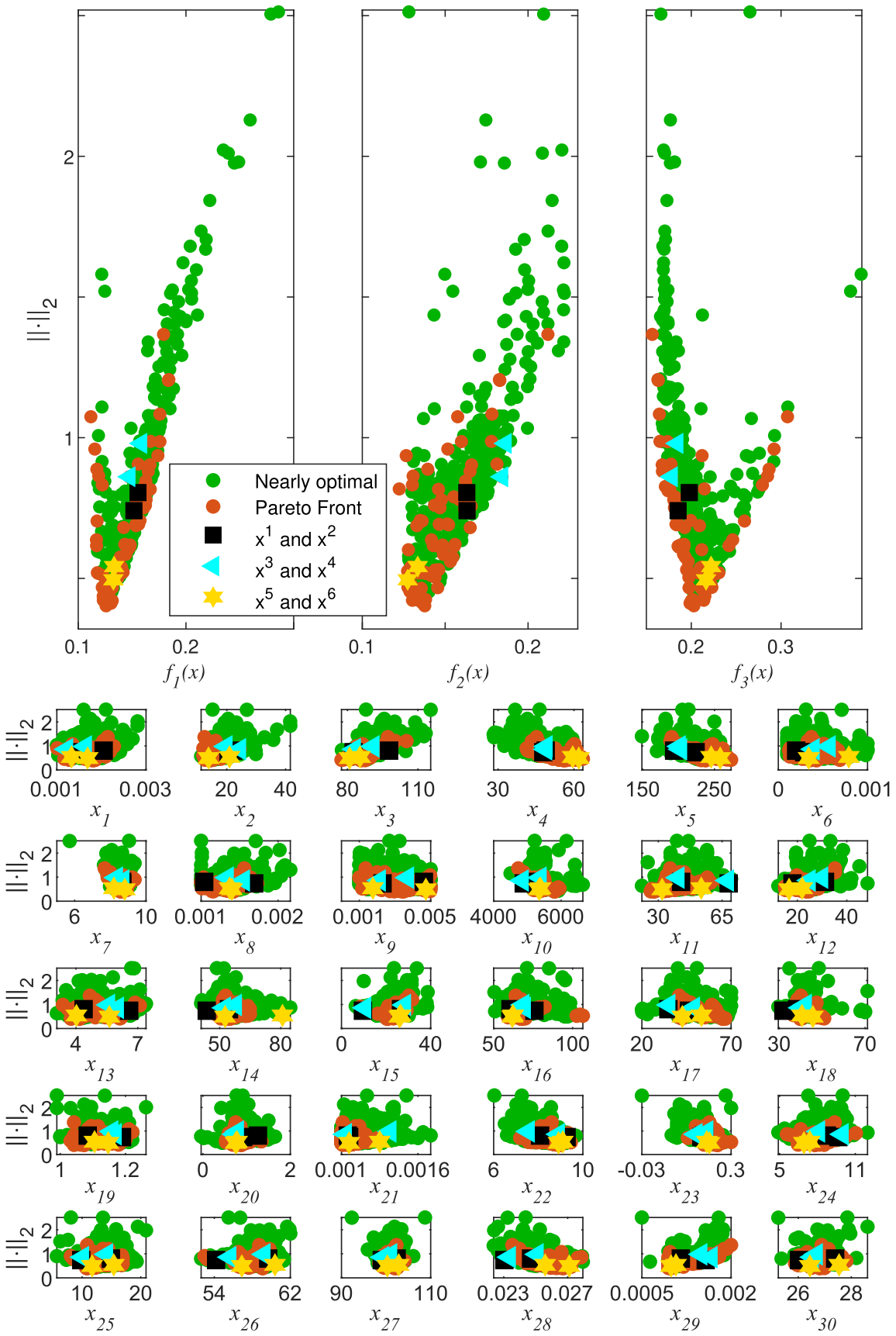


FIGURE 10. Discrete set of optimal (orange) and nearly optimal models nondominated in their neighborhood (green). The models are represented by means of the LD tool, using 2-norm ($||\cdot||_2$). Above, the objective space. Down, the decision space (x_1 to x_{30}).

TABLE 4. Models x^1, x^2, x^3, x^4, x^5 and x^6.

