IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received April 18, 2019, accepted April 27, 2019, date of publication May 9, 2019, date of current version June 4, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2915556

A New Point of View in Multivariable Controller Tuning Under Multiobjective Optimization by Considering Nearly Optimal Solutions

ALBERTO PAJARES[1], XAVIER BLASCO[1], JUAN MANUEL HERRERO[1], AND GILBERTO REYNOSO-MEZA[2]

[1]Instituto Universitario de Automática e Informática Industrial, Universitat Politécnica de Valéncia, 46022 Valencia, Spain
[2]Programa de Pós-Graduação em Engenharia de Produção e Sistemas (PPGEPS), Pontifícia Universidade Católica do Paraná (PUCPR), Curitiba 80215-901, Brazil

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

This work was supported in part by the Ministerio de Economía y Competitividad (Spain) under Grant DPI2015-71443-R, Grant RTI2018-096904-B-I00, and Grant FPU15/01652, in part by the Generalitat Valenciana Regional Government through Project GV/2017/029, and in part by the National Council of Scientific and Technological Development of Brazil (CNPq) under Grant PQ2/304066/2016-8 and Grant UN/437105/2018-0.t.

ABSTRACT In this paper, we present the adjustment of controller parameters using multiobjective optimization techniques. Unlike other works, where only the Pareto optimal solutions are considered, we also consider the set of nearly optimal solutions nondominated in their neighborhood. These solutions are potentially useful for two reasons: 1) they are similar to the optimal solutions for the optimized objectives, and; 2) they differ significantly in their parameters. This last point makes them interesting, since they bring diversity and different characteristics to the set of solutions for analyzing in the decision stage. In problems of controller parameter adjustment, especially for multivariable processes, there are many conflicting objectives. To simplify the optimization problem and decision stage, it is common to aggregate some of the objectives, and so simplify the initial problem. In this scenario, some controllers that were optimal for the initial problem can become nearly optimal in the simplified case. When these controllers are nondominated in their neighborhood, they are especially interesting because they usually present a different trade-off for the initial objectives. For the calculation of nearly optimal solutions nondominated in their neighborhood, the evolutionary algorithm nevMOGA was used. In this paper, the usefulness of considering these solutions is revealed in two controller design problems: the Wood & Berry distillation column and the CIC2018 control benchmark.

INDEX TERMS Multiobjective optimization, multivariable control systems, nearly optimal solutions.

I. INTRODUCTION

Problems in many areas of engineering are solved using optimization tools [18], [34] including, for example, control system design problems [11], [37]. In this paper, the design of control systems is addressed through the optimal adjustment of the controller parameters [7] and not through the determination of the optimal control actions, which is commonly known as optimal control [47]. Typically, the controller tunning parameters problem contain conflicting objectives: performance, control effort, robustness, etc. For this reason, it is natural to pose them as multiobjective optimization problems (MOPs [18], [19], [21], [42]) for the design of control systems [7]. In an MOP, there is usually no single optimal solution, but a set of optimal solutions where none is better than the rest for all the proposed objectives [8], [27], [41]. The stages of an MOP are: problem definition, optimization process [9], and multicriteria decision stage (MCDM).

Many industrial processes include multivariable control systems. In these systems, the number of objectives increases considerably: performance of each output to be controlled, the control effort of each control action, and so on. The greater the number of objectives in MOPs, the more complex becomes the optimization process and especially the

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

decision stage. For this reason, it is common to aggregate objectives and so simplify both stages. However, this approach has drawbacks. Let us look at a specific example.

Suppose we have a multivariable system with three inputs and three outputs that we wish to control using three proportional controllers. An MOP is defined with six design goals (f_{11}, f_{12}, f_{13}. f_{21}, f_{22}, f_{23}) and three parameters (Kc_1, Kc_2 and Kc_3) for adjustment. We will call this a complete MOP. The first three objectives are the integral absolute errors (IAE) in each of the three outputs; while the last three are the integral of absolute variation of control signal (IAVU) of each of the controllers. The three parameters to adjust correspond to the gain of each of the three proportional controllers of the control system. We have four control systems (each with its three proportional controllers) as shown in Table 1, and for which we obtain the value of the design objectives (see Table 2). None of the systems is dominated by another and we will assume that all the control systems are optimal alternatives for the proposed MOP.

TABLE 1. Proposed control systems with three decision variables (Kc_1, Kc_2, and Kc_3) and design objectives for the reduced MOP (f_1 and f_2).

Controllers	Kc_1	Kc_2	Kc_3	$f_1(\boldsymbol{x})$	$f_2(\boldsymbol{x})$
$\boldsymbol{x^1}$	0.5	0.3	0.7	**1.5**	**1.5**
$\boldsymbol{x^2}$	0.75	0.5	0.25	**1.5**	**1.5**
$\boldsymbol{x^3}$	0.25	0.5	0.5	1.55	1.55
$\boldsymbol{x^4}$	0.49	0.31	0.71	1.51	1.51

TABLE 2. Value of the design objectives of the complete MOP for the control systems defined in Table 1.

Controllers	f_{11}	f_{12}	f_{13}	f_{21}	f_{22}	f_{23}
$\boldsymbol{x^1}$	0.25	0.75	0.5	0.75	0.25	0.5
$\boldsymbol{x^2}$	0.5	0.5	0.5	0.5	0.5	0.5
$\boldsymbol{x^3}$	0.75	0.5	0.3	0.25	0.5	0.8
$\boldsymbol{x^4}$	0.24	0.76	0.51	0.76	0.24	0.51

To simplify the optimization process and decision stage, the design objectives are aggregated to form two single objectives: one is the aggregation of the IAEs of all the outputs, and the other is the aggregation of the IAVUs of the control efforts. We obtain $f_1 = f_{11} + f_{12} + f_{13}$ and $f_2 = f_{21} + f_{22} + f_{23}$ to define a new MOP that we will call MOP reduced. When making this aggregation (see Table 1), $\boldsymbol{x^1}$ and $\boldsymbol{x^2}$ are optimal alternatives and have the same value in the objectives (multimodal solutions [38]). In addition, $\boldsymbol{x^3}$ and $\boldsymbol{x^4}$, which were optimal in the full MOP, are no longer optimal in the reduced MOP, despite having a performance similar to $\boldsymbol{x^1}$ and $\boldsymbol{x^2}$. These alternatives ($\boldsymbol{x^3}$ and $\boldsymbol{x^4}$) may be useful for the designer, and as a consequence of having objectives added, they have stopped being optimal solutions (becoming nearly optimal) and a traditional MO optimizer would obviate them. These solutions can continue to be considered with the use of an optimizer that maintains the set of nearly optimal solutions, that is, solutions with performances similar to the optimal solutions.

In general, finding all of the nearly optimal alternatives can considerably increase the number of solutions and this has two drawbacks: it slows down the algorithm and makes the decision stage more complex. Therefore, it is important to reduce the number of nearly optimal alternatives to consider and discard those that are irrelevant.

Let us suppose we have the four control systems as in Table 1. In this case, for the reduced MOP previously described (with design goals f_1 and f_2), we have two optimal solutions ($\boldsymbol{x^1}$ and $\boldsymbol{x^2}$) and two nearly optimal solutions ($\boldsymbol{x^3}$ and $\boldsymbol{x^4}$). $\boldsymbol{x^3}$ is a control system with parameters significantly different from $\boldsymbol{x^1}$ and $\boldsymbol{x^2}$. Therefore, there is no control system that dominates it with similar parameters, that is, there is no better control system in their neighborhood (same parameter zone). This solution gives the designer an interesting option in a different neighborhood. Maintaining it enables an analysis a posteriori (for example, when including new indicators or when considering the physical sense of the solutions), and it can be considered as a possible final solution to the detriment of $\boldsymbol{x^1}$ and $\boldsymbol{x^2}$. $\boldsymbol{x^4}$ is a control system dominated by $\boldsymbol{x^1}$ and both parameters are very similar, meaning that they are neighboring solutions. These alternatives will presumably have similar characteristics. Therefore, the designer will choose the best of the two options for the design objectives ($\boldsymbol{x^1}$). For this reason, under our criteria, $\boldsymbol{x^4}$ is a non-essential or irrelevant option.

Therefore, we consider potentially useful options to be optimal and nearly optimal solutions that are nondominated in their neighborhood ($\boldsymbol{x^1}$, $\boldsymbol{x^2}$ and $\boldsymbol{x^3}$ in the reduced MOP). These options can be useful in any MOP [1] as they produce diversity in the set found without excessively increasing the number of possible alternatives. In an MOP where objectives are aggregated, these solutions will presumably have different trade-offs among the aggregated objectives and give the designer more information at the decision stage.

In the multicriteria decision stage, it is possible to study new features that were not included in the design objectives before choosing the final control system. This procedure may be appropriate because of: the impossibility of including them in the optimization phase due to limited resources (time or economic reasons), or decision support (change of scenario, validation objectives, analysis of the design objectives). Obtaining nearly optimal solutions that are nondominated in their neighborhood enables studying these new indicators on alternatives with characteristics that differ to the optimal options. Perhaps nearly optimal solutions offer better value with respect to the new indicators than the optimal options and are therefore preferred.

For example, in the decision-making phase of the proposed MOP, the robustness of each control system found can be analyzed as a new indicator. $\boldsymbol{x^1}$, $\boldsymbol{x^2}$ and $\boldsymbol{x^3}$ are significantly different (not neighboring), and they will presumably have different characteristics and differing levels of robustness. If we assume that $\boldsymbol{x^3}$ is more robust than $\boldsymbol{x^1}$ and $\boldsymbol{x^2}$, the designer could choose $\boldsymbol{x^3}$ (nearly optimal) to the detriment

$\boldsymbol{x^1}$ and $\boldsymbol{x^2}$ (optimals). So, the nearly optimal controllers are specially relevant, since when uncertainty is considered in the controlled process, they can result more robust, while they maintain the nearly optimal response in closed loop [46].

It is clear that the design objectives included in the optimization stage, and for which the set of optimal and nearly optimal solutions nondominated in their neighborhood are obtained, are ''preferred'' to the objectives not included in the optimization phase (although these objectives can condition the final decision, they cannot condition the obtained set, in this sense we mean that they are preferred).

This paper highlights the importance in controller design problems of considering nearly optimal alternatives nondominated in their neighborhood. To obtain these alternatives the nevMOGA algorithm[1] is used [1]. By means of two controller design problems, it is shown how nearly optimal solutions can be preferable to the optimal solutions following a detailed analysis in the final stage of decision making.

In the literature, there exist other different methodologies to obtain the optimal solutions, or the nearly optimal solutions that are of interest for the designer ([44], [47]). One of them is game theory [43] based on Nash equilibrium [40]. By means of this technique, an equilibrium solution is obtained, and it is possible to gather different solutions by defining various games [44]. However, this methodology has two drawbacks: 1) it cannot be guaranteed that the Nash equilibrium solutions are optimal (there may be better solutions with respect to the design objectives that are not obtained) and 2) the set of solutions that it provides may be poorer distributed [45]. The nearly optimal alternatives can be useful for the designer. However, in order to assess their usefulness, it is necessary to compare their performance with the performance of the optimal solutions. Moreover, game theory ignores, according to our criterion, potentially useful solutions for the designer (optimal and nearly optimal solutions nondominated in their neighborhood). Consequently, the designer would choose the final solution with less information than is desired. For all these reasons, the use of this technique has not been studied in this work.

This work is structured as follows. In Section 2 some basic definitions are presented. In Section 3 the nevMOGA algorithm used in this work is described in detail. In Section 4 the problems of designing controllers as MOPs and the results obtained using nevMOGA are presented. Finally, the conclusions are presented in Section 5.

II. BACKGROUND

In this section, the concepts of MOP, Pareto set and ϵ-efficient solutions are formally introduced. A multiobjective optimization problem[2] can be stated as follows:

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

subject to:

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

where $\boldsymbol{x} = [x_1, \ldots, x_k]$ is defined as a decision vector in the domain $Q \subset \Re^k$ and $\boldsymbol{f}$: $Q \rightarrow \Re^m$ is defined as the vector of objective functions $\boldsymbol{f}(\boldsymbol{x}) = [f_1(\boldsymbol{x}), \ldots, f_m(\boldsymbol{x})]$. $\underline{x_i}$ and $\overline{x_i}$ are the lower and upper bounds of each component of $\boldsymbol{x}$.

Definition 1 (Dominance [4]): 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): the Pareto set (denoted by 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): given a set of Pareto optimal solutions P_Q, the Pareto front $\boldsymbol{f}(P_Q)$ is defined as:

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

Definition 4 ($-\epsilon$-Dominance [3]): 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 (ϵ-Efficiency [2]): The set of ϵ-efficient solutions (denoted by $P_{Q,\epsilon}$) is the set of solutions in Q which are not $-\epsilon$-dominated by another solution in Q:

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

Generally, the manner to proceed is to obtain the two discrete sets $P^*_Q \subset P_Q$ and $P^*_{Q,\epsilon} \subset P_{Q,\epsilon}$, in such a way that P^*_Q and $P^*_{Q,\epsilon}$ appropriately characterize P_Q and $P_{Q,\epsilon}$, respectively. This is because determining P_Q and $P_{Q\epsilon}$ is usually unapproachable, since they may have infinite solutions. Note that the sets P^*_Q and $P^*_{Q,\epsilon}$ are not unique.

