ISA Transactions 139 (2023) 143–155

Contents lists available at ScienceDirect

ISA Transactions

journal homepage: www.elsevier.com/locate/isatrans

Research article

Advantages of using relevant nearly optimal solutions in multi-objective tuning of robust controllers

Alberto Pajares *, Xavier Blasco, Juan Manuel Herrero, Uriel Veyna

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

ARTICLE INFO

Article history:
Received 23 May 2022
Received in revised form 9 May 2023
Accepted 9 May 2023
Available online 15 May 2023

Keywords:
Multi-objective optimization problem
Uncertainty
Robustness
Controller tuning

ABSTRACT

This paper presents a new approach for the multiobjective optimal design of robust controllers in systems with stochastic parametric uncertainty. Traditionally, uncertainty is incorporated into the optimization process. However, this can generate two problems: (1) low performance in the nominal scenario; and (2) high computational cost. For the first point, it is possible to ensure that the controllers produce an acceptable performance for the nominal scenario in exchange for being lightly robust. For the second point, the methodology proposed in this work reduces the computational cost significantly. This approach addresses uncertainty by analyzing the robustness of optimal and nearly optimal controllers in the nominal scenario. The methodology guarantees obtaining controllers that are similar/neighboring to lightly robust controllers. Two examples of controller design are shown: one for a linear model and another for a nonlinear model. Both examples demonstrate the usefulness of the proposed new approach.

1. Introduction

During the design process of a control system, the designer must choose a structure for the controller and adjust its parameters. As in many other engineering designs, the parameter tuning of the controller can be formulated as an explicit optimization problem [1]. In this case, the objectives that will measure the performance of the controller are chosen, for example: the overshoot; rise or peak time; settling time; tracking error; and control effort for reference tracking or disturbance rejection. Many of these objectives conflict and result in a multi-objective optimization problem (MOP [2–4]).

In addition, it is common to have a certain degree of uncertainty in the process models used for controller design and tuning. This uncertainty will have an impact on the performance of the designed control. Consequently, an optimal controller could be nearly optimal or totally inadequate when applied in the actual process. Therefore, to obtain a good controller, two characteristics are desirable: good performance in the nominal model and robustness (insensitivity to model uncertainties).

For this purpose, it is helpful for the designer to analyze and incorporate uncertainties in the controller tuning procedure. When considering parametric uncertainty, an effective approach involves using stochastic programming [5]. In this case, the model parameters are considered as random variables, where each combination of parameters (possible model) is considered as a scenario. This approach is especially useful with nonlinear models [6]. In this way, it is possible to choose a robust controller with an adequate performance in all scenarios.

MOPs and uncertainty have been deeply and independently studied in the literature. However, the development of a robust optimization theory for uncertain multi-objective optimization problems ($\mathcal{U}$MOP) has been initiated only recently [7–9].

A widely extended alternative is the point-based minmax robust efficiency [7]. Under this approach, the supremum value for each objective function is optimized separately for each controller and considering all scenarios. However, this may generate controllers with low performance in the nominal scenario. As a consequence, the concept of light robustness for $\mathcal{U}$MOPs [7,9–11] arises. This approach ensures that in exchange for being lightly robust, the controllers have an acceptable performance in the nominal scenario. Finding these robust controllers assumes incorporating in the optimization process the scenarios to be considered (including the nominal scenario). This involves a high computational cost that increases linearly with the number of scenarios and sometimes makes the optimization problem unaffordable.

This paper presents a new proposal that is computationally efficient for characterizing lightly robust controllers. For this purpose, an MOP is proposed on the nominal scenario, and both optimal and nearly optimal controllers [12–14] (also called approximate or ϵ-efficient controllers) that are non-dominated in

* Corresponding author.
E-mail address: alpafer1@upv.es (A. Pajares).

https://doi.org/10.1016/j.isatra.2023.05.003

their neighborhood [15] are characterized. The analysis of these controllers to determine which are robust is left for the decision-making phase. In this type of problem, computational costs are kept low by considering a limited but relevant number of controllers in the optimization stage. There are already algorithms (e.g. nevMOGA [16]) that enable a proper and simultaneous characterization of sets of optimal and nearly optimal controllers.