Parameter	x^1	x^2	x^3	x^4	x^5	x^6	Parameter	x^1	x^2	x^3	x^4	x^5	x^6
V_{t1}	**0.0014**	**0.0021**	0.0012	0.0013	0.0017	0.0013	$h_{w_{max}}$	**75.8**	**60.63**	69.42	69.51	61.8	62.19
$h_{t_{min}}$	20.62	22.19	24.34	19.75	13.81	20.93	$h_{w_{min}}$	46.24	34.84	48.92	35.1	42.68	53.37
$h_{t_{max}}$	82.6	97.72	82.93	90.84	80.93	85.34	h_{aw}	31.97	38.38	39.97	42.15	45.67	40.6
$h_{s_{min}}$	47.88	48.82	48.82	48.98	61.89	59.5	$cal_{T_{w_{out}}}$	1.19	1.09	1.17	1.16	1.15	1.11
$h_{s_{max}}$	233.67	193.81	202.82	200.88	257.96	245.36	$cal_{T_{a_{out}}}$	0.92	1.26	0.72	0.79	0.75	0.78
V_{p1}	0.0003	0.0002	0.0004	0.0006	**0.0003**	**0.0008**	V_{p4}	0.0011	0.001	0.001	0.0014	0.0011	0.0013
$h_{p1_{loss}}$	8.67	8.66	8.7	8.34	8.83	8.2	$h_{p4_{loss}}$	9.27	8.08	8.9	7.55	8.91	9.1
V_w	**0.0017**	**0.001**	0.0016	0.0013	0.0014	0.0014	$cal_{T_{p4_{out}}}$	0.19	0.2	0.17	0.21	0.22	0.21
V_a	**0.0023**	**0.0046**	**0.0021**	**0.0037**	**0.0047**	**0.0018**	$h_{r_{OFFmin}}$	9.66	8.74	9.96	8.09	6.93	7.39
k_a	4893.4	5294.7	4848.9	5379.9	5344.2	5386.6	$h_{r_{OFFmax}}$	9.86	14.9	10.07	14.17	11.71	15.4
$h_{fc_{2max}}$	**68.64**	**42.49**	**67.94**	**40.82**	**32.1**	**53.58**	$h_{r_{ONmin}}$	59.68	54.19	55.63	59.23	56.9	60.39
$h_{fc_{2min}}$	18.02	31.17	24.74	29.82	16.63	21.92	$h_{r_{ONmax}}$	99.04	102.37	99.23	101.03	100.28	102.83
$h_{fc_{loss}}$	6.58	4.37	6.01	5.58	5.62	4.02	V_{t2}	0.023	0.025	0.023	0.025	0.027	0.026
$h_{a_{max}}$	43.64	54.56	52.28	57.71	**52.66**	**80.85**	V_r	**0.0017**	**0.0012**	0.0018	0.0016	0.0011	0.0011
$h_{a_{min}}$	**9.61**	**26.69**	**10.4**	**28.24**	25.94	26.56	T_{amb_r}	26.03	27.39	26.53	26.68	26.44	27.54

In this MOP, the parameters to be estimated and the design objectives have a physical meaning. This fact makes it easy to choose the ϵ and n parameters of nevMOGA. With the defined nevMOGA configuration, the objective function is evaluated 16250 times ($Generations \cdot Nind_GA + Nind_P$). With a standard CPU,[4] one evaluation of the objective function has a computational cost of approximately 2.1 seconds (16250 evaluations have an approximate computational cost of 9.5 hours). nevMOGA obtains $P^*_{Q,n}$ at a computational cost of approximately 9.8 hours. Figure 10 shows $P^*_{Q,n}$ (discrete set of models obtained) using nevMOGA for the defined MOP. Given the great number of parameters, the level diagrams tool (LD [39], [40]) has been used for graphical representation.

The LD tool is based on level diagrams, where each design objective and parameter is represented on a separate diagram. This representation is based on two ideas: 1) classification of individuals according to their proximity to the ideal points measured with a specific norm (norm 1, norm 2, or infinite norm, in this example the norm 2 $|| \cdot ||_2$) with the normalized objectives; 2) synchronization of design objectives and parameters. In this way, each level diagram represents the objective value or parameter on the abscissa axis and its norm value on the ordinate axis. In addition, thanks to synchronization, by selecting one or more alternatives in any diagram, its representation in the remainder is automatically selected. To set designer preferences and aid decision making, the points in the level diagrams can be colored.

When analyzing the $P^*_{Q,n}$ set obtained, the objectives f_1 and f_3 conflict. That is, the models that perform better for the objective f_1, obtain worse performances for objective f_3 and vice versa. In addition, there are a many nearly optimal models (green solutions) with similar performances to the optimal ones. To compare their performance, we select three pairs of solutions in three different zones in the objective space.

[4]DELL computer, Windows 10, processor Intel Core i7-8700, 3.2 GHz and 16GB RAM.

TABLE 5. Value of the defined objectives for the models x^1, x^2, x^3, x^4, x^5 and x^6 for the identification test.

Model	f_1	f_2	f_3	f_4	f_5	f_6
x^1	**0.152**	**0.163**	**0.185**	1.52	0.274	0.250
x^2	0.155	0.163	0.198	**0.242**	**0.192**	**0.171**
x^3	**0.147**	**0.184**	**0.178**	1.02	0.258	0.225
x^4	0.158	0.186	0.184	**0.256**	**0.205**	**0.154**
x^5	**0.133**	**0.127**	**0.217**	**0.268**	0.214	0.181
x^6	0.133	0.133	0.222	0.401	**0.174**	**0.171**

First, we select the models x^1 and x^2 (represented in black in Figure 10). x^1 is an optimal model while x^2 is a nearly optimal model slightly dominated by x^1. The parameter values of these models and their objective values can be seen in Tables 4 and 5 respectively. Some parameters of these models differ significantly (represented in bold in the Table 4). For example, the parameters $h_{a_{min}}$ (air side heat transfer parameter in the stack at minimum air flow range) and V_a (volume of air inside the stack) of both models are significantly different. The model x^1 has a volume of air inside the stack significantly less than x^2. In addition, x^1 has a significantly lower heat transfer parameter $h_{a_{min}}$ than x^2. Furthermore, these models also differ significantly in the parameters V_{t1}, V_w, $h_{fc2_{max}}$, $h_{w_{max}}$ and V_r. Therefore, both models are significantly different. Figure 11 shows the temporal response of both models on the identification test.