In any MOP there may be multimodal solutions [29]–[31] or nearly optimals [32], [33] that are ignored and can be useful for the designer. These options are sometimes used as an intermediate population [23], [24] to find the optimal alternatives more efficiently and are rarely alternatives provided to the designer [2], [28]. The usefulness of these solutions is even greater when objectives are added [1].

Figure 1 shows an example to illustrate the optimal, nearly optimal, and nearly optimal solutions nondominated in their neighborhood. The Figure 1a shows a monobjective problem, while in Figure 1b a multiobjetive problem is shown.

[1] Multiobjective Evolutionary Algorithm available in Matlab Central: https://www.mathworks.com/matlabcentral/fileexchange/71448-nevmoga-multiobjective-evolutionary-algorithm

[2] A maximization problem can be converted into a minimization problem. For each of the objectives that have to be maximized, the transformation: $\max f_i(\boldsymbol{x}) = -min(-f_i(\boldsymbol{x}))$ can be applied.

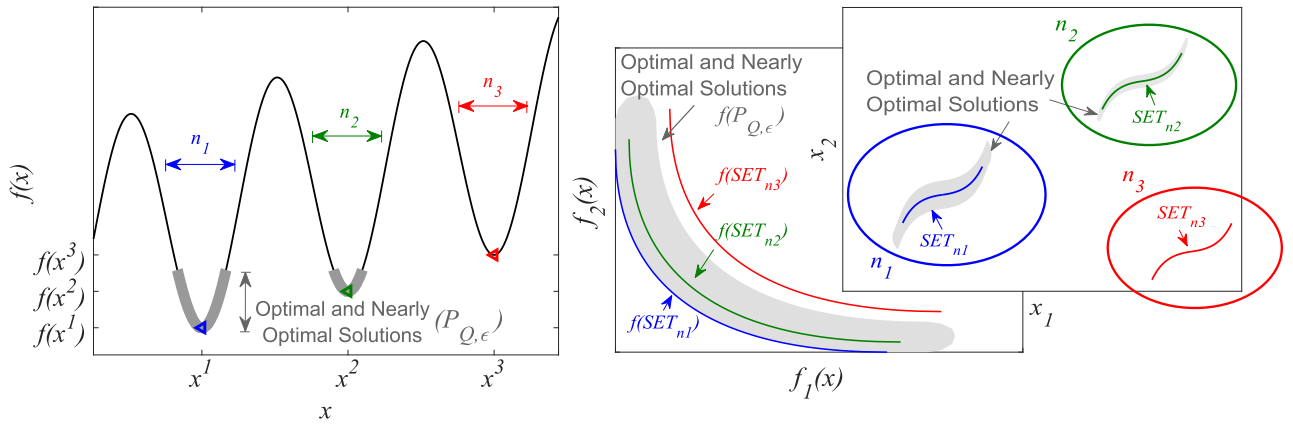


FIGURE 1. A monobjective example (on the left) and a multiobjective example (on the right). In the monobjective problem, the optimal solution is x^1 (blue triangle) and the nearly optimal solution nondominated in their neighborhood is x^2 (green triangle). In the multiobjective problem, the optimal solutions are the ones in set SET_{n1} (blue front) and the nearly optimal solutions nondominated in their neighborhood are the ones in SET_{n2} (green front).

Let us assume the monobjective case. In this case we have a single optimal solution $\boldsymbol{x^1}$ (P_Q). In addition, the shaded alternatives in gray are the set of optimal and nearly optimal solutions ($P_{Q,\epsilon}$). These alternatives are found in two different neighborhoods n_1 and n_2, while the neighborhood n_3 is discarded because it does not contain any nearly optimal solution. Therefore, a new neighborhood is observed, that differs from the neighborhood where the Pareto front is located (n_1), in which there are new feasible alternatives for studying differing characteristics and making the final decision with greater criterion. But adding the whole set $P_{Q,\epsilon}$ could slow down the algorithm and excessively complicate the decision stage. Therefore, the potentially useful solutions for the designer are the best solutions of each neighborhood, that is, the nearly optimal solutions nondominated in their neighborhood. In this case, these alternatives are $\boldsymbol{x^1}$ and $\boldsymbol{x^2}$. We believe these solutions provide the designer with useful knowledge without adding options that are dispensable (solutions dominated by $\boldsymbol{x^1}$ and $\boldsymbol{x^2}$ and with parameters similar to $\boldsymbol{x^1}$ and $\boldsymbol{x^2}$ respectively). Now let us look at the multiobjective case (see Figure 1b). In this case, we have a set of optimal solutions SET_{n1} that are in the neighborhood n_1. In addition, we have a set of optimal and nearly optimal solutions $P_{Q,\epsilon}$. All nearly optimal solutions could again slow down the algorithm and overcomplicate the decision stage. Therefore, in this case extrapolating the previous case, the potentially useful options are the sets SET_{n1} and SET_{n2}. The neighborhood n_3 is discarded because it does not contain any nearly optimal solution.

III. nevMOGA

In this work we use the algorithm nevMOGA [1], based on the algorithm ev-MOGA [6]. nevMOGA is an evolutionary algorithm designed to find the discrete set of optimal and nearly optimal solutions nondominated in their neighborhood. This set is defined as the set of $n-$efficient options ($P_{Q,n}$, see Definition 8).

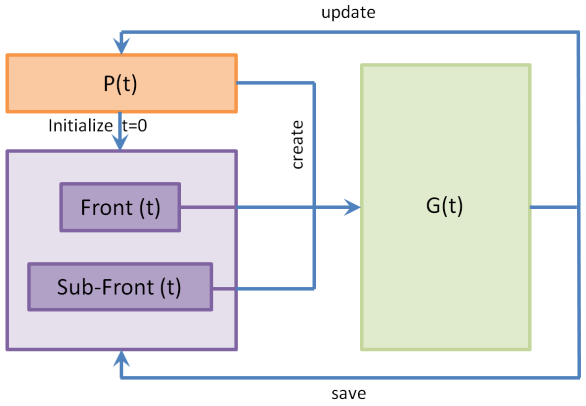


FIGURE 2. Structure of nevMOGA formed by four populations.

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 ($n-$Efficiency): The set of $n-$efficient solutions (denoted by $P_{Q,n}$) is the set of solutions of $P_{Q,\epsilon}$ which are not $n-$dominated by another solution in $P_{Q,\epsilon}$:

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

Normally, the manner to proceed is to obtain a discrete set $P^*_{Q,n} \subset P_{Q,n}$, in such a way that $P^*_{Q,n}$ appropriately characterizes $P_{Q,n}$. This is because determining $P_{Q,n}$ is usually unapproachable, since it may have infinite solutions. Note that the set $P^*_{Q,n}$ is not unique.

nevMOGA manages four populations (see Figure 2):

1) P(t) is the main population. The goal is that this population converges towards $P_{Q,n}$ and not only on P_Q, in order to achieve diversity in the set of solutions.

The number of individuals in this population is constant and equal to $Nind_P$.

2) Front(t) is the archive where P_Q^* is stored, i.e., a discrete approximation of the Pareto front. The size of this population varies but is always less than or equal to a given maximum size which depends on the number of boxes previously defined by the user.
3) Sub-front(t) is the archive where $P_{Q,n}^* \setminus P_Q^*$ is stored, i.e., a discrete approximation of the nearly optimal solutions nondominated in their neighborhood. Its size is variable but bounded, depending on the number of boxes.
4) G(t) is an auxiliary population where the new individuals generated by the algorithm in each iteration are stored. The number of individuals of this population is $Nind_G$, which must be multiple of 4.

IV. CONTROL EXAMPLES

This section presents the methodology for the optimal design of controllers with two real examples: the Wood & Berry distillation column and the CIC2018 control benchmark.

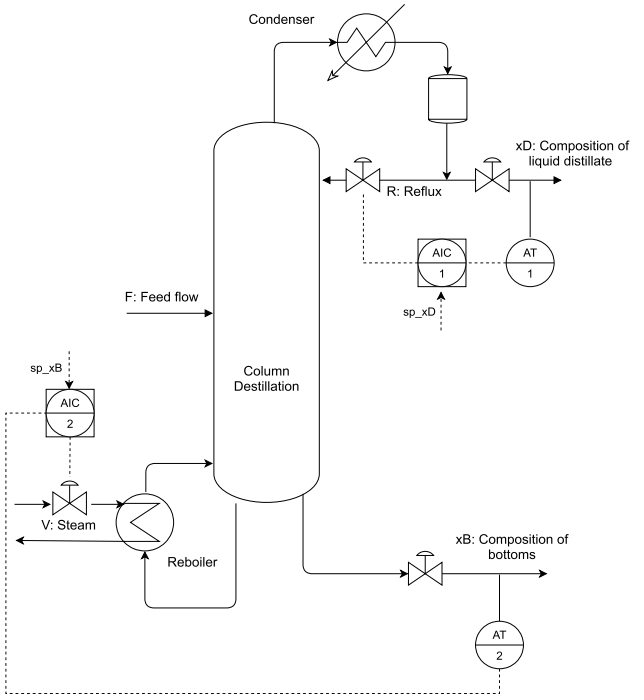


FIGURE 3. P&ID of the Wood & Berry distillation column [5].

A. WOOD & BERRY DISTILLATION COLUMN

The design of the control system of the Wood & Berry distillation column is proposed as the first application example [5] (see Figure 3). This control problem has been studied by many authors [15]–[17]. The system has two manipulated variables: steam flow V and reflux R in lb/min; and two outputs: concentration of distilled product xD, and at the bottom xB by weight of methanol. In addition, the feed flow F is a system disturbance in lb/min. The model of the plant for the point of operation ($xD = 0.96$, $xB = 0.02$, $R = 1.95$ lb/min, $S = 1.71$ lb/min y $F = 2.45lb/min$) is:

$$Y = \begin{pmatrix} xD \\ xB \end{pmatrix} = G(s) \begin{pmatrix} R \\ V \end{pmatrix} + Gd(s)\, F$$

$$G(s) = \begin{pmatrix} \dfrac{12.8e^{-s}}{16.7s+1} & \dfrac{-18.9e^{-3s}}{21s+1} \\ \dfrac{6.6e^{-7s}}{10.9s+1} & \dfrac{-19.4e^{-3s}}{14.4s+1} \end{pmatrix}$$

$$Gd(s) = \begin{pmatrix} \dfrac{3.8e^{-8.1s}}{14.9s+1} \\ \dfrac{4.9e^{-3.4s}}{13.2s+1} \end{pmatrix} \tag{2}$$

where time and delays are measured in minutes.

In this paper, the control structure is defined as a multi-loop PI control which uses a diagonal pairing scheme, i.e., output xD is controlled by R and xB by V. That is to say,

$$V = Kc_1 \left(e_1(s) + \frac{1}{Ti_1} \frac{1}{s} e_1(s) \right)$$

$$R = -Kc_2 \left(e_2(s) + \frac{1}{Ti_2} \frac{1}{s} e_2(s) \right) \tag{3}$$

where Kc_1 and Kc_2 are the proportional gains, Ti_1 and Ti_2 are the integral time constants (in minutes) and $e_1 = sp_{xD} - xD$ and $e_2 = sp_{xB} - xB$ are the output errors, where sp_{xD} and sp_{xB} are the setpoints for xD and xB respectively.

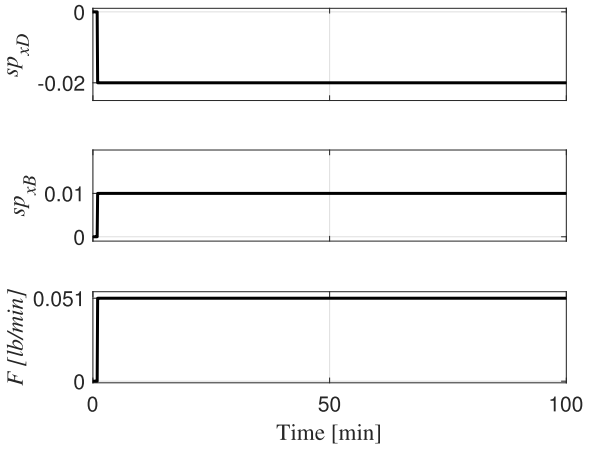


FIGURE 4. Steps regarding the operation point independently introduced for the control of the Wood & Berry distillation column.

To analyze the performance of the controller, three situations are analyzed with respect to the operation point: a step in sp_{xD} with $sp_{xB} = F = 0$, a step in sp_{xB} with $sp_{xD} = F = 0$ and a step in the disturbance F with $sp_{xD} = sp_{xB} = 0$. These three steps are shown in Figure 4.