Two approaches are compared to show the benefits of using relevant nearly optimal solutions for obtaining lightly robust controllers. Approach 1 (traditional way) incorporates uncertainty in the optimization stage, adding constraints to avoid overly degrading performance in the nominal scenario. Approach 2 (proposed in this work) obtains the optimal and relevant nearly optimal solutions with the optimization process for the nominal model and identifies the most robust solutions (which will characterize the lightly robust controllers) in the decision-making phase.

Both approaches are applied to two examples: a linear SISO model (which is an illustrative example) and a nonlinear MIMO model. The tuned controllers are PID type. Both examples show that approach 2 characterizes the lightly robust controllers determined by approach 1: but with a significantly lower computational cost. In addition, as shown in the second example, MOPs can have multimodalities, i.e., different controllers producing the same values in the design objectives. These multimodalities are very common when the objectives are composed of an aggregation of partial functions. For example, in the control of multivariable systems, an objective could be the total control effort obtained from an aggregation of the control effort of each input in the process. Approach 2 obtains multimodal solutions if they are significantly different in the parameter space and these will be available for the designer to study their robustness and other characteristics. These solutions may vary the designer's final decision.

Therefore, the main contribution of this paper are the following:

- To present a novel methodology to characterize lightly robust controllers.
- To guarantee mathematically that this methodology obtains controllers similar/neighboring to lightly robust controllers.
- To reduce the computational cost of obtaining lightly robust controllers, with acceptable performance in the nominal scenario.
- Finally, to apply and validate the methodology presented in this work.

This paper is structured as follows. Section 2 presents some definitions that are used to describe MOPs, $\mathcal{U}$MOPs, and lightly robust controllers. Section 3 defines the robust design approaches that are compared in this work. Section 4 presents examples of a lightly robust controller design for a linear model, and Section 5 presents the example of the nonlinear and multivariable model. Finally, Section 6 discusses the conclusions.

2. Background

This work seeks to tune a controller $\boldsymbol{x}$ using a multi-objective optimization approach considering uncertainty in the process model. We assume that the uncertainty is associated with the parameters $\boldsymbol{\xi}$ of the model. In the problem formulation, uncertainty is incorporated as scenarios which are defined by a combination of parameter values $\boldsymbol{\xi} = [\xi_1, \xi_2, \ldots, \xi_p]$.

Given a scenario $\boldsymbol{\xi}$, a multi-objective optimization problem[1] can be defined as follows:

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

where Q can be defined[2] as:

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

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

The function $\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\xi})$ incorporates different objectives that measure the quality of control (performance, control efforts, etc.).

Definition 1 (*Dominance [3,17]*)**.** Given a scenario $\boldsymbol{\xi}$, a decision vector $\boldsymbol{x^1}$ is dominated by any other decision vector $\boldsymbol{x^2}$ if $f_i(\boldsymbol{x^2}, \boldsymbol{\xi}) \le f_i(\boldsymbol{x^1}, \boldsymbol{\xi})$ for all $i \in [1, \ldots, m]$ and $f_j(\boldsymbol{x^2}, \boldsymbol{\xi}) < f_j(\boldsymbol{x^1}, \boldsymbol{\xi})$ 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^{ξ}) is the set of solutions in Q that are nondominated by another solution in Q for a scenario $\boldsymbol{\xi}$:

$$P_Q^{\xi} := \{\boldsymbol{x} \in Q |\ \nexists \boldsymbol{x^o} \in Q : \boldsymbol{x^o} \preceq \boldsymbol{x}\}$$

In addition to the optimal solutions P_Q^{ξ} (see Definition 2), there is a set of nearly optimal solutions P_{ϵ}^{ξ} (see Definition 4) that can be useful to the designer.

Definition 3 ($-\epsilon$-*Dominance [18]*)**.** Defined $\boldsymbol{\epsilon} = [\epsilon_1, \ldots, \epsilon_m]$ as the maximum acceptable performance degradation. Given a scenario $\boldsymbol{\xi}$, a decision vector $\boldsymbol{x^1}$ is $-\epsilon$-dominated by another decision vector $\boldsymbol{x^2}$ if $f_i(\boldsymbol{x^2}, \boldsymbol{\xi}) + \epsilon_i \le f_i(\boldsymbol{x^1}, \boldsymbol{\xi})$ for all $i \in [1, \ldots, m]$ and $f_j(\boldsymbol{x^2}, \boldsymbol{\xi}) + \epsilon_i < f_j(\boldsymbol{x^1}, \boldsymbol{\xi})$ for at least one j, $j \in [1, \ldots, m]$. This is denoted by $\boldsymbol{x^2} \preceq_{-\epsilon} \boldsymbol{x^1}$.