When analyzing the response in depth, there are some time intervals where x^1 better adjusts the behavior of the process than x^2. For example, this occurs on all outputs in most of the interval [6200 7500] seconds. However, in some intervals the opposite can be observed (better behavior of the nearly optimal model x^2). For the outputs $T_{w_{out}}$ and $T_{w_{in}}$ this happens, for example, in most of the interval [0 2500] seconds. Furthermore, with respect to the output T_{t2}, this happens in the interval [3500 5000] seconds. Because the objectives contemplate the error throughout the entire experiment, this effect is masked. This is due, as previously mentioned, to the aggregation that occurs in the design objectives when adding

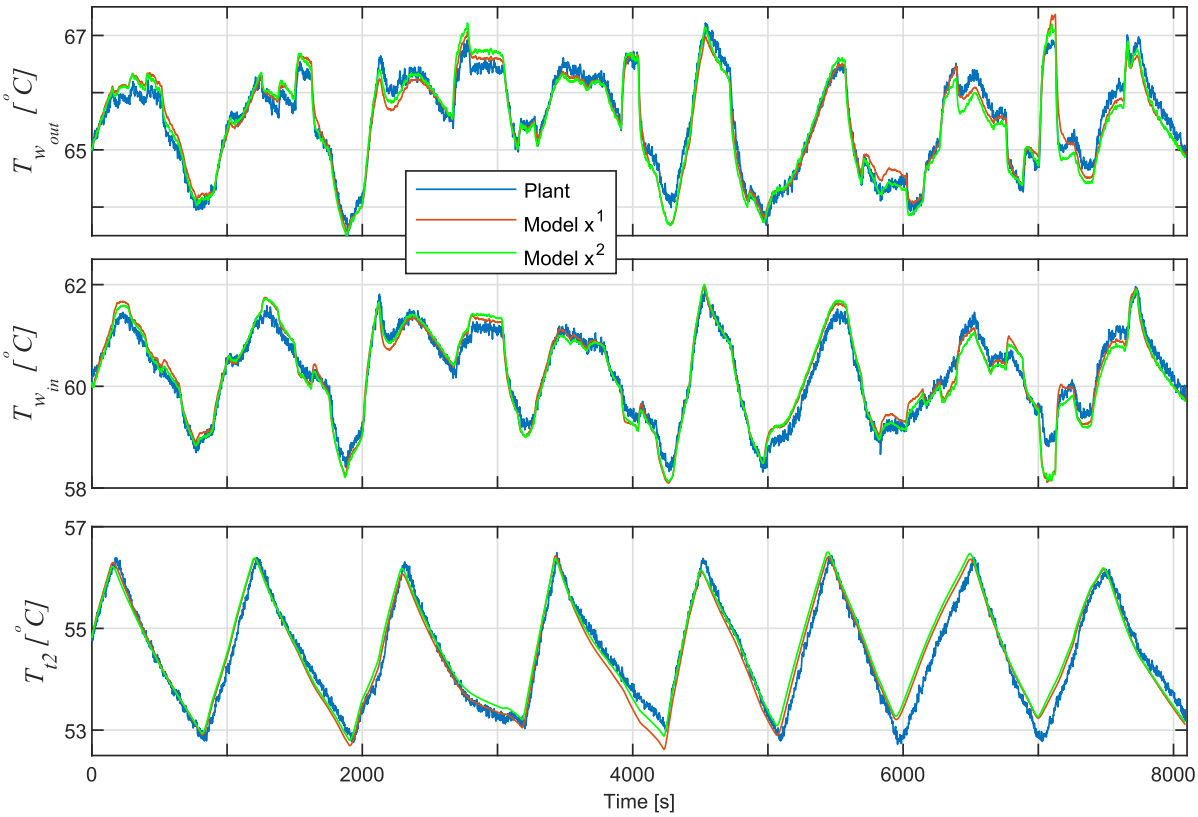


FIGURE 11. Outputs $T_{w_{out}}$, $T_{w_{in}}$ and T_{t2} obtained for the models x^1 and x^2 on the identification test. The errors of these outputs are evaluated in the design objectives (see Table ??).

the error throughout different operating points. Thus, it is shown that significantly different models can produce significant differences in different intervals of the experiment. However, they can obtain similar values of the design objectives. Therefore, the identification problem is a multimodal optimization problem. Despite this, neither model compared shows a clear advantage over the other. In addition, it is necessary to validate these models. For these reasons, we will analyze new objectives in order to make a decision with more information.

We now analyze the objectives not included in the optimization phase. Figure 12 shows the response of the models on the secondary outputs ($T_{a_{out}}$, $T_{s_{in}}$ and $T_{s_{out}}$) under the identification test. The nearly optimal model $\boldsymbol{x^2}$ obtains a smaller error in all secondary outputs than the optimal model $\boldsymbol{x^1}$ (see also the objective values f_4, f_5 and f_6 in Table 5). This improvement is not significant (but is greater than that observed on the design objectives) in the outputs $T_{s_{in}}$ and $T_{s_{out}}$. However, the improvement is significant for the output $T_{a_{out}}$ (objective f_4). In fact, in this output, the optimal model $\boldsymbol{x^1}$ has an inadequate response from 5500 to 7500 seconds (see Figure 12).