The Wood & Berry MOP is stated as follows:

$$\min_{\boldsymbol{x}} \boldsymbol{f}(\boldsymbol{x}) = [f_1(\boldsymbol{x}) \quad f_2(\boldsymbol{x}) \tag{4}$$

where

$$f_1(\boldsymbol{x}) = \sum_{i=11}^{16} f_i$$
$$f_2(\boldsymbol{x}) = \sum_{i=21}^{26} f_i$$
$$\boldsymbol{x} = [Kc_1 \quad Ti_1 \quad Kc_2 \quad Ti_2] \tag{5}$$

where

$$f_{11}(\boldsymbol{x}) = \frac{IAE_1(\boldsymbol{x})}{IAE_1(\boldsymbol{x^R})}\Big|_{F=0}^{\substack{sp_{xD}=1\\sp_{xB}=0}} dt \tag{6}$$
$$f_{12}(\boldsymbol{x}) = \frac{IAE_2(\boldsymbol{x})}{IAE_2(\boldsymbol{x^R})}\Big|_{F=0}^{\substack{sp_{xD}=1\\sp_{xB}=0}} dt \tag{7}$$
$$f_{13}(\boldsymbol{x}) = \frac{IAE_1(\boldsymbol{x})}{IAE_1(\boldsymbol{x^R})}\Big|_{F=0}^{\substack{sp_{xD}=0\\sp_{xB}=1}} dt \tag{8}$$
$$f_{14}(\boldsymbol{x}) = \frac{IAE_2(\boldsymbol{x})}{IAE_2(\boldsymbol{x^R})}\Big|_{F=0}^{\substack{sp_{xD}=0\\sp_{xB}=1}} dt \tag{9}$$
$$f_{15}(\boldsymbol{x}) = \frac{IAE_1(\boldsymbol{x})}{IAE_1(\boldsymbol{x^R})}\Big|_{F=1}^{\substack{sp_{xD}=0\\sp_{xB}=0}} dt \tag{10}$$
$$f_{16}(\boldsymbol{x}) = \frac{IAE_2(\boldsymbol{x})}{IAE_2(\boldsymbol{x^R})}\Big|_{F=1}^{\substack{sp_{xD}=0\\sp_{xB}=0}} dt \tag{11}$$
$$f_{21}(\boldsymbol{x}) = \frac{IAVU_1(\boldsymbol{x})}{IAVU_1(\boldsymbol{x^R})}\Big|_{F=0}^{\substack{sp_{xD}=1\\sp_{xB}=0}} dt \tag{12}$$
$$f_{22}(\boldsymbol{x}) = \frac{IAVU_2(\boldsymbol{x})}{IAVU_2(\boldsymbol{x^R})}\Big|_{F=0}^{\substack{sp_{xD}=1\\sp_{xB}=0}} dt \tag{13}$$
$$f_{23}(\boldsymbol{x}) = \frac{IAVU_1(\boldsymbol{x})}{IAVU_1(\boldsymbol{x^R})}\Big|_{F=0}^{\substack{sp_{xD}=0\\sp_{xB}=1}} dt \tag{14}$$
$$f_{24}(\boldsymbol{x}) = \frac{IAVU_2(\boldsymbol{x})}{IAVU_2(\boldsymbol{x^R})}\Big|_{F=0}^{\substack{sp_{xD}=0\\sp_{xB}=1}} dt \tag{15}$$
$$f_{25}(\boldsymbol{x}) = \frac{IAVU_1(\boldsymbol{x})}{IAVU_1(\boldsymbol{x^R})}\Big|_{F=1}^{\substack{sp_{xD}=0\\sp_{xB}=0}} dt \tag{16}$$
$$f_{26}(\boldsymbol{x}) = \frac{IAVU_2(\boldsymbol{x})}{IAVU_2(\boldsymbol{x^R})}\Big|_{F=1}^{\substack{sp_{xD}=0\\sp_{xB}=0}} dt \tag{17}$$

where

$$IAE_i = \int_0^{100min} |e_i(t)|\, dt$$
$$IAVU_i = \int_0^{100min} \left|\frac{du_i(t)}{dt}\right| dt$$
$$\boldsymbol{x^R} = [0.6523 \quad 16.7 \quad -0.1237 \quad 14.4] \tag{18}$$

subject to

$$\underline{\boldsymbol{x}} \le \boldsymbol{x} \le \overline{\boldsymbol{x}}$$
$$f_{ij}(\boldsymbol{x}) < 2 \quad \forall i = [1, 2], \quad j = [1, \ldots, 5]$$

and where

$$\underline{\boldsymbol{x}} = [0.01 \quad 0.1 \quad -0.5 \quad 0.1]$$
$$\overline{\boldsymbol{x}} = [2 \quad 50 \quad -0.01 \quad 50]$$

In this MOP, different objectives have been aggregated to simplify the optimization process and the decision phase. This aggregation is made with the same weight because they have same units and the relative importance of these objectives is the same. Therefore, we have an MOP with two design objectives. To calculate the first objective f_1 the IAEs are added in both outputs relativized on the reference controller $\boldsymbol{x^R}$ calculated using the S-IMC technique [36] in each of the three cases: set point on output 1 (f_{11} and f_{12}), set point on output 2 (f_{13} and f_{14}) and disturbance (f_{15} and f_{16}). In the same way, the derivatives of the control actions relativized on the reference controller are measured ($\boldsymbol{x^R}$) in each of the three mentioned cases (f_{21}, f_{22}, f_{23}, f_{24}, f_{25} and f_{26}). These values are added to form f_2. The decision variables are Kc and Ti of the two PIs proposed for the control of the plant (see Equation 3).

To optimize the Wood & Berry MOP defined in (4) nevMOGA is used with the following configuration:

$$Nind_G = 8$$
$$Nind_P = 250$$
$$Iterations = 1000$$
$$n\_box = 80$$
$$\boldsymbol{\epsilon} = [0.5 \quad 0.5]$$
$$\boldsymbol{n} = [0.25 \quad 5 \quad 0.025 \quad 5]$$

for the definition of the remaining parameters, the values suggested in [39] are used for the original algorithm (ev-MOGA).

Figure 5 shows a set of controllers found using nevMOGA (optimals and nearly optimals) for the problem posed. There is also an isolated neighborhood ($Neighborhood_1$, solutions in green) with higher values of Ti_1. These solutions are the nearly optimal options with performances more similar to the optimal Pareto front. Two solutions are chosen to compare behavior. Firstly, the controller with the lowest infinite norm of the Pareto front ($\boldsymbol{x^1}$) is chosen. This alternative has a compensated trade-off and could be the final choice of the designer if just the Pareto front solutions were analyzed. Secondly, we choose a solution ($\boldsymbol{x^2}$) of the isolated neighborhood ($Neighborhood_1$) that has a similar performance to the solution $\boldsymbol{x^1}$ (see Table 3).

The values of the aggregated objectives of both options can be seen in Table 4. Firstly, it can be seen that in response to the step on xD (Figure 6) the nearly optimal solution produces a slightly lower error for both outputs (f_{11} and f_{12}).

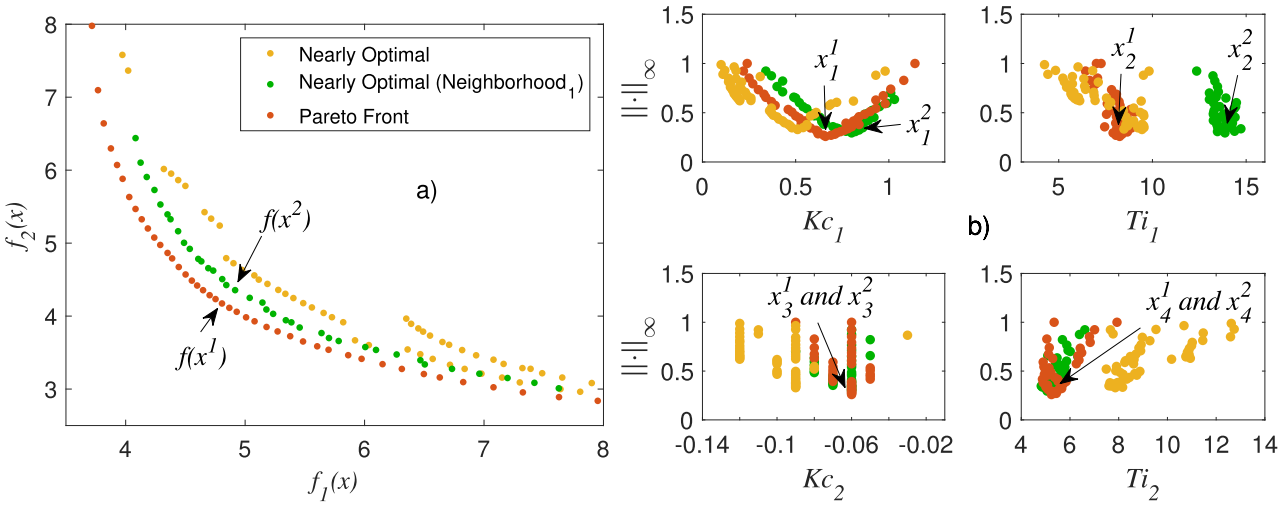


FIGURE 5. Set of optimal controllers (red) and nearly optimal controllers nondominated in their neighborhood (green and yellow) for the Wood & Berry MOP. Figure 5a shows the objective value of each of the options found. Figure 5b shows the decision variables represented with Level Diagrams (LD [35]) using the infinite norm.

TABLE 3. Controllers x^1 and x^2 for a Wood & Berry distillation column.

Controllers	Kc_1	Ti_1	Kc_2	Ti_2	$f_1(\boldsymbol{x})$	$f_2(\boldsymbol{x})$
$\boldsymbol{x^1}$	0.66	8.25	-0.06	5.24	**4.806**	**4.173**
$\boldsymbol{x^2}$	0.8	13.87	-0.06	5.06	4.916	4.356

TABLE 4. Objective value of the controllers x^1 and x^2 for a Wood & Berry distillation column.

C.	$f_{11}(\boldsymbol{x})$	$f_{12}(\boldsymbol{x})$	$f_{13}(\boldsymbol{x})$	$f_{14}(\boldsymbol{x})$	$f_{15}(\boldsymbol{x})$	$f_{16}(\boldsymbol{x})$
$\boldsymbol{x^1}$	0.856	1.04	**0.499**	0.86	**0.798**	0.751
$\boldsymbol{x^2}$	**0.819**	**0.973**	0.678	**0.823**	0.894	**0.73**
C.	$f_{21}(\boldsymbol{x})$	$f_{22}(\boldsymbol{x})$	$f_{23}(\boldsymbol{x})$	$f_{24}(\boldsymbol{x})$	$f_{25}(\boldsymbol{x})$	$f_{26}(\boldsymbol{x})$
$\boldsymbol{x^1}$	**0.965**	0.417	0.431	0.387	**1.147**	0.826
$\boldsymbol{x^2}$	1.186	**0.374**	**0.405**	**0.381**	1.19	**0.82**

With respect to the control actions, the controller $\boldsymbol{x^1}$ obtains smoother control actions in R (f_{21}) and more abrupt actions in V (f_{22}) with respect to $\boldsymbol{x^2}$. Secondly, the setpoint tracking for xB (see Figure 7) the greatest difference between the controllers is observed. While in xB (f_{14}) the controller $\boldsymbol{x^2}$ has a slightly smaller error, in xD (f_{13}) it is controller $\boldsymbol{x^1}$ that produces a significantly smaller error. Both control actions (f_{23} and f_{24}) are smoother for the controller $\boldsymbol{x^2}$. Finally, a response to a step in the disturbance can be seen (see Figure 8). In this case the controller $\boldsymbol{x^1}$ obtains a smaller error for xD (f_{15}) with smoother control actions in R (f_{25}). In addition, $\boldsymbol{x^2}$ obtained smaller errors for xB (f_{16}) with smoother control actions V (f_{26}).

Therefore, if we focus on the value of the optimized targets, the controller $\boldsymbol{x^1}$ dominates $\boldsymbol{x^2}$. But as seen in their responses, despite having similar values for f_1 and f_2, both controllers have different characteristics. After observing their responses, you may wish prioritize, for example, a controller with less overshoot, which would favor a choice towards $\boldsymbol{x^2}$ (with significantly lower overshoot in xD the setpoint tracking in sp_{xB}, see Figure 7). Therefore, it is very useful to compare controllers from different neighborhoods. This detailed comparison provides us with new information for making a final decision with more criteria on which control to use.

We now analyze a new indicator, related to robustness, that is not included in the optimization phase. For the study of the robustness of the controllers $\boldsymbol{x^1}$ (optimal) and $\boldsymbol{x^2}$ (nearly optimal), we will obtain the singular minimum values of the multiplicative uncertainty [10]–[12]. To do this, we observe the frequency plot (see Figure 9). The controller $\boldsymbol{x^2}$ has a singular minimum value of multiplicative uncertainty ($\gamma = 0.55$) greater than $\boldsymbol{x^1}$ ($\gamma = 0.5$). This value indicates that the nearly optimal controller $\boldsymbol{x^2}$ is more robustly stable than the Pareto front controller $\boldsymbol{x^1}$. This characteristic has not been taken into account in the design objectives and could change the choice towards a nearly optimal solution, since both have very similar f_1 and f_2 and one of them ($\boldsymbol{x^2}$) is slightly more robust.

Therefore, in this PI multi-loop controller design problem, it is clear that obtaining nearly optimal solutions nondominated in their neighborhood can be useful for the designer. With them, a detailed study can be made that enables the designer to choose with greater criteria an optimal or nearly optimal option, since both can perform similarly with respect to the design objectives.

B. CIC2018 CONTROL BENCHMARK

In the contest in control engineering 2018 (CIC2018 [14]) the design of a refrigeration system control was proposed. This process has two controlled variables: evaporator secondary fluid outlet temperature $T_{e,sec,out}$ and degree of overheating T_{SH} in ^oC. There are two manipulated variables: opening of the expansion valve Av in (%) and compressor speed N in Hz. There are also two disturbances: inlet temperature of the secondary fluid of the evaporator ($T_{e,sec,in}$) and the condenser ($T_{c,sec,in}$) in ^oC (see Figure 10).