Definition 4 (ϵ-*Efficiency [19]*)**.** The set of ϵ-efficient solutions (denoted by P_{ϵ}^{ξ}) is the set of solutions in Q which are not $-\epsilon$-dominated by another solution in Q for a scenario $\boldsymbol{\xi}$:

$$P_{\epsilon}^{\xi} := \{\boldsymbol{x} \in Q |\ \nexists \boldsymbol{x^o} \in Q : \boldsymbol{x^o} \preceq_{-\epsilon} \boldsymbol{x}\}$$

However, finding all the nearly optimal solutions can considerably increase the number of alternatives obtained and this increases the computational cost of the optimization algorithms and makes the decision stage more difficult. To restrict the number of solutions, it is proposed to consider only non-dominated solutions in their neighborhood P_n^{ξ} (see Definition 7). P_n^{ξ} characterizes all neighborhoods where nearly optimal solutions exist. These solutions are potentially helpful since they offer the designer nearly optimal alternatives in the objective space but differ significantly in the parameter space. In this way, the diversity of solutions in the parameter space is preserved without excessively increasing the number of possible alternatives.

Definition 5 (*Neighborhood [20]*)**.** 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 6 (*n-Dominance [20]*)**.** A decision vector $\boldsymbol{x^1}$ is n-dominated by another decision vector $\boldsymbol{x^2}$ if they are neighboring solutions (Definition 5) and $\boldsymbol{x^2} \preceq \boldsymbol{x^1}$. This is denoted by $\boldsymbol{x^2} \preceq_n \boldsymbol{x^1}$.

Definition 7 (*n-Efficiency [20]*)**.** The set of n-efficient solutions (denoted by P_n^{ξ}) is the set of solutions of P_{ϵ}^{ξ} which are not n-dominated by another solution in P_{ϵ}^{ξ} for a scenario $\boldsymbol{\xi}$:

$$P_n^{\xi} := \{\boldsymbol{x} \in P_{\epsilon}^{\xi} |\ \nexists \boldsymbol{x^o} \in P_{\epsilon} : \boldsymbol{x^o} \preceq_n \boldsymbol{x}\}$$

[1] A maximization problem can be easily converted into a minimization problem. For each of the objectives $\boldsymbol{f_i}(\boldsymbol{x}, \boldsymbol{\xi})$ that have to be maximized it can be changed by $-\boldsymbol{f_i}(\boldsymbol{x}, \boldsymbol{\xi})$.

[2] In general, any other constraints can be included.

Uncertainty can be taken into account in the optimization or decision-making stages. When uncertainty is considered in the decision-making stage, the design objectives are evaluated under a nominal model $\boldsymbol{\xi}^{\mathbf{0}}$ (by means of Eq. (1)). Once the Pareto set has been obtained P^{ξ^0} (or $P_n^{\xi^0}$ when nearly optimal solutions are considered), it is possible to analyze the uncertainty of these solutions. For this purpose, a set of scenarios is defined as $\mathcal{U}$. Subsequently, the performance of the solutions P^{ξ^0} (or $P_n^{\xi^0}$) is analyzed for the different scenarios $\boldsymbol{\xi} \in \mathcal{U}$.

When uncertainty is incorporated into the optimization process, a uncertain multi-objective optimization problem ($\mathcal{U}$MOP) arises where the design objectives contemplate such uncertainty.

Definition 8. An uncertain multi-objective optimization problem $\mathcal{P}(\mathcal{U}) = (\mathcal{P}(\boldsymbol{\xi}), \boldsymbol{\xi} \in \mathcal{U})$ is defined as the family of parametrized problems [7]:

$$\mathcal{P}(\boldsymbol{\xi}) : \min_{\boldsymbol{x}\in Q} \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\xi}) \tag{3}$$

where $\mathcal{P}(\boldsymbol{\xi})$ is an instance of $\mathcal{P}(\mathcal{U})$. Consequently, an $\mathcal{U}$MOP is defined as a family of optimization problems where each problem obtains a different Pareto set. From here, it would be necessary to analyze all these fronts to establish which solutions are the most robust. Nevertheless, this analysis is not feasible since each front offers different controllers in differing scenarios.