Subsequently, we defined a validation test (see Figure 13). Figure 14 shows the time response of the models on this new scenario. Table 6 shows the objective values for the validation test. As expected in the validation test, some deterioration of the fitting is obtained. Assuming this deterioration, with respect to the analyzed models, $\boldsymbol{x^2}$ obtains a smaller error than $\boldsymbol{x^1}$ on the six outputs. Among these outputs, $T_{a_{out}}$ (objective f_4) stands out again. The nearly optimal model $\boldsymbol{x^2}$ significantly improves on the optimal model $\boldsymbol{x^1}$ in this output.

Therefore, $\boldsymbol{x^1}$ and $\boldsymbol{x^2}$ are two significantly different models with very similar performance on design objectives, where $\boldsymbol{x^1}$ obtains slightly better performance. However, regarding other objectives (errors related to the secondary outputs), the model $\boldsymbol{x^2}$ achieves a better performance than $\boldsymbol{x^1}$. This improvement is significant with respect to the output $T_{a_{out}}$. Furthermore, regarding the validation test, the model $\boldsymbol{x^2}$ improves on the optimal model $\boldsymbol{x^1}$ in all the outputs. Again, this improvement is significant with respect to $T_{a_{out}}$. Therefore, between these two models, we chose the nearly optimal model $\boldsymbol{x^2}$ as the designer's preferred solution.

Repeating this analyze for two additional pairs of solutions we obtain similar results:

- For the pair of models $\boldsymbol{x^3}$ and $\boldsymbol{x^4}$. The optimal model $\boldsymbol{x^3}$ slightly dominates the nearly optimal model $\boldsymbol{x^4}$ in the design objectives (see Figure 10 and Table 5). However, again, the nearly optimal model $\boldsymbol{x^4}$ produces an improvement in the objectives not included in the optimization phase (f_4, f_5, and f_6). This improvement is quite significant in the objective f_4. Subsequently, we analyzed the objectives defined on the validation

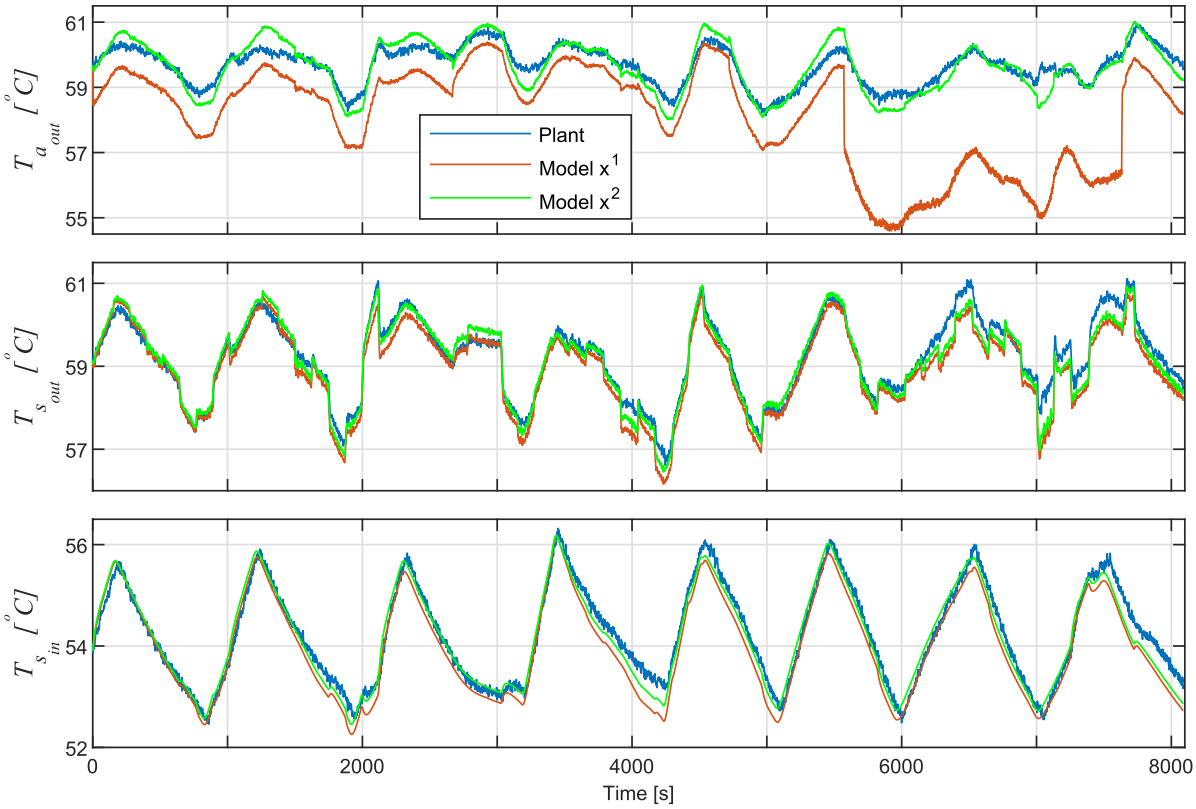


FIGURE 12. Outputs $T_{a_{out}}$, $T_{s_{in}}$ and $T_{s_{out}}$ obtained for the models x^1 and x^2 on the identification data set. The errors of these outputs are evaluated in the objectives f_4, f_5 and f_6 (see Table 5).