The control structure is defined as a multi-loop PI control which uses the following loop pairing scheme: output

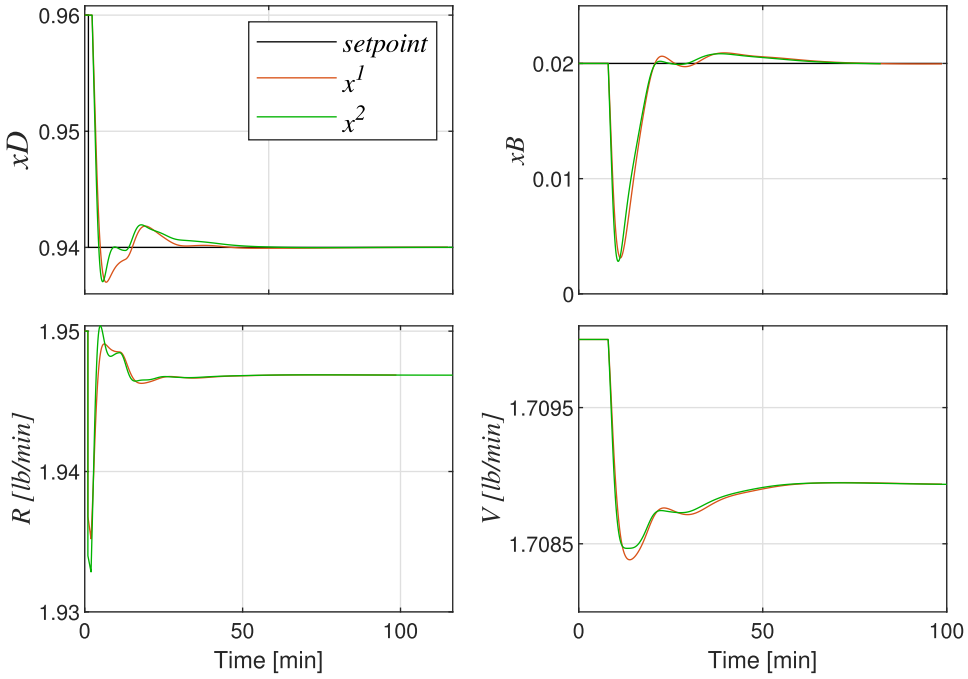


FIGURE 6. Comparison of the response to a setpoint for *xD* of the controllers x^1 and x^2 in a Wood & Berry distillation column.

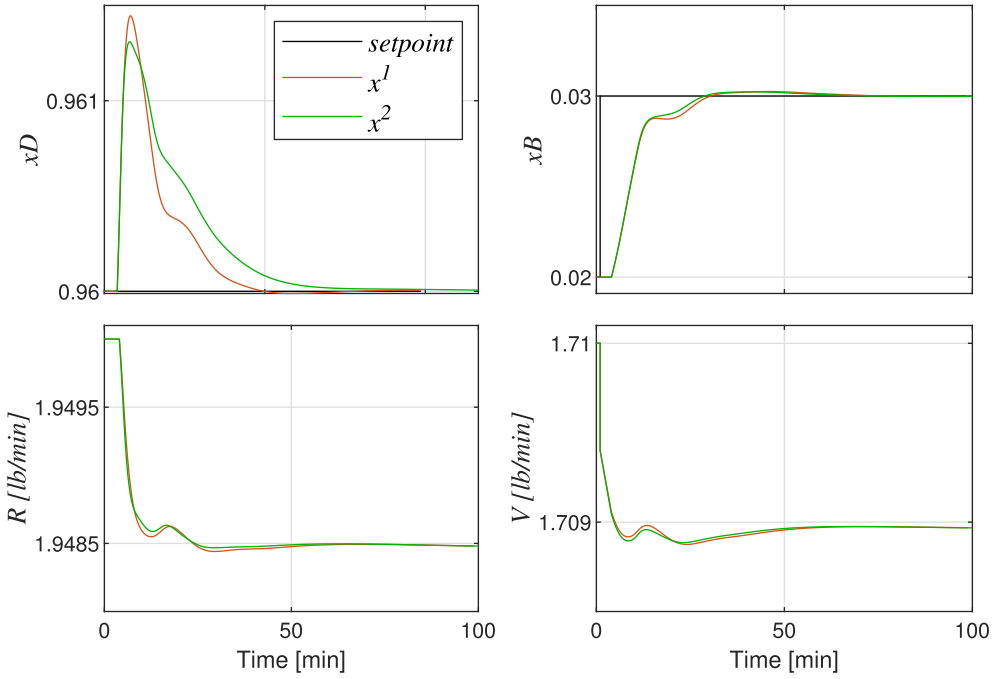


FIGURE 7. Comparison of the response to a setpoint for *xB* of the controllers x^1 and x^2 in a Wood & Berry distillation column.

$T_{e,sec,out}$ is controlled by Av and T_{SH} by N. That is to say,

$$Av = Gr_1(s)\, e_1(s) = -Kc_1 \left(e_1(s) + \frac{1}{Ti_1} \frac{1}{s} e_1(s) \right)$$
$$N = Gr_2(s)\, e_2(s) = Kc_2 \left(e_2(s) + \frac{1}{Ti_2} \frac{1}{s} e_2(s) \right) \quad (19)$$

where Kc_1 and Kc_2 are the proportional gains, Ti_1 and Ti_2 are the integral time constants in seconds and $e_1 = sp_{Tsec} - T_{e,sec,out}$ and $e_2 = sp_{TSH} - T_{SH}$ are the output errors, where sp_{Tsec} and sp_{TSH} are the setpoints for $T_{e,sec,out}$ and T_{SH} respectively. For the first phase of the contest, four profiles are provided: two for the setpoints sp_{Tsec} and sp_{TSH}

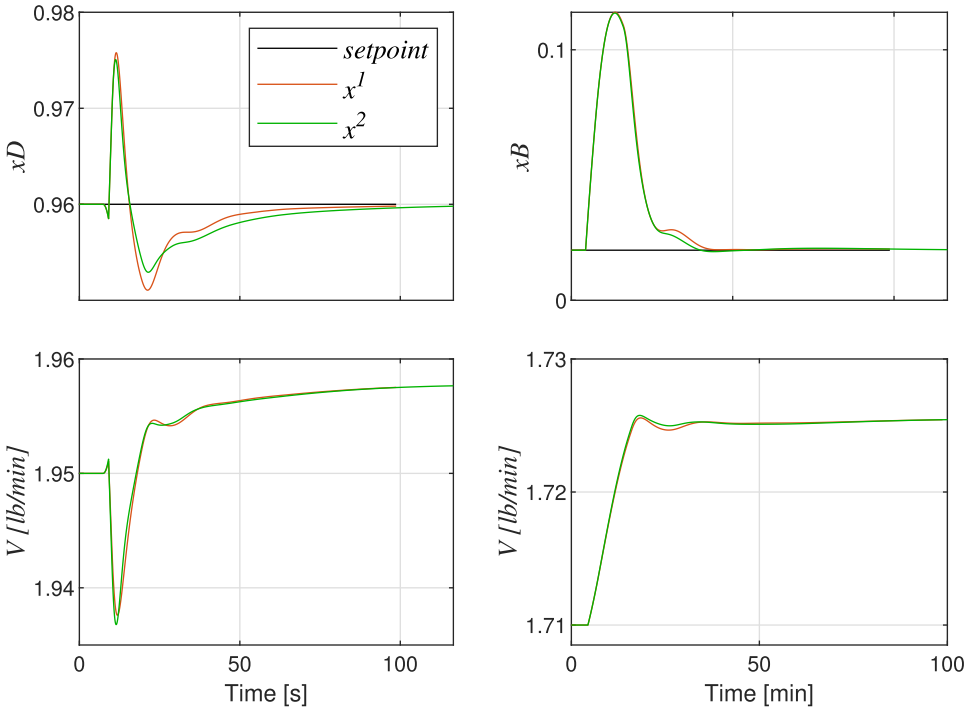


FIGURE 8. Comparison of a response to a step for *F* of the controllers x^1 and x^2 in a Wood & Berry distillation column.

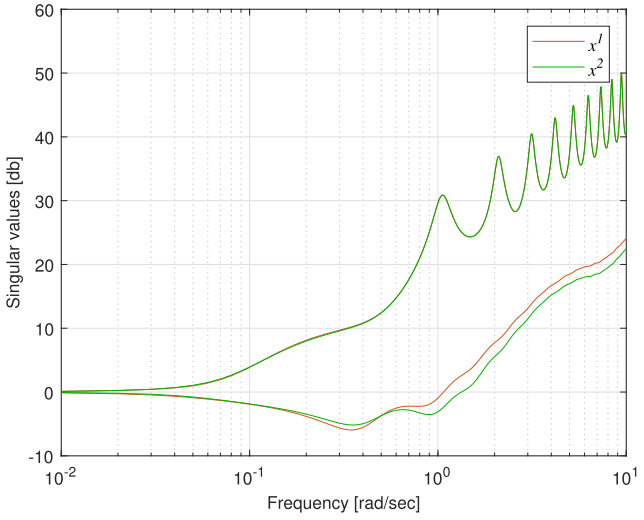


FIGURE 9. Singular value of multiplicative uncertainty for the Wood & Berry of the controllers x^1 and x^2.

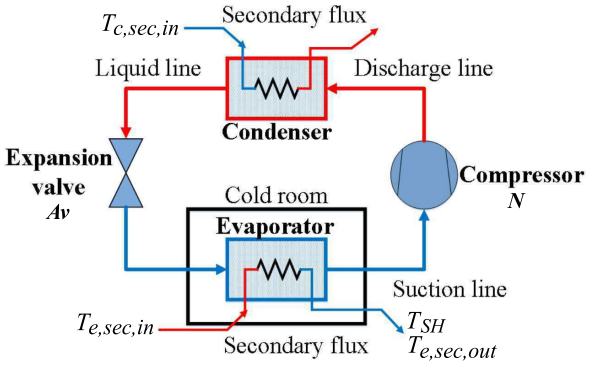


FIGURE 10. Schematic picture of one-compression-stage, one-load-demand vapor-compression refrigeration cycle [14].

(see Figure 12) and two for the disturbances $T_{e,sec,in}$ and $T_{c,sec,in}$ (see Figure 13). In addition, the operating point is defined (see Table 5).

1) MONOBJECTIVE OPTIMIZATION PROBLEM

Firstly, the problem is defined with a single design objective. This objective is the aggregation of various objectives. This simplifies the optimization process and the decision phase. For the calculation of these aggregate objectives, we start with a reference controller x^R provided by the contest that is defined by Gr_1 output controller $T_{e,sec,out}$ and Gr_2 and output controller T_{SH}:

$$Gr_1(s) = \frac{-0.6000s^2 - 0.0300s + 0.5000}{s^2 - 1.9853s + 0.9853}$$

$$Gr_2(s) = \frac{0.238s + 0.25}{0.95s} \tag{20}$$

The optimization problem is stated as follows:

$$\min_{\boldsymbol{x}} f(\boldsymbol{x}) \tag{21}$$

where

$$f(\boldsymbol{x}) = \frac{\sum_{i=11}^{18} f_i}{\sum_{i=1}^{8} w_i}$$
$$\boldsymbol{x} = [Kc_1 \quad Ti_1 \quad Kc_2 \quad Ti_2] \tag{22}$$

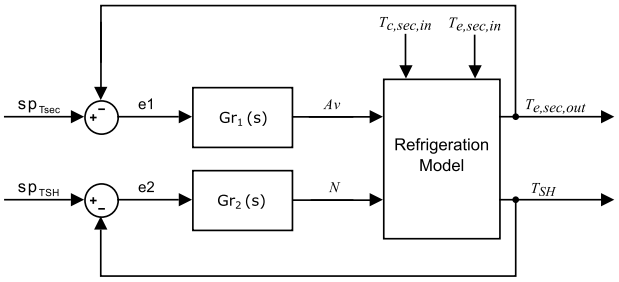


FIGURE 11. Control system for the CIC2018 benchmark control [14].

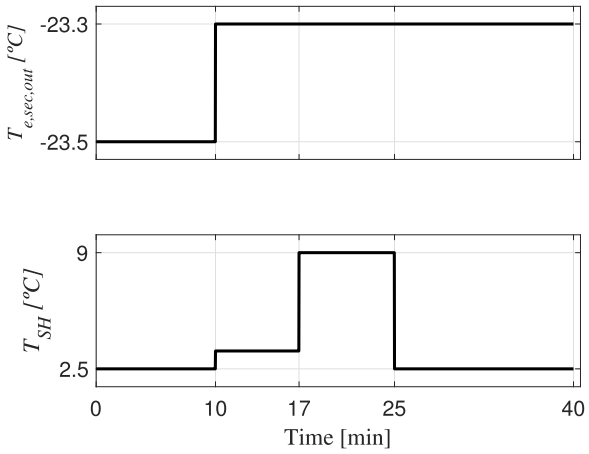


FIGURE 12. Profile introduced for sp_{Tsec} and sp_{TSH} for the CIC2018 control benchmark.