To obtain a robust controller (that does not excessively degrade performance for all scenarios) it is necessary to evaluate such a controller in the optimization stage for all scenarios considered. Thus, a controller $\boldsymbol{x}$ generates a set of points in the objective space, one for each scenario $\boldsymbol{\xi}$ contemplated in $\mathcal{U}$:

$$\mathcal{F}_{\mathcal{U}} := \{\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\xi}) : \boldsymbol{\xi} \in \mathcal{U}\}$$

Once this set is obtained, it is possible to make an analysis using different robustness criteria. In the literature, different possibilities are presented [7–9]: flimsily and highly robust efficiency; set-based minmax robust efficiency; lightly robust efficiency; point-based minmax robust efficiency; etc. These criteria enable transforming the $\mathcal{U}$MOP into a deterministic problem [7,8] and so solving a single optimization problem for the set of scenarios $\mathcal{U}$.

The robustness criteria can be set-based or point-based. In a set-based approach, each solution has a representative set of $\mathcal{U}$ (obtained from analyzing $\mathcal{F}_{\mathcal{U}}$). The concept of dominance between sets must be defined for the analysis of these representative sets, and the multi-objective optimization algorithms must be modified. In the case of point-based analysis, it is necessary to obtain a point representing $\mathcal{U}$ (obtained from $\mathcal{F}_{\mathcal{U}}$). With this representative, the definition of dominance is used (see Definition 1), and the Pareto set is obtained as in a traditional MOP (see Definition 2).

A widely extended alternative is the point-based minmax robust efficiency [7]. Under this approach, each solution $\boldsymbol{x}$ defines as representative the supremum value for every objective function separately and considering all scenarios $\boldsymbol{\xi} \in \mathcal{U}$. In this paper, we will use this definition to obtain robust solutions in a $\mathcal{U}$MOP. Thus, we can define a robust $\mathcal{U}$MOP (hereinafter $\mathcal{RU}$MOP) as:

$$\min_{\boldsymbol{x}\in Q} \boldsymbol{f}_{\mathcal{U}}^{\boldsymbol{max}}(\boldsymbol{x}) \tag{4}$$

$$\boldsymbol{f}_{\mathcal{U}}^{\boldsymbol{max}}(\boldsymbol{x}) = [\sup_{\boldsymbol{\xi}\in\mathcal{U}} f_1(\boldsymbol{x}, \boldsymbol{\xi}),\ \sup_{\boldsymbol{\xi}\in\mathcal{U}} f_2(\boldsymbol{x}, \boldsymbol{\xi}), \ldots, \sup_{\boldsymbol{\xi}\in\mathcal{U}} f_m(\boldsymbol{x}, \boldsymbol{\xi})]$$

The Pareto set obtained with this $\mathcal{RU}$MOP will be denoted as $P_{\mathcal{R}}^{\mathcal{U}}$. However, optimizing the supremum value can significantly degrade the performance in the nominal scenario. To avoid this problem, the concept of light robustness for $\mathcal{U}$MOPs [7,9] arises, where the obtained robust solutions are guaranteed to have acceptable performance in the nominal case. This concept