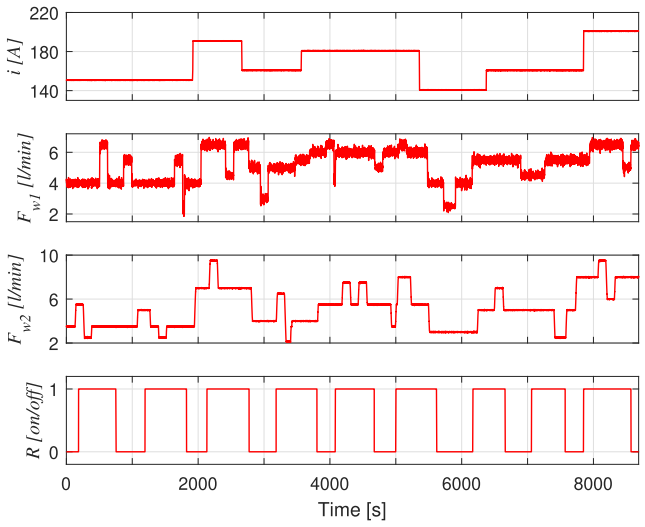


FIGURE 13. Validation test of the cooling process of a μ-CHP system.

test (see Table 6). In this scenario, the nearly optimal model $\boldsymbol{x^4}$ improves in all the defined objectives. Again, this improvement is a significant improvement over f_4. Therefore, between these two models, we chose the nearly optimal model $\boldsymbol{x^4}$ as the designer's preferred solution.

TABLE 6. Value of the defined objectives for the models x^1, x^2, x^3, x^4, x^5 and x^6 for the validation test.

Model	f_1	f_2	f_3	f_4	f_5	f_6
$\boldsymbol{x^1}$	0.873	0.771	0.55	1.95	1.02	0.84
$\boldsymbol{x^2}$	**0.856**	**0.764**	**0.447**	**0.558**	**0.802**	**0.633**
$\boldsymbol{x^3}$	0.845	0.736	0.511	1.29	0.984	0.795
$\boldsymbol{x^4}$	**0.764**	**0.632**	**0.485**	**0.578**	**0.819**	**0.631**
$\boldsymbol{x^5}$	0.813	0.693	0.451	0.693	0.879	0.668
$\boldsymbol{x^6}$	**0.771**	**0.684**	**0.423**	**0.332**	**0.777**	**0.654**

- For the pair of models $\boldsymbol{x^5}$ and $\boldsymbol{x^6}$. The optimal model $\boldsymbol{x^5}$ slightly dominates the nearly optimal model $\boldsymbol{x^6}$ in the design objectives (see Figure 10 and Table 5). In addition, the nearly optimal model $\boldsymbol{x^6}$ is better than f_5, while f_6 (errors in the outputs $T_{s_{in}}$ and $T_{s_{out}}$) is worse than f_4 (errors in the output $T_{a_{out}}$). Subsequently, we analyzed the objectives defined on the validation test (see Table 6). In this scenario, the nearly optimal model $\boldsymbol{x^6}$ improves in all the defined objectives. This improvement is significant when compared to f_4.

Therefore, with the nearly optimal solutions nondominated in their neighborhood, the designer can carry out a detailed analysis with more diverse solutions and take a final decision with better information. Also, this analysis may cause the designer to rethink the MOP. In the analyzed scenario,