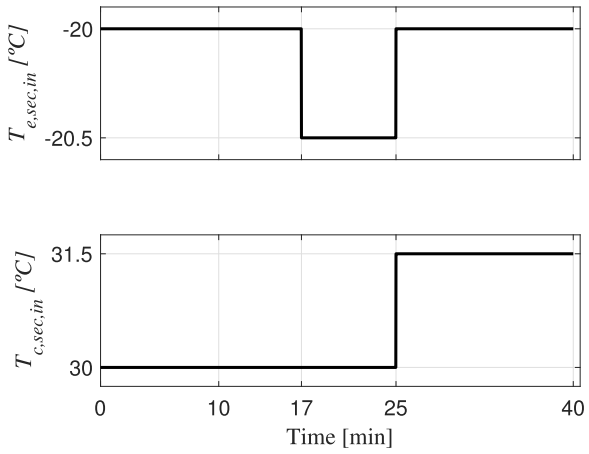


FIGURE 13. Profile introduced for the disturbances $T_{e,sec,in}$ and $T_{c,sec,in}$ for the CIC2018 control benchmark.

where

$$
\begin{aligned}
f_{11}(\boldsymbol{x}) &= w_1 \cdot RIAE_1(\boldsymbol{x}, x^R)\\
f_{12}(\boldsymbol{x}) &= w_2 \cdot RIAE_2(\boldsymbol{x}, x^R)\\
f_{13}(\boldsymbol{x}) &= w_3 \cdot RITAE_1(\boldsymbol{x}, x^R, t_{c1}, t_{s1})\\
f_{14}(\boldsymbol{x}) &= w_4 \cdot RITAE_2(\boldsymbol{x}, x^R, t_{c2}, t_{s2})\\
f_{15}(\boldsymbol{x}) &= w_5 \cdot RITAE_2(\boldsymbol{x}, x^R, t_{c3}, t_{s3})\\
f_{16}(\boldsymbol{x}) &= w_6 \cdot RITAE_2(\boldsymbol{x}, x^R, t_{c4}, t_{s4})\\
f_{17}(\boldsymbol{x}) &= w_7 \cdot RIAVU_1(\boldsymbol{x}, x^R)\\
f_{18}(\boldsymbol{x}) &= w_8 \cdot RIAVU_2(\boldsymbol{x}, x^R)
\end{aligned}
\tag{23}
$$

TABLE 5. Operation point for CIC2018 control benchmark.

Variable		Value	Units
Manipulated variables	Av	$\cong$ 74.45	%
	N	$\cong$ 42.45	Hz
Disturbances	$T_{c,sec,in}$	30	^oC
	$m_{c,sec}$	150	$g\ s^{-1}$
	$P_{c,sec,in}$	1	bar
	$T_{e,sec,in}$	-20	^oC
	$m_{e,sec}$	64.503	$g\ s^{-1}$
	$P_{e,sec,in}$	1	bar
	T_{surr}	25	^oC
Controlled variables	$T_{e,sec,out}$	$\cong$ -23.5	^oC
	T_{SH}	$\cong$ 2.5	^oC

where

$$
\begin{aligned}
IAE_i &= \int_0^{40min} |e_i(t)|\, dt\\
ITAE_i &= \int_{tc_i}^{tc_i+ts_i} (t - tc_i)\, |e_i(t)|\, dt\\
IAVU_i &= \int_0^{40min} \left|\frac{du_i(t)}{dt}\right| dt
\end{aligned}
\tag{24}
$$

$$
\begin{aligned}
RIAE_i(\boldsymbol{x}, x^R) &= \frac{IAE_i(\boldsymbol{x})}{IAE_i(x^R)}\\
RITAE_i(\boldsymbol{x}, x^R, tc_i, ts_i) &= \frac{ITAE_i(\boldsymbol{x}, tc_i, ts_i)}{ITAE_i(x^R, tc_i, ts_i)}\\
RIAVU_i(\boldsymbol{x}, x^R) &= \frac{IAVU_i(\boldsymbol{x})}{IAVU_i(x^R)}
\end{aligned}
\tag{25}
$$

subject to

$$
\begin{aligned}
&\underline{\boldsymbol{x}} \le \boldsymbol{x} \le \overline{\boldsymbol{x}}\\
&f(\boldsymbol{x}) < 0.2
\end{aligned}
$$

and where

$$
\begin{aligned}
&\underline{\boldsymbol{x}} = [-35 \;\; 0.75 \;\; 0.25 \;\; 1.75]\\
&\overline{\boldsymbol{x}} = [-5 \;\; 1.5 \;\; 1.25 \;\; 3.75]\\
&w_1 = 0.229 \quad w_2 = 0.120 \quad w_3 = 0.224 \quad w_4 = 0.114\\
&w_5 = 0.114 \quad w_6 = 0.113 \quad w_7 = 0.0264 \quad w_8 = 0.0593\\
&tc_1 = tc_2 = 10\,min \quad ts_1 = ts_2 = tc_3 = 17\ min\\
&ts_3 = tc_4 = 25\ min \quad ts_4 = 40\ min
\end{aligned}
$$

Eight aggregate performance goals are evaluated for the calculation of the design objective. The first two aggregate goals are the integral absolute errors (IAEs) in both controlled variables ($T_{e,sec,out}$ and T_{SH}). The third is the integral time absolute error (ITAE) for the first controlled variable ($T_{e,sec,out}$) evaluated after the step that occurs at 10 *min* (see Figure 12). The fourth, fifth, and sixth aggregate goals are the ITAE for the second controlled variable (T_{SH}) evaluated after the three steps that occur at 10, 17 and 25 *min* (see Figure 12). The seventh and eighth aggregate objectives are the integral of absolute variation of control signal (IAVU) for the two manipulated variables (Av and N). The overall objective to be optimized is obtained using the average value of the eight

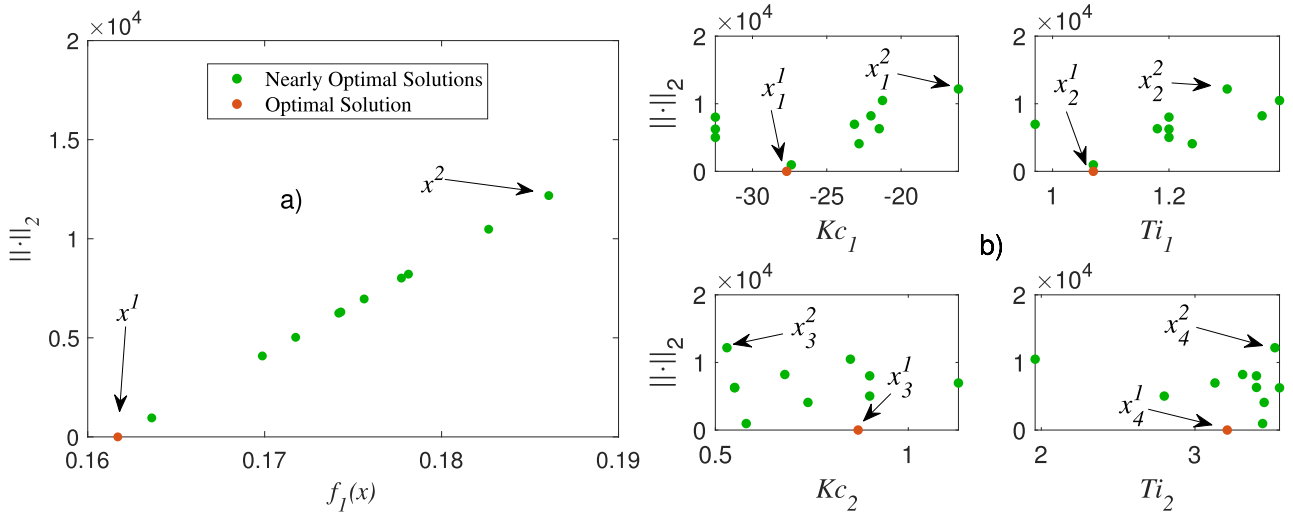


FIGURE 14. Optimal controller (red) and nearly optimal controllers nondominated in their neighborhood (green) for the CIC2018 benchmark control. Shown with LD using norm 2. The Figure shows the design objective. Figure b shows the decision variables.

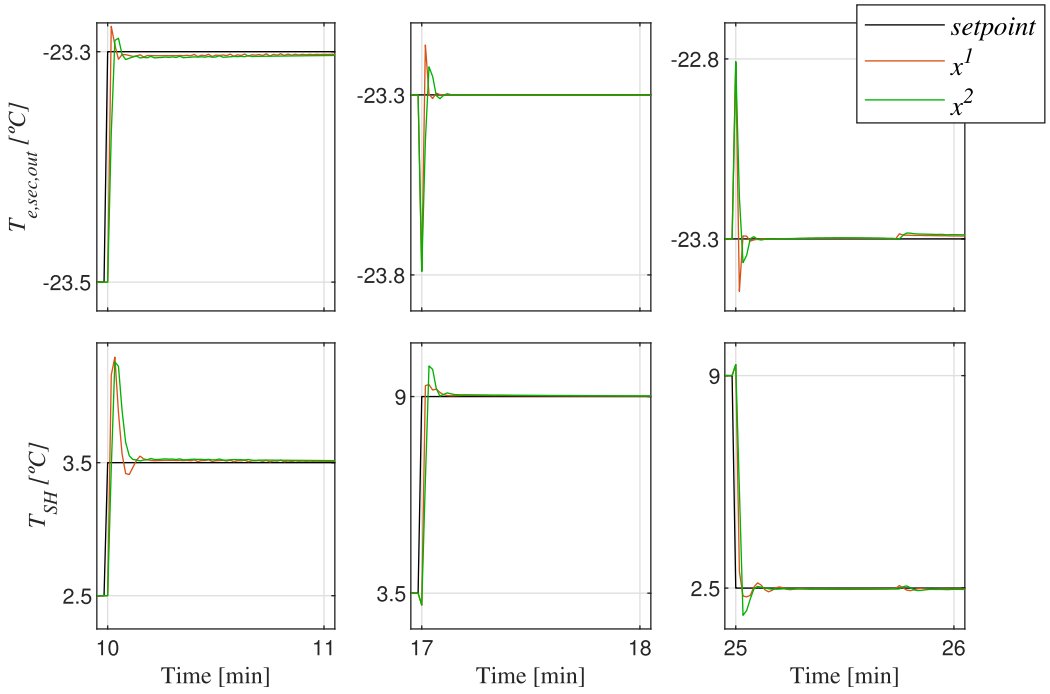


FIGURE 15. Comparison of the controller responses x^1 and x^2 and for the CIC2018 control benchmark. Details of the moments in time when setpoint changes occurred.

individual aggregate objectives with a weighting factor w_i (proposed by the benchmark) for each aggregate objective.

To optimize the problem defined in (21) nevMOGA is used with the following configuration:

$$\begin{aligned} Nind_P &= 250 \\ Nind_G &= 4 \\ Iterations &= 800 \\ N\_box &= 100 \\ \epsilon &= [0.025] \\ \boldsymbol{n} &= [5 \quad 0.1 \quad 0.25 \quad 1] \end{aligned}$$

for the definition of the remaining parameters, the default values suggested by [39] are taken from the original algorithm (ev-MOGA).

Firstly, the optimal solution and the set of nearly optimal solutions nondominated in their neighborhood are obtained – as shown in Figure 14.

The optimal solution $\boldsymbol{x^1}$ is now selected together with the nearly optimal solution $\boldsymbol{x^2}$, which is in a different neighborhood. In Figures 15 and 16 the response of both controls is observed. In addition, their aggregate objectives and overall target value are shown in the Tables 6 and 7. The controller $\boldsymbol{x^1}$ has a lower error in the outputs as shown in Figure 15 and the

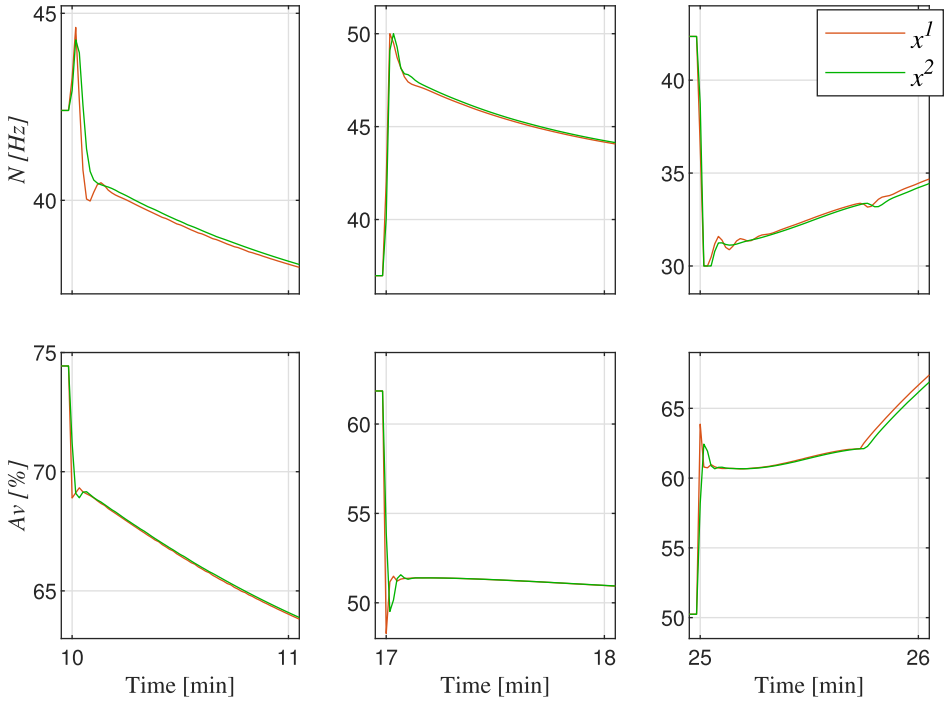


FIGURE 16. Comparison of the controller control actions x^1 and x^2 for the CIC2018 benchmark control. Detail of the moments in time when changes in the setpoints occurred.