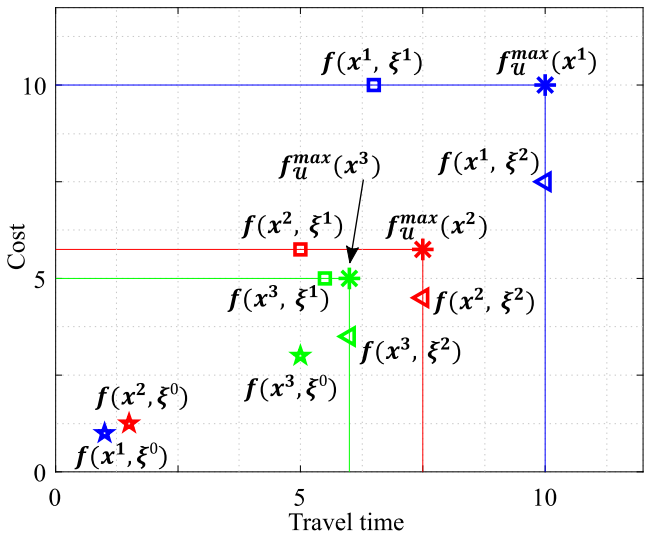

Fig. 1. Illustrative example of a $\mathcal{U}$MOP problem. There are three possible scenarios: nominal $\boldsymbol{\xi}^{\mathbf{0}}$ (stars); $\boldsymbol{\xi}^{\mathbf{1}}$ (squares); and $\boldsymbol{\xi}^{\mathbf{2}}$ (triangles). The asterisks show the supremum value $\boldsymbol{f}_{\mathcal{U}}^{\boldsymbol{max}}$ of each solution.

for single-objective optimization problems has been introduced in [10]. Furthermore, this concept was generalized in [11]. Based on the point-based minmax robust efficiency concept, a lightly robust $\mathcal{U}$MOP (hereafter $\mathcal{LRU}$MOP) is defined as:

$$\min_{\boldsymbol{x}\in Q} \boldsymbol{f}_{\mathcal{U}}^{\boldsymbol{max}}(\boldsymbol{x})$$
$$s.t.\ x \in P_{\epsilon}^{\xi^0} \tag{5}$$

The Pareto set obtained with this $\mathcal{LRU}$MOP will be denoted as $P_{\mathcal{LR}}^{\mathcal{U}}$.

Let us consider an example. Suppose we want to go from point A to point B. There are three possible paths: (1) x^1 is the shortest path through the city center, (2) x^2 is the intermediate path with a single-lane road, and (3) x^3 is the longest path with a wide multi-lane road and a toll. We want to minimize travel time and travel cost (design objectives).

For normal conditions (no traffic, no rain), a nominal scenario ξ^0 is defined. Fig. 1 shows the performance obtained by each solution for ξ^0 ($\boldsymbol{f}(x^i, \xi^0)$, stars). Under this scenario, path x^1 has a shorter travel time and lower cost ($\boldsymbol{f}(x^1, \xi^0) = [1, 1]$). x^2 has a slightly worse performance than x^1 ($\boldsymbol{f}(x^2, \xi^0) = [1.5, 1.25]$). Finally, x^3 has a significantly higher travel time and cost than the other options ($\boldsymbol{f}(x^3, \xi^0) = [5, 3]$).

However, in addition to the nominal scenario ξ^0, there are two possible alternatives: (1) it starts raining (scenario ξ^1) and (2) there is a traffic jam (scenario ξ^2). In Fig. 1, the performance of each solution for the two new scenarios proposed ($\boldsymbol{f}(x^i, \xi^1)$, squares, and $\boldsymbol{f}(x^i, \xi^2)$, triangles) is plotted. In addition, the supremum value for every objective function is plotted separately for the set of scenarios proposed ($\boldsymbol{f}_{\mathcal{U}}^{\boldsymbol{max}}(x^i)$, asterisks). For the last two scenarios, the path x^3 has a slightly worse performance compared to the nominal scenario ($\boldsymbol{f}_{\mathcal{U}}^{\boldsymbol{max}}(x^3) = [6, 5]$). This is due to the fact that it is a wide road with several lanes, and therefore, it is less sensitive to rain and traffic (ξ^1 and ξ^2). On the other hand, x^1 is very sensitive to the scenarios ξ^1 and ξ^2 because the road crosses the city center and is therefore susceptible to traffic jams ($\boldsymbol{f}_{\mathcal{U}}^{\boldsymbol{max}}(x^1) = [10, 10]$). Finally, x^2 obtains a better $\boldsymbol{f}_{\mathcal{U}}^{\boldsymbol{max}}$ than x^1 and worse than x^3 for the scenarios ($\boldsymbol{f}_{\mathcal{U}}^{\boldsymbol{max}}(x^2) = [7.5, 5.75]$).

Based on the point-based minmax robust efficiency concept, x^3 is the most robust solution. However, this solution performs significantly worse than its contenders in the nominal case ξ^0

(the most probable one). To avoid this significant loss, we use the concept of light robustness. Under this concept, x^2 can be considered a lightly robust solution because it has the best $\boldsymbol{f}_{\mathcal{U}}^{\boldsymbol{max}}$ with a nearly optimal behavior in the nominal scenario ξ^0 (slightly worse than x^1).