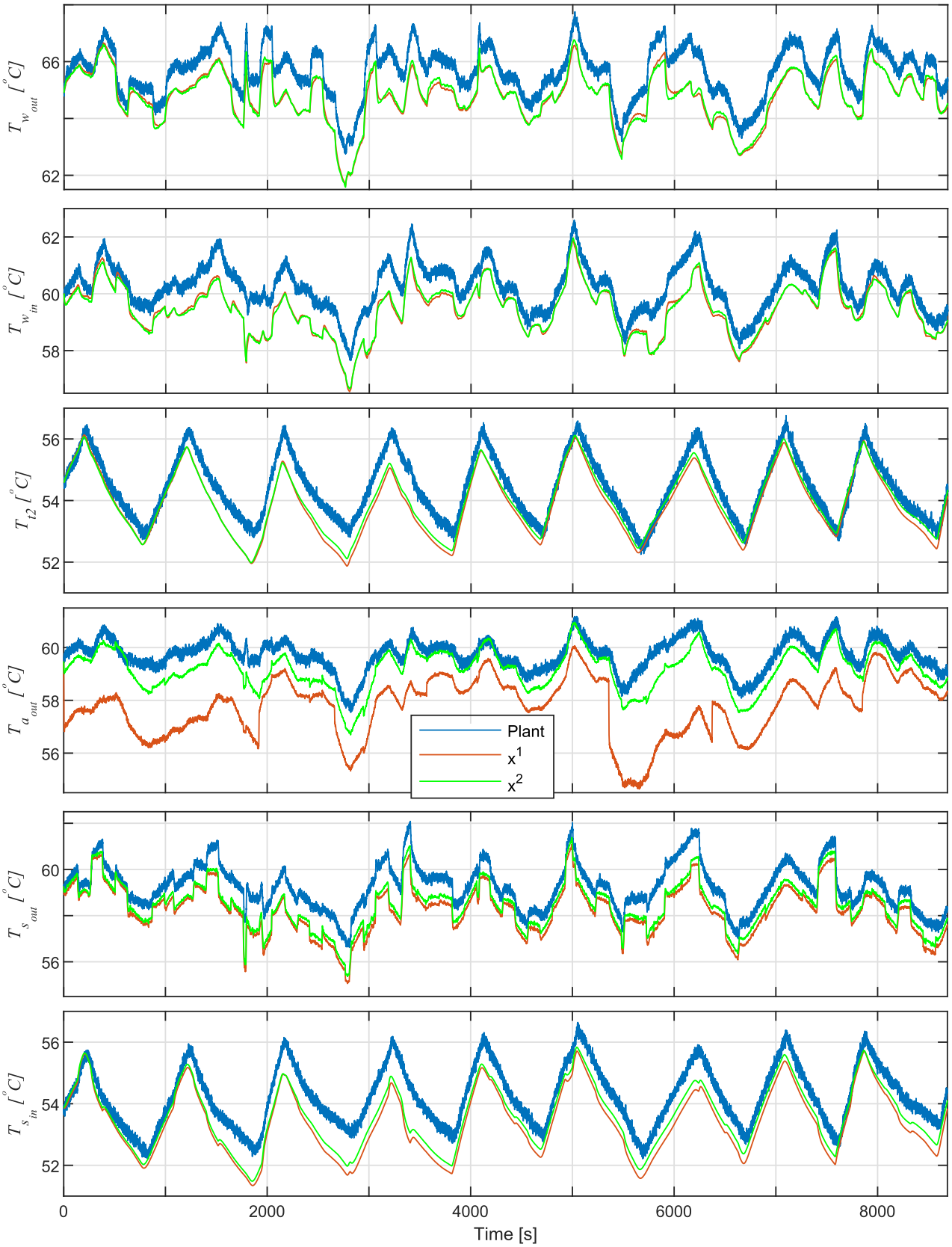


FIGURE 14. Outputs of the system obtained for the models x^1 and x^2 on the validation test. The errors of these outputs are evaluated in the objectives f_1 to f_6 (see Table 6).

the designer can now define a new MOP including a new objective in the optimization phase. For example, the objective f_4 can be included in design objectives since several optimal solutions obtain an undesirable performance on this objective. It is also possible to contemplate this objective as a constraint in a new MOP (for example, only models with $f_4 < 0.5$ will be considered). In this way, the designer could obtain only models with an acceptable response on the output $T_{a_{out}}$. In addition, in the selected models, the advantage of the nearly optimal models over the optimal ones seems clear (especially for $\boldsymbol{x^2}$ and $\boldsymbol{x^4}$ models). These models obtain a similar performances on the design objectives. Furthermore, they obtain a better performance on the analyzed objectives not included in the optimization phase (but interesting for the designer) and on the validation scenario.

VI. CONCLUSIONS

This article has presented the parametric identification of a cooling system in a PEM fuel cell using a multi-objective approach. This design has as a novelty the consideration, in addition to the optimal solutions, of the nearly optimal solutions nondominated in their neighborhood. These alternatives, ignored in a classic MOP, provide greater diversity without excessively increasing the number of alternatives to be analyzed at the decision stage.

In the system identification, the design objectives measure the errors between the prediction model and the real system for each system output. To measure this difference, the designer must define the input signals (identification test) to reach different operating points. Thus, in reality, an indirect aggregation occurs in the design objectives: aggregation of the error on each operation point that makes up the identification test. This fact, especially in nonlinear systems, causes the appearance of nearly optimal models. These models have similar performances in the design objectives but present different trade-offs in the aggregate objectives. Thus, it is common to obtain models that perform well at one operating point and obtain worse performances at another.

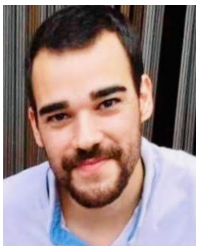
Furthermore, in the systems identification, it is necessary to validate the models obtained. Therefore, new objectives for this purpose arise naturally. In this scenario, it is possible to obtain models that, with a similar performance in the design objectives, obtain a significantly better performance in the new validation objectives.

In both scenarios, as shown in the results obtained, nearly optimal models nondominated in their neighborhood provide potentially useful new information. Thanks to the presented methodology, the designer can make a final decision with additional valuable information (ignored in the classic MOP). Therefore, this work has highlighted the relevance of taking into account the nearly optimal alternatives nondominated in their neighborhood in the identification of the multivariable system.