TABLE 6. Controllers x^1 and x^2 for the monobjective approach of the CIC2018 benchmark control.

Controller	Kc_1	Ti_1	Kc_2	Ti_2	$f(\boldsymbol{x})$
$\boldsymbol{x^1}$	-27.7	1.07	0.87	3.21	**0.162**
$\boldsymbol{x^2}$	-16.14	1.3	0.53	3.52	0.185

TABLE 7. Value of the aggregated objectives for the controllers x^1 and x^2.

Controller	$f_{11}(\boldsymbol{x})$	$f_{12}(\boldsymbol{x})$	$f_{13}(\boldsymbol{x})$	$f_{14}(\boldsymbol{x})$
$\boldsymbol{x^1}$	**0.026**	**0.013**	**0.019**	**0.005**
$\boldsymbol{x^2}$	0.033	0.017	0.027	0.008
Controller	$f_{15}(\boldsymbol{x})$	$f_{16}(\boldsymbol{x})$	$f_{17}(\boldsymbol{x})$	$f_{18}(\boldsymbol{x})$
$\boldsymbol{x^1}$	**0.004**	**0.007**	0.029	0.059
$\boldsymbol{x^2}$	0.007	0.011	**0.027**	**0.056**

value of the aggregated objectives f_{11} and f_{16}. The controller $\boldsymbol{x^2}$ shows smoother control actions as seen in Figure 16 and the aggregated objectives f_{17} to f_{18}. This reveals the conflict between the aggregate objectives relative to the errors in the outputs and control actions. This conflict has been revealed because we have found the set of nearly optimal options nondominated in their neighborhood. To analyze this conflict in greater detail, a multiobjective optimization problem is posed with two objectives, where aggregate objectives that enter into conflict are grouped independently. The results of this multiobjective approach are discussed in the next section.

2) MOP

Two objectives are defined for the design of the multiobjective optimization problem. The first objective is formed by the first six aggregate objectives defined in (23) with their weighting factors w_i. The second objective is formed by the last two aggregate objectives with their corresponding weighting factors. In this way, the aggregate objectives related to the errors on the outputs and the control actions are studied independently. The MOP for the CIC2018 control benchmark is:

$$\min_{\boldsymbol{x}} \boldsymbol{f}(\boldsymbol{x}) = [f_1(\boldsymbol{x}) \quad f_2(\boldsymbol{x})] \tag{26}$$

where

$$\begin{aligned} f_1(\boldsymbol{x}) &= \frac{\sum_{i=11}^{16} f_i}{\sum_{i=1}^{6} w_i} \\ f_2(\boldsymbol{x}) &= \frac{\sum_{i=17}^{18} f_i}{\sum_{i=7}^{8} w_i} \\ \boldsymbol{x} &= [Kc_1 \quad Ti_1 \quad Kc_2 \quad Ti_2] \end{aligned} \tag{27}$$

subject to

$$\begin{aligned} \underline{\boldsymbol{x}} \leq \boldsymbol{x} \leq \overline{\boldsymbol{x}} \\ f_i(\boldsymbol{x}) < 1.4 \quad \forall i \in [1 \quad 2] \end{aligned}$$

where $f_{11}(\boldsymbol{x}) \ldots f_{18}(\boldsymbol{x})$, $\underline{\boldsymbol{x}}$, $\overline{\boldsymbol{x}}$, w_i, t_{ci} y t_{si} are those defined in (23) and (24) and x^R is defined in (20).

For the optimization of the MOP defined in (26) nevMOGA is used with the following configuration:

$$\begin{aligned} Nind_P &= 100 \\ Nind_G &= 4 \\ Iterations &= 800 \\ n\_box &= 100 \\ \boldsymbol{\epsilon} &= [0.0125 \quad 0.0125] \\ \boldsymbol{n} &= [2.5 \quad 0.2 \quad 0.25 \quad 0.5] \end{aligned}$$

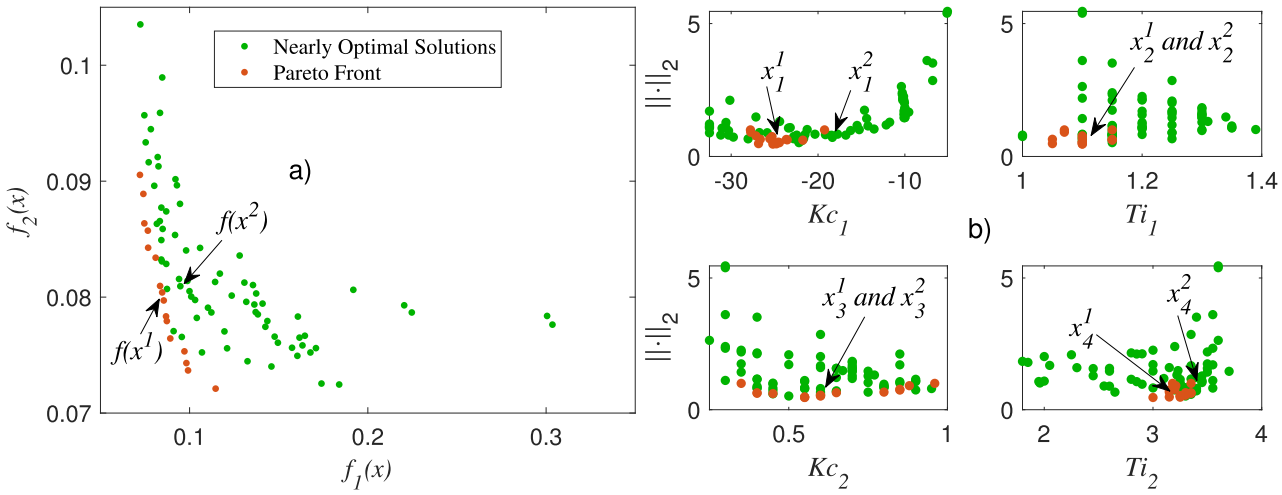


FIGURE 17. Set of optimal controllers (red) and nearly optimal controllers nondominated in their neighborhood (green) for the CIC2018 control benchmark. Figure 17 shows the trade-off of the design objectives. Figure 17b shows the decision variables represented with LD using norm 2.

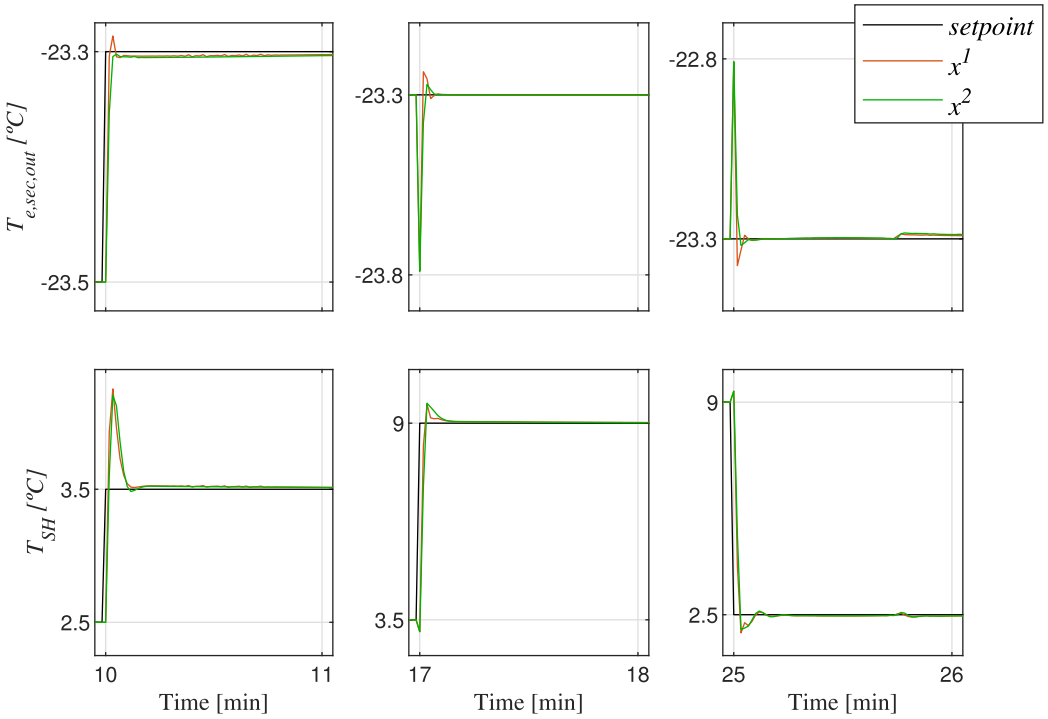


FIGURE 18. Comparison of the response for the controller setpoints x^1 and x^2 for the CIC2018 control benchmark. Detail of the time instants when changes in the setpoints were produced.

TABLE 8. Controllers x^1 and x^2 for the MOP of the CIC2018 control benchmark.

Controller	Kc_1	Ti_1	Kc_2	Ti_2	$f_1(\boldsymbol{x})$	$f_2(\boldsymbol{x})$
$\boldsymbol{x^1}$	-24.45	1.1	0.6	3.25	**0.085**	**0.08**
$\boldsymbol{x^2}$	-18.35	1.1	0.65	3.4	0.095	0.081

TABLE 9. Values of the aggregated objectives of the controllers x^1 and x^2.

Controller	$f_{11}(\boldsymbol{x})$	$f_{12}(\boldsymbol{x})$	$f_{13}(\boldsymbol{x})$	$f_{14}(\boldsymbol{x})$
$\boldsymbol{x^1}$	**0.027**	**0.015**	**0.021**	0.007
$\boldsymbol{x^2}$	0.03	0.015	0.028	**0.007**
Controller	$f_{15}(\boldsymbol{x})$	$f_{16}(\boldsymbol{x})$	$f_{17}(\boldsymbol{x})$	$f_{18}(\boldsymbol{x})$
$\boldsymbol{x^1}$	0.006	0.01	0.027	**0.054**
$\boldsymbol{x^2}$	**0.005**	**0.009**	**0.024**	0.056

for the definition of the remaining parameters, the default values suggested by [39] are taken from the original algorithm (ev-MOGA).

Figure 17 shows the optimization results using nevMOGA for the proposed problem. Two solutions are again chosen for analysis. Firstly, an optimal solution is chosen with a compensated trade-off ($\boldsymbol{x^1}$). Secondly, a nearly optimal solution $\boldsymbol{x^2}$ dominated by $\boldsymbol{x^1}$ in a different neighborhood is selected (see Tables 8 and 9 and Figure 17) since Kc_1 and Ti_2 differ.

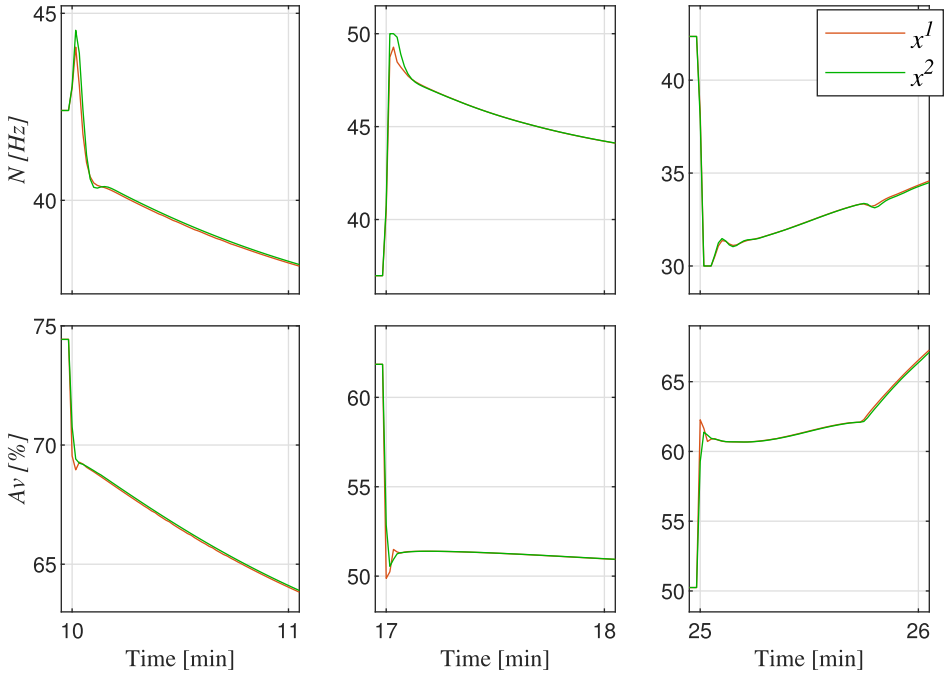


FIGURE 19. Comparison of the controller control actions x^1 and x^2 for the CIC2018 control benchmark. Detail of the instants in time when changes in setpoints are produced.