3. Methodology proposed

In the previous section, alternatives for considering robustness in a $\mathcal{U}$MOP have been mentioned. In general, uncertainty can be considered in either: (1) the optimization stage; or (2) the decision-making phase. Both approaches will be stated and compared below.

3.1. First approach

This approach considers uncertainty at the optimization stage. Based on the concept of point-based minmax robust efficiency, a $\mathcal{RU}$MOP can be defined as shown in (4). However, these solutions can be significantly degraded in the nominal scenario. To avoid this problem, the $\mathcal{LRU}$MOP (lightly robust $\mathcal{U}$MOP) arises, where in addition to robust, the solutions guarantee an acceptable performance in the nominal model. To obtain these solutions, we proceed as follows:

1. Using the nominal scenario $\boldsymbol{\xi^0}$ we obtain $P^{\xi^0,*}$ as an approximation to the Pareto set P^{ξ^0} (see MOP defined in (1)).
2. We define $\boldsymbol{\epsilon}$ as the maximum acceptable loss over design objectives for the nominal scenario.
3. We obtain $P_{\mathcal{LR}}^{\mathcal{U},*}$ as an approximation to the set of lightly robust solutions $P_{\mathcal{LR}}^{\mathcal{U}}$ (see $\mathcal{LRU}$MOP defined in (5))

In this case, for each controller analyzed in the optimization process, it is first checked if it has a nearly optimal performance in the nominal scenario. If so, then $\boldsymbol{f}_{\mathcal{U}}^{\boldsymbol{max}}(\boldsymbol{x})$ for the set of defined scenarios is obtained.

The advantage of approach 1 is that lightly robust controllers are obtained, but it has a high computational cost because:

- It needs to carry out a pre-optimization to determine $P^{\xi^0,*}$.
- It needs to carry out an optimization, considering the set of scenarios $\mathcal{U}$, to determine $P_{\mathcal{LR}}^{\mathcal{U},*}$.

To obtain the approximations to the Pareto sets $P^{\xi^0,*}$ and $P_{\mathcal{LR}}^{\mathcal{U},*}$ the algorithm evMOGA[3] is used — although any other multiobjective optimization algorithm could be used.

3.2. Second approach

This approach analyzes uncertainty at the decision-making stage. In this case, the following steps are followed:

1. We define $\boldsymbol{\epsilon}$ as the maximum acceptable loss over the design objectives for the nominal scenario.
2. Neighborhood $\boldsymbol{n}$ is defined as the maximum distance between neighboring controllers.
3. An MOP is defined for the nominal scenario $\boldsymbol{\xi^0}$ (see MOP (1)). But now, in addition to the optimal controllers, the optimization also obtains $P_n^{\xi^0,*}$.
4. In the decision-making phase, the robustness is evaluated on $P_n^{\xi^0,*}$, through its representative $\boldsymbol{f}_{\mathcal{U}}^{\boldsymbol{max}}(\boldsymbol{x})$ and a robust controller is chosen.

The set $P_n^{\xi^0}$ characterizes all neighborhoods where optimal or nearly optimal solutions exist for the nominal scenario ($P_\epsilon^{\xi^0}$). $P_n^{\xi^0}$ will always obtain at least one neighboring solution of any robust solution $\boldsymbol{x} \in P_\epsilon^{\xi^0}$ (regardless of what is considered a robust solution). Therefore, we can affirm that $P_n^{\xi^0}$ characterizes any robust set $\mathcal{R} \subset P_\epsilon^{\xi^0}$ (regardless of the concept of robustness applied). This statement is proved mathematically by Theorem 1 with the sets $P_n^{\xi^0}$ and $P_{\mathcal{LR}}^{\mathcal{U}}$. This theorem is extrapolable to any robust set $\mathcal{R} \subset P_\epsilon^{\xi^0}$. Moreover, the characterization will be better or worse depending on the neighborhood $\boldsymbol{n}$ defined by the designer. Therefore, it is the designer who defines how similar will be, in the worst case, the set $P_{\boldsymbol{n}}^{\xi^0}$ obtained and any set $\mathcal{R} \subset P_\epsilon^{\xi^0}$.