REFERENCES

[1] R. Russell Rhinehart, *Engineering Optimization: Applications, Methods, and Analysis*. Chichester, U.K.: Wiley, 2018.
[2] S. S. Rao, *Engineering Optimization: Theory and Practice*. Hoboken, NJ, USA: Wiley, 2020.
[3] I. Kacem, "Applications of optimization methods to industrial engineering problems," *Comput. Ind. Eng.*, vol. 112, pp. 467–468, Oct. 2017.
[4] C. Fernández, N. Pantano, S. Godoy, E. Serrano, and G. Scaglia, "Parameters optimization applying Monte Carlo methods and evolutionary algorithms. Enforcement to a trajectory tracking controller in non-linear systems," *Revista Iberoamericana de Automática e Informática Ind.*, vol. 16, no. 1, pp. 89–99, 2018.
[5] K. Miettinen, "Nonlinear multi-objective optimization," in *Erscheinungsort Nicht Ermittelbar*. New York, NY, USA: Springer, 1998.
[6] K. Deb, *Multi-Objective Optimization Using Evolutionary Algorithms*. Chichester, U.K.: Wiley, 2001.
[7] G. Reynoso-Meza, X. Blasco, J. Sanchis, and J. M. Herrero, *Controller Tuning With Evolutionary Multi-Objective Optimization a Holistic Multi-Objective Optimization Design Procedure*. Cham, Switzerland: Springer, 2018.
[8] N. Gunantara, "A review of multi-objective optimization: Methods and its applications," *Cogent Eng.*, vol. 5, no. 1, 2018, Art. no. 1502242.
[9] L. Ljung, *System Identification—Theory for the User*, 2nd ed. Upper Saddle River, NJ, USA: Prentice-Hall, 1999.
[10] P. Loridan, "ε-solutions in vector minimization problems," *J. Optim. Theory Appl.*, vol. 43, no. 2, pp. 265–276, 1984.
[11] D. J. White, "Epsilon efficiency," *J. Optim. Theory Appl.*, vol. 49, no. 2, pp. 319–337, 1986.
[12] A. Engau and M. M. Wiecek, "Generating ε-efficient solutions in multiobjective programming," *Eur. J. Oper. Res.*, vol. 177, no. 3, pp. 1566–1579, 2007.
[13] O. Schütze, M. Vasile, and C. A. C. Coello, "Approximate solutions in space mission design," in *Proc. Int. Conf. Parallel Problem Solving Nature*. Berlin, Germany: Springer, Sep. 2008, pp. 805–814.
[14] A. Pajares, X. Blasco, J. M. Herrero, and G. Reynoso-Meza, "A multiobjective genetic algorithm for the localization of optimal and nearly optimal solutions which are potentially useful: nevMOGA," *Complexity*, vol. 2018, pp. 1–22, Oct. 2018.
[15] O. Schütze, M. Vasile, and C. A. C. Coello, "Computing the set of epsilon-efficient solutions in multi-objective space mission design," *J. Aerosp. Comput., Inf., Commun.*, vol. 8, no. 3, pp. 53–70, 2011.
[16] A. Pajares, X. Blasco, J. M. Herrero, and G. Reynoso-Meza, "A new point of view in multivariable controller tuning under multiobjective optimization by considering nearly optimal solutions," *IEEE Access*, vol. 7, pp. 66435–66452, 2019.
[17] A. Pajares, "Development of a multi-objective optimization methodology considering nearly optimal solutions. Application to problems in control engineering," Ph.D. dissertation, Dept. Ingeniería Sistemas Automática, Universitat Politècnica de València, Valencia, Spain, 2019. [Online]. Available: https://riunet.upv.es/handle/10251/133974
[18] V. Mehta and J. S. Cooper, "Review and analysis of PEM fuel cell design and manufacturing," *J. Power Sources*, vol. 114, no. 1, pp. 32–53, 2003.
[19] J. Andújar and F. Segura, "Fuel cells: History and updating. A walk along two centuries," *Renew. Sustain. Energy Rev.*, vol. 13, no. 9, pp. 2309–2322, 2009.
[20] S. N. Giménez, J. M. H. Durá, X. B. Ferragud, and R. S. Fernández, "Control-oriented modeling of the cooling process of a PEMFC-based μ-CHP system," *IEEE Access*, vol. 7, pp. 95620–95642, 2019.
[21] M. M. Maghanki, B. Ghobadian, G. Najafi, and R. J. Galogah, "Micro combined heat and power (MCHP) technologies and applications," *Renew. Sustain. Energy Rev.*, vol. 28, pp. 510–524, Dec. 2013.
[22] D. A. Notter, K. Kouravelou, T. Karachalios, M. K. Daletou, and N. T. Haberland, "Life cycle assessment of PEM FC applications: Electric mobility and μ-CHP," *Energy Environ. Sci.*, vol. 8, no. 7, pp. 1969–1985, 2015.
[23] R. Beith, *Small and Micro Combined Heat and Power (CHP) Systems*. Cambridge, U.K.: Woodhead Publishing Limited, 2011.
[24] P. Martinez, M. Serra, and R. Costa-Castello, "Modeling and control of HTPEMFC based combined heat and power for confort control," in *Proc. 22nd IEEE Int. Conf. Emerg. Technol. Factory Autom. (ETFA)*, Sep. 2017, pp. 1–6.
[25] I. Staffell, "Fuel cells for domestic heat and power: Are they worth it?" Ph.D. dissertation, School Chem. Eng., Univ. Birmingham, Birmingham, U.K., 2010.
[26] G. Petrecca, *Energy Conversion and Management*. Cham, Switzerland: Springer, 2014.