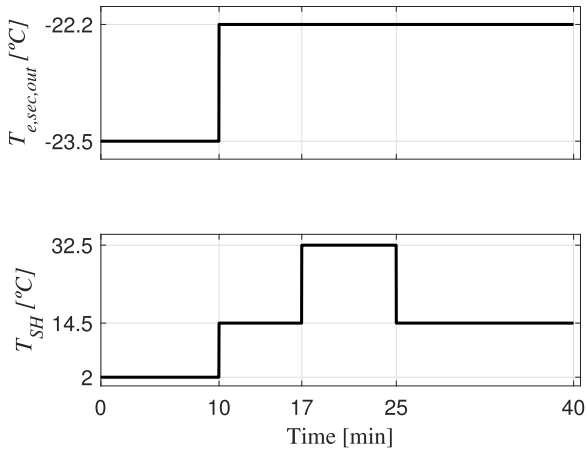


FIGURE 20. Steps in sp_{Tsec} and sp_{TSH} introduced on $T_{e,sec,out}$ and T_{SH} respectively for the CIC2018 control benchmark validation test.

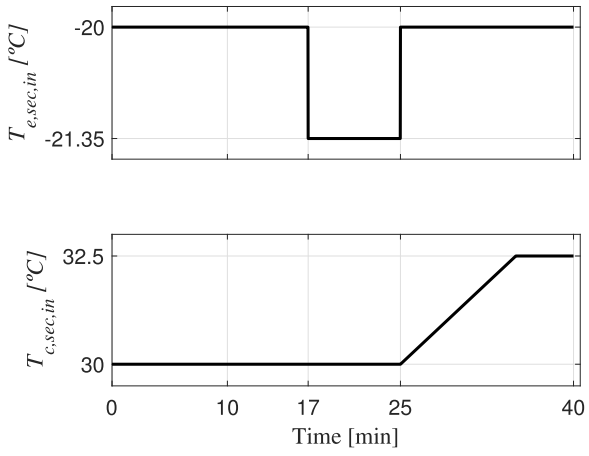


FIGURE 21. Steps on the disturbances $T_{e,sec,in}$ and $T_{c,sec,in}$ for the CIC2018 control benchmark validation test.

The response of both controllers is seen in Figures 18 and 19. The controller $\boldsymbol{x^1}$ shows lower IAE in both outputs (f_{11} y f_{12}). It also shows better ITAE over $T_{e,sec,out}$ (f_{13}). The nearly optimal solution ($\boldsymbol{x^2}$) shows better ITAE for the second output T_{SH} (f_{14} to f_{16}).

The control action Av of $\boldsymbol{x^1}$ is smoother (f_{18}), while the action of N is less smooth (f_{17}). Therefore $\boldsymbol{x^2}$ wins in four (f_{14} to f_{17}) and loses in four (f_{11} to f_{13} and f_{18}) aggregated objectives. It can be seen that there are two controllers with similar performances, and despite $\boldsymbol{x^2}$ being dominated by $\boldsymbol{x^1}$ no significant loss is observed in any of the proposed aggregate objectives.

The CIC2018 contest also proposes a second phase in which four new profiles of validation are provided for the setpoints sp_{Tsec} and sp_{TSH} and the disturbances $T_{e,sec,in}$ and $T_{c,sec,in}$ (see Figure 20 and 21) to compare the correct functioning of the chosen controllers with the initial profile. This implies a change of scenario for the problem posed in the first phase. In this way, we assess in simulation the robustness of the controllers compared, by analyzing a new scenario.

Figures 22 and 23 and Table 10 show the response and the values of the aggregate objectives obtained on the validation test. Solution $\boldsymbol{x^2}$ shows better IAE on outputs (f_{11} y f_{12}) on this trial. In addition, $\boldsymbol{x^2}$ shows an improvement in ITAE for both outputs (f_{13} a f_{16}). The most significant difference is

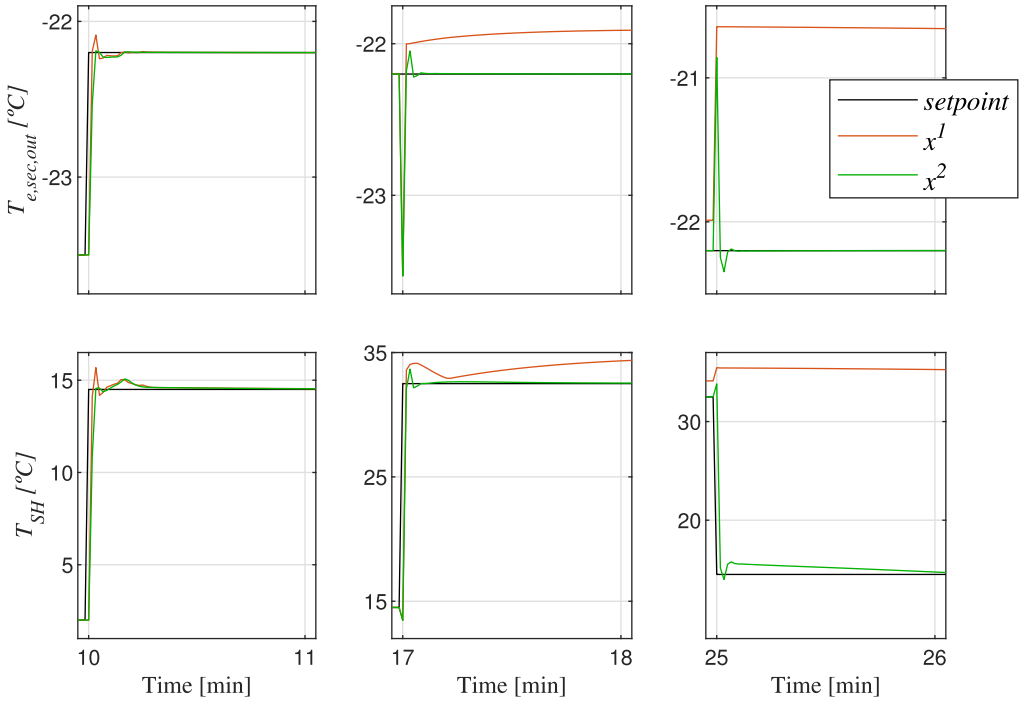


FIGURE 22. Comparison of the response for the controller validation test x^1 and x^2 for the CIC2018 control benchmark. Detail of the instants in time when changes are produced in the setpoints.

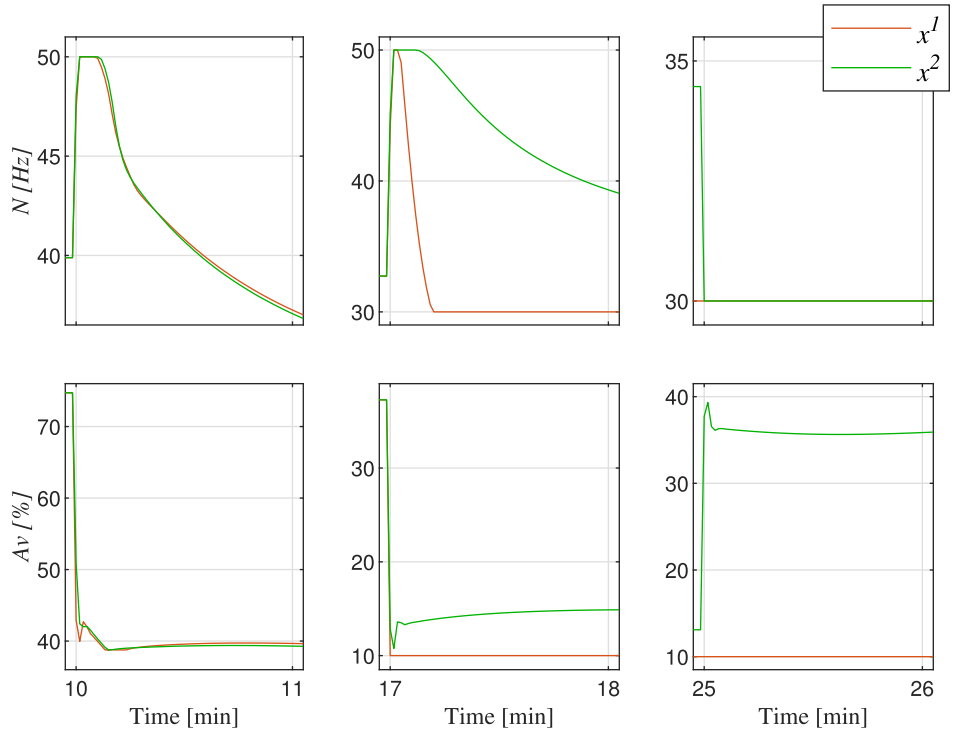


FIGURE 23. Comparison of the control actions for the controller validation test x^1 and x^2 for the CIC2018 control benchmark. Detail of the instants in time when changes in the setpoints were produced.

observed in the ITAE of the last step entered for T_{SH} (f_{16}). The difference is such that the controller $\boldsymbol{x^1}$ fails to reach the reference due to the saturation of the control action in that interval (see Figures 22 and 23). $\boldsymbol{x^2}$ reaches the reference and so obtains a performance that is much better in this interval. Both control actions are slightly less smooth for $\boldsymbol{x^2}$ (f_{17} and f_{18}). Nevertheless, the significant difference in performance (especially for f_{16}, interval 25 to 40 *min* in

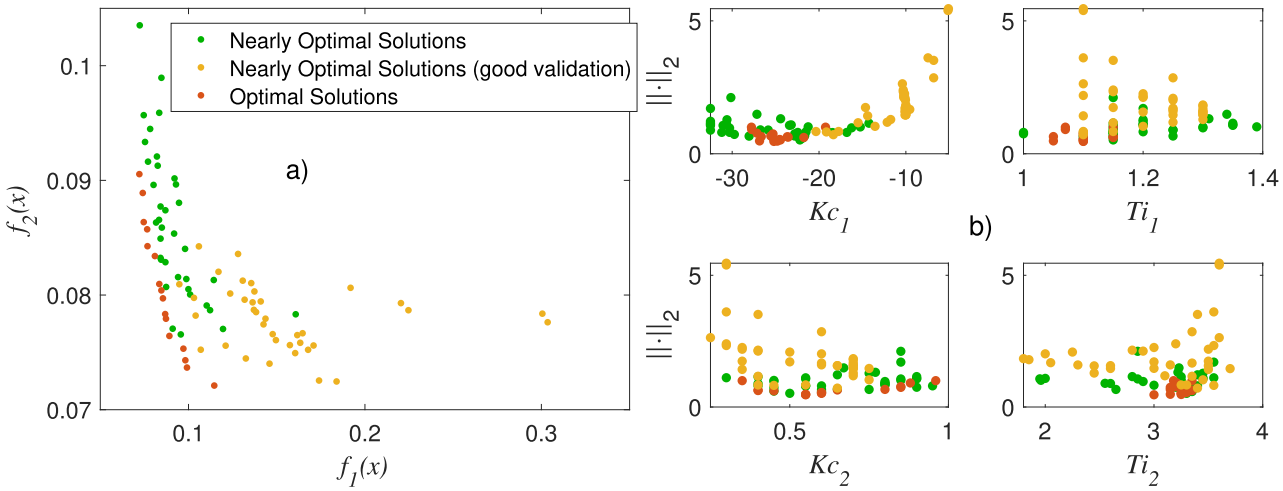


FIGURE 24. Set of optimal solutions (red) and nearly optimal solutions nondominated in their neighborhood (green and yellow) for the initial profile introduced in the CIC2018 control benchmark. The solutions in yellow (nearly optimals) perform well in the validation test.

TABLE 10. Values of the design objectives and the objectives aggregated for the controllers x^1 and x^2 in the validation test.

Controller	$f_{11}(\boldsymbol{x})$	$f_{12}(\boldsymbol{x})$	$f_{13}(\boldsymbol{x})$	$f_{14}(\boldsymbol{x})$	$f_1(\boldsymbol{x})$
$\boldsymbol{x^1}$	3.7361	2.2980	0.0065	0.0053	38.6
$\boldsymbol{x^2}$	**0.0156**	**0.0150**	**0.0042**	**0.0036**	**0.056**
Controller	$f_{15}(\boldsymbol{x})$	$f_{16}(\boldsymbol{x})$	$f_{17}(\boldsymbol{x})$	$f_{18}(\boldsymbol{x})$	$f_2(\boldsymbol{x})$
$\boldsymbol{x^1}$	1.7415	30.8361	**0.0120**	**0.0282**	**0.04**
$\boldsymbol{x^2}$	**0.0037**	**0.0134**	0.0173	0.0336	0.051

Figure 22) of both controllers makes us choose a nearly optimal solution ($\boldsymbol{x^2}$) in contrast to what could have been expected a priori.

To make a more detailed study of the solutions produced in the proposed MOP, we obtained its response for the defined validation test (Figure 24). In this study we can see that there is a set of nearly optimal solutions (solutions shown in yellow) that manage to reach the introduced profile in a similar way to $\boldsymbol{x^2}$. No optimal solution reaches this reference, as occurs with $\boldsymbol{x^1}$. The importance of finding nearly optimal solutions is therefore shown. Thanks to these solutions we have been able to analyze a new scenario for the validation test (for which there was no information initially). This analysis has enabled us to find nearly optimal (similar to optimal) controllers for the initially introduced profiles that perform significantly better than the optimum controllers in the validation test. This analysis could not have been carried out without redefining the MOP (and later obtaining the optimal controllers) in a classic MOP.

V. CONCLUSIONS

We have presented the optimal adjustment of controllers using multiobjective optimization techniques that take into account, in addition to the set of optimal solutions (Pareto front), the set of nearly optimal solutions nondominated in their neighborhood. The inclusion of these solutions in the decision-making stage increases the diversity of the set of relevant solutions available to the designer.

When there are many objectives, it is common to aggregate objectives to simplify the optimization process and decision phase. This situation is quite common in problems requiring an adjustment of controller parameters, especially in multivariable systems, where the number of objectives can grow exponentially with respect to the number of inputs and outputs.

When this happens, the aggregation process converts many of the controllers that are optimal in the complete MOP into nearly optimal controllers in the reduced MOP. Including all these controllers can hinder the decision-making stage unnecessarily, given that the controllers are very similar in the parameter space. Therefore, we propose keeping only the nearly optimal solutions that are nondominated in their neighborhood. These controllers have similar performances and different characteristics, and so may provide relevant information at the decision-making stage. This fact has been shown in the examples presented: the Wood & Berry distillation column and the CIC2018 control benchmark. In the first example, it is observed that a nearly optimal alternative obtain a greater robustness than a solution that dominate it. This new information enables the designer to opt for a nearly optimal alternative – given that both are very similar in the design objectives proposed in the MOP. In the second example, the CIC2018 control benchmark, a change of scenario is proposed to evaluate the controller using a validation test. In this case, it is seen how the set of optimal controllers obtains undesirable results in this new scenario. Because we have a set of alternatives nondominated in their neighborhood, we obtain a set of nearly optimal controllers (in a new neighborhood) that produces a good result in the new proposed scenario. This result is similar to the optimal results in the initial scenario included in the MOP.

Therefore, the importance of the study of nearly optimal alternatives nondominated in their neighborhood in the design of controllers has become clear.

REFERENCES

[1] 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, Feb. 2018, Art. no. 1792420.

[2] O. Schütze, M. Vasile, and C. A. C. Coello, "Computing the set of epsilon-efficient solutions in multiobjective space mission design," *J. Aerosp. Comput., Inf., Commun.*, vol. 8, no. 3, pp. 53–70, Mar. 2011.

[3] O. Schütze, C. A. C. Coello, and E.-G. Talbi, "Approximating the ϵ-efficient set of an mop with stochastic search algorithms," in *Advances in Artificial Intelligence* (Lecture Notes in Computer Science). 2007, pp. 128–138.

[4] V. Pareto, *Manual of Political Economy*. New York, NY, USA: A.M. Kelley, 1971.

[5] R. K. Wood and M. W. Berry, "Terminal composition control of a binary distillation column," *Chem. Eng. Sci.*, vol. 28, no. 9, pp. 1707–1717, 1973.

[6] 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* multiobjective evolutionary algorithm," *Struct. Multidisciplinary Optim.*, vol. 39, no. 2, pp. 203–215, Aug. 2008.

[7] G. R. Meza, X. B. Ferragud, J. S. Saez, and J. M. H. Durá, *Controller Tuning With Evolutionary Multiobjective Optimization a Holistic Multiobjective Optimization Design Procedure*. Cham, Switzerland: Springer, 2017.

[8] K. Deb, "Multi-objective optimization," in *Search Methodologies*. Boston, MA, USA: Springer, 2014, pp. 403–449.

[9] A. Konak, D. W. Coit, and A. E. Smith, "Multi-objective optimization using genetic algorithms: A tutorial," *Rel. Eng. Syst. Saf.*, vol. 91, no. 9, pp. 992–1007, Sep. 2006.

[10] T. N. L. Vu and M. Lee, "Multi-loop PI controller design based on the direct synthesis for interacting multi-time delay processes," *ISA Trans.*, vol. 49, no. 1, pp. 79–86, Jan. 2010.

[11] S. Skogestad and I. Postlethwaite, *Multivariable Feedback Control: Analysis and Design*, vol. 2. New York, NY, USA: Wiley, 2007, pp. 359–368.

[12] V. V. Kumar, V. S. R. Rao, and M. Chidambaram, "Centralized PI controllers for interacting multivariable processes by synthesis method," *ISA Trans.*, vol. 51, no. 3, pp. 400–409, May 2012.

[13] K. J. Åström, H. Panagopoulos, and T. Hägglund, "Design of PI controllers based on non-convex optimization," *Automatica*, vol. 34, no. 5, pp. 585–601, May 1998.

[14] G. Bejarano, J. A. Alfaya, D. Rodríguez, F. Morilla, and M. G. Ortega, "Benchmark for PID control of refrigeration systems based on vapour compression," *IFAC-PapersOnLine*, vol. 51, no. 4, pp. 497–502, 2017.

[15] C. B. G. Meyer, R. K. Wood, and D. E. Seborg, "Experimental evaluation of analytical and Smith predictors for distillation column control," *AIChE J.*, vol. 25, no. 1, pp. 24–32, Jan. 1979.

[16] S. Treiber, "Feasibility of decoupling in conventionally controlled distillation columns. Comments," *Ind. Eng. Chem. Fundam.*, vol. 21, no. 1, p. 99, Feb. 1982.

[17] R. N. Pawar and S. P. Jadhav, "Design of NDT and PSO based decentralised PID controller for wood-berry distillation column," in *Proc. IEEE Int. Conf. Power, Control, Signals Instrum. Eng. (ICPCSI)*, Sep. 2017, pp. 719–723.

[18] K. Miettinen, *Nonlinear Multiobjective Optimization*, vol. 1. Norwell, MA, USA: Kluwer, 1998.

[19] K. Deb, *Multi-Objective Optimization Using Evolutionary Algorithms*. Chichester, U.K.: Wiley, 2001.

[20] G. Reynoso-Meza, J. Sanchis, X. Blasco, and S. García-Nieto, "Physical programming for preference driven evolutionary multi-objective optimization," *Appl. Soft Comput.*, vol. 24, pp. 341–362, Nov. 2014.

[21] M. T. M. Emmerich and A. H. Deutz, "A tutorial on multiobjective optimization: Fundamentals and evolutionary methods," *Natural Comput.*, vol. 17, no. 3, pp. 585–609, Sep. 2018.

[22] J. Sanchis, M. Martínez, and X. Blasco, "Integrated multiobjective optimization and *a priori* preferences using genetic algorithms," *Inf. Sci.*, vol. 178, no. 4, pp. 931–951, Feb. 2008.

[23] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach," *IEEE Trans. Evol. Comput.*, vol. 3, no. 4, pp. 257–271, Nov. 1999.

[24] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: Improving the strength Pareto evolutionary algorithm," *TIK-Rep.*, vol. 103, 2001.

[25] E. Zitzler, K. Deb, and L. Thiele, "Comparison of multiobjective evolutionary algorithms: Empirical results," *Evol. Comput.*, vol. 8, no. 2, pp. 173–195, 2000.

[26] J. Branke, J. Branke, K. Deb, K. Miettinen, and R. Slowiński, *Multiobjective Optimization: Interactive and Evolutionary Approaches*, vol. 5252. Berlin, Germany: Springer Science & Business Media, 2008.

[27] Y. Collette, *Multiobjective Optimization: Principles and Case Studies*. Berlin, Germany: Springer Sciencie & Business Media, 2013.

[28] M. Vasile and M. Locatelli, "A hybrid multiagent approach for global trajectory optimization," *J. Global Optim.*, vol. 44, no. 4, pp. 461–479, Aug. 2008.

[29] K. Deb and A. Saha, "Finding multiple solutions for multimodal optimization problems using a multi-objective evolutionary approach," in *Proc. 12th Annu. Conf. Genetic Evol. Comput. (GECCO)*, vol. 10, Jul. 2010, pp. 447–454.

[30] J. J. Liang, C. T. Yue, and B. Y. Qu, "Multimodal multi-objective optimization: A preliminary study," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Jul. 2016, pp. 2454–2461.

[31] B.-Y. Qu, P. N. Suganthan, and J.-J. Liang, "Differential evolution with neighborhood mutation for multimodal optimization," *IEEE Trans. Evol. Comput.*, vol. 16, no. 5, pp. 601–614, Oct. 2012.

[32] P. Loridan, "ϵ-solutions in vector minimization problems," *J. Optim. Theory Appl.*, vol. 43, no. 2, pp. 265–276, 1984.

[33] D. J. White, "Epsilon efficiency," *J. Optim. Theory Appl.*, vol. 49, no. 2, pp. 319–337, May 1986.

[34] C. A. C. Coello, "A comprehensive survey of evolutionary-based multiobjective optimization techniques," *Knowl. Inf. Syst.*, vol. 1, no. 3, pp. 269–308, Aug. 1999.

[35] X. Blasco, J. M. Herrero, J. Sanchis, and M. Martínez, "A new graphical visualization of *n*-dimensional Pareto front for decision-making in multiobjective optimization," *Inf. Sci.*, vol. 178, no. 20, pp. 3908–3924, Oct. 2008.

[36] S. Skogestad and C. Grimholt, "The SIMC method for smooth PID controller tuning," in *PID Control in the Third Millennium* (Advances in Industrial Control). London, U.K.: Springer, 2012, pp. 147–175.

[37] R. C. Dorf and R. H. Bishop, *Modern Control Systems*. Reading, MA, USA: Addison-Wesley, 1998.

[38] R. E. Smith, S. Forrest, and A. S. Perelson, "Searching for diverse, cooperative populations with genetic algorithms," *Evol. Comput.*, vol. 1, no. 2, pp. 127–149, 1993.

[39] J. M. Herrero, "Identificación Robusta de Sistemas no Lineales mediante Algoritmos Evolutivos," Ph.D. dissertation, Dept. en Ingeniería de Sistemas y Automática, Editorial Univ. Politècnica de València, 2006.

[40] F. Lian, A. Chakrabortty, and A. Duel-Hallen, "Game-theoretic multiagent control and network cost allocation under communication constraints," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 2, pp. 330–340, Feb. 2017.

[41] K. Li, R. Wang, T. Zhang, and H. Ishibuchi, "Evolutionary many-objective optimization: A comparative study of the state-of-the-art," *IEEE Access*, vol. 6, pp. 26194–26214, 2017.

[42] N. Gunantara, "A review of multi-objective optimization: Methods and its applications," *Cogent Eng.*, vol. 5, no. 1, May 2018, Art. no. 1502242.

[43] V. Manousiouthakis, "A game theoretic approach to robust controller synthesis," *Comput. Chem. Eng.*, vol. 14, nos. 4–5, pp. 381–389, May 1990.

[44] D. Greiner, J. Periaux, J. M. Emperador, B. Galván, and G. Winter, "Game theory based evolutionary algorithms: A review with nash applications in structural engineering optimization problems," *Arch. Comput. Methods Eng.*, vol. 24, no. 4, pp. 703–750, Nov. 2016.

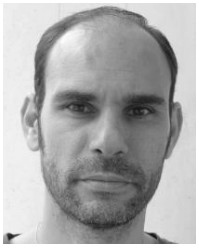
[45] A. E. Ohazulike and T. Brands, "Multi-objective optimization of traffic externalities using tolls," in *Proc. IEEE Congr. Evol. Comput.*, Jun. 2013, pp. 2465–2472.

[46] A. Chakrabortty and M. Arcak, "Robust stabilization and performance recovery of nonlinear systems with unmodeled dynamics," *IEEE Trans. Autom. Control*, vol. 54, no. 6, pp. 1351–1356, Jun. 2009.

[47] P. Kumar, W. Kohn, and Z. B. Zabinsky. (2017). "Near optimal Hamiltonian-control and learning via chattering." [Online]. Available: https://arxiv.org/abs/1703.06485

ALBERTO PAJARES received the B.S. and M.Sc. degrees in automatic and industrial computing from the Universitat Politècnica de València (UPV), Spain, in 2013 and 2014, respectively, where he is currently pursuing the Ph.D. degree. His main research interests are control and modeling of systems, process optimization, and multiobjective optimization techniques.

XAVIER BLASCO born in Paris, in 1966. He received the B.S. and Ph.D. degrees in industrial engineering from the Universitat Politècnica de València (UPV), Spain, 1991 and 1999, respectively, and the Diplome de Spécialisation en Génie Electrique from the Ecole Supérieure d'Electricité (SUPELEC), France, in 1992. Since 1994, he has been teaching with the Department of Systems Engineering and Automation, UPV, where he is currently a Full Professor. His research work is developed at the Institute of Automatic Control (ai2), UPV. His research interests include 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, where he is currently an Associate Professor with the Department of Systems Engineering and Automation. His main research interests are multivariable predictive control, process optimization, and computational intelligence methods for control engineering.

GILBERTO REYNOSO-MEZA received the B.Sc. degree in mechanical engineering from the Tecnológico de Monterrey, Campus Querétaro, México, in 2001, and the Ph.D. degree in automation from the Universitat Politècnica de València (UPV), Spain. He is currently with the Industrial and Systems Engineering Graduate Program (PPGEPS), Pontifical Catholic University of Parana (PUCPR), Brazil, as an Associate Professor. His main research interests are computational intelligence methods for control engineering and design, multi-objective optimization, many-objectives optimization, multi-criteria decision making, evolutionary algorithms, and machine learning.