[27] Q. Yan, H. Toghiani, and H. Causey, "Steady state and dynamic performance of proton exchange membrane fuel cells (PEMFCs) under various operating conditions and load changes," *J. Power Sources*, vol. 161, no. 1, pp. 492–502, 2006.
[28] S. G. Kandlikar and Z. Lu, "Thermal management issues in a PEMFC stack—A brief review of current status," *Appl. Thermal Eng.*, vol. 29, no. 7, pp. 1276–1280, 2009.
[29] A. De las Heras, F. J. Vivas, F. Segura, M. J. Redondo, and J. M. Andújar, "Air-cooled fuel cells: Keys to design and build the oxidant/cooling system," *Renew. Energy*, vol. 125, pp. 1–20, Sep. 2018.
[30] G. Zhang and S. G. Kandlikar, "A critical review of cooling techniques in proton exchange membrane fuel cell stacks," *Int. J. Hydrogen Energy*, vol. 37, no. 3, pp. 2412–2429, 2012.
[31] J.-W. Ahn and S.-Y. Choe, "Coolant controls of a PEM fuel cell system," *J. Power Sources*, vol. 179, no. 1, pp. 252–264, 2008.
[32] W. Schmittinger and A. Vahidi, "A review of the main parameters influencing long-term performance and durability of PEM fuel cells," *J. Power Sources*, vol. 180, no. 1, pp. 1–14, May 2008.
[33] Y. Xing, J. Na, and R. Costa-Castello, "Adaptive online parameter estimation algorithm of PEM fuel cells," in *Proc. 18th Eur. Control Conf. (ECC)*, Jun. 2019, pp. 441–446.
[34] V. Pareto, *Manual of Political Economy*. New York, NY, USA: Augustus M. Kelley, 1971.
[35] O. Schütze, C. A. C. Coello, and E. G. Talbi, "Approximating the ϵ-efficient set of an MOP with stochastic search algorithms," in *Proc. Mexican Int. Conf. Artif. Intell.* Berlin, Germany: Springer, Nov. 2007, pp. 128–138.
[36] J. M. Herrero, S. García-Nieto, X. Blasco, V. Romero-García, J. V. Sánchez-Pérez, and L. M. Garcia-Raffi, "Optimization of sonic crystal attenuation properties by *ev-MOGA* multi-objective evolutionary algorithm," *Struct. Multidisciplinary Optim.*, vol. 39, no. 2, pp. 203–215, 2008.
[37] T. Elmer, M. Worall, S. Wu, and S. B. Riffat, "Fuel cell technology for domestic built environment applications: State of-the-art review," *Renew. Sustain. Energy Rev.*, vol. 42, pp. 913–931, Feb. 2015.
[38] A. Hawkes, I. Staffell, D. Brett, and N. Brandon, "Fuel cells for micro-combined heat and power generation," *Energy Environ. Sci.*, vol. 2, no. 7, pp. 729–744, 2009.
[39] X. Blasco, J. Herrero, J. Sanchis, and M. Martínez, "A new graphical visualization of n-dimensional Pareto front for decision-making in multi-objective optimization," *Inf. Sci.*, vol. 178, no. 20, pp. 3908–3924, 2008.
[40] X. Blasco, J. M. Herrero, G. Reynoso-Meza, and M. A. M. Iranzo, "Interactive tool for analyzing multiobjective optimization results with level diagrams," in *Proc. Genetic Evol. Comput. Conf. Companion (GECCO)*, Jul. 2017.



ALBERTO PAJARES received the B.S. and Ph.D. degrees in automation from the Universitat Politècnica de València (UPV), Spain, in 2013 and 2019, respectively. He is currently a Researcher with the Institute of Automatic Control (ai2), UPV. His main research interests are in the control and modeling of systems, process optimization, and multiobjective optimization techniques.

F. XAVIER BLASCO (Member, IEEE) was born in Paris, in 1966. He received the B.S. degree in industrial engineering from the Universitat Politècnica de València (UPV), Spain, in 1991, the Diplome de Spécialisation en Génie Electrique from Ecole Supérieure d'Electricité (SUPELEC), France, in 1992, and the Ph.D. degree in industrial engineering from the UPV, in 1999. Since 1994, he has been teaching with the Department of Systems Engineering and Automation, UPV. He is currently a full professor. His research work is developed at the Institute of Automatic Control (ai2), UPV. His research interests are model-based predictive control, evolutionary optimization, and multi-objective optimization applied to engineering, as well as dynamic modeling and process control.

JUAN MANUEL HERRERO received the B.S. and Ph.D. degrees in control systems engineering from the Universitat Politècnica de València (UPV), in 1999 and 2006, respectively. He is an Associate Professor with the Department of Systems Engineering and Automation, UPV. His main research interests are multivariable predictive control, process optimization, and computational intelligence methods for control engineering.

JOSÉ VICENTE SALCEDO received the B.Sc. and Ph.D. degrees in industrial engineering from the Universitat Politècnica de València (UPV), Spain, in 1997 and 2005, respectively. He is currently a Professor with the Department of Systems Engineering and Control, UPV. He has published numerous papers in conference proceedings and journals. His research interests include model predictive control, fuzzy systems, and evolutionary optimization applied to identification and process control.