This approach has a low computational cost since it is only necessary to evaluate the set of scenarios $\mathcal{U}$ over a bounded set of controllers $|P_n^{\xi^0,*}|$ (number of optimal and nearly optimal controllers obtained for the nominal scenario $P_n^{\xi^0,*}$). Moreover, the relevant nearly optimal controllers are significantly different from the optimal controllers in the parameter space, and this can produce significant differences in their robustness.

The limitations of this proposed approach are:

- Using a specific algorithm that characterizes $P_n^{\xi^0,*}$
- Definition of the neighborhood $\boldsymbol{n}$
- Number of solutions to consider in the optimization stage.

The need to use an algorithm that can determine $P_n^{\xi^0,*}$ (optimal and nearly optimal solutions) is a handicap in this approach. In this work, the algorithm nevMOGA[4] is used. nevMOGA was defined and validated in [20], showing that it is stochastically robust in characterizing the set of optimal and nearly optimal solutions non-dominated in its neighborhood. In addition, a statistical analysis of three archivers (strategy that selects and updates a solution set during the evolutionary process) used by algorithms characterizing optimal and nearly optimal solutions is performed in [22]. There it is shown that the archiver used by nevMOGA obtains a better approximation, with lower variability, and fewer solutions. The definition of the neighborhood $\boldsymbol{n}$ is not trivial when the decision variables are physically meaningless. For the correct definition of the neighborhood $\boldsymbol{n}$, it is recommended to perform a sensitivity analysis of the parameters and observe which variation produces significant changes in the system. This approach analyzes a wider number of solutions in the optimization process (optimal and nearly optimal). Therefore, we must be careful with the number of solutions considered, if we want to avoid a high computational cost. Choosing an excessively high value of $\boldsymbol{\epsilon}$, or an excessively low value of the neighborhood $\boldsymbol{n}$, can cause an excessive number of solutions to be considered, and with it, a high computational cost. A large number of design objectives can also lead to high computational cost.

3.3. Analysis of the defined approaches

As mentioned above, $P_n^{\xi^0}$ characterizes any robust set $\mathcal{R} \subset P_\epsilon^{\xi^0}$. This statement is proved for the sets $P_n^{\xi^0}$ and $P_{\mathcal{LR}}^{\mathcal{U}}$ by means of Theorem 1. A Venn diagram[5] [23] is shown in Fig. 2 to help understand Theorem 1 and some of its lemmas.

Lemma 1. *In a $\mathcal{LRU}$MOP (defined in (5)), the set $P_{\mathcal{LR}}^{\mathcal{U}}$ belongs to the set $P_\epsilon^{\xi^0}$ ($P_{\mathcal{LR}}^{\mathcal{U}} \subset P_\epsilon^{\xi^0}$).*

[3] evMOGA [21] available on Matlabcentral https://es.mathworks.com/matlabcentral/fileexchange/31080-ev-moga-multiobjective-evolutionary-algorithm?s_tid=srchtitle_evMOGA_1.

[4] nevMOGA [20] available on Matlabcentral https://es.mathworks.com/matlabcentral/fileexchange/71448-nevmoga-multiobjective-evolutionary-algorithm.

[5] A Venn diagram is a figure that uses overlapping circles to illustrate the logical relationships between two or more sets of items.

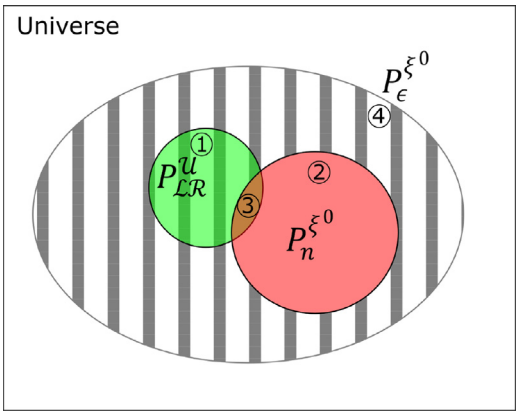

Fig. 2. Venn diagram for the sets $P_{\mathcal{LR}}^{\mathcal{U}}$ (1), $P_n^{\xi^0}$ (2) and $P_{\epsilon}^{\xi^0}$ (4). $P_{\mathcal{LR}}^{\mathcal{U}} \cup P_n^{\xi^0}$ forms the area (3).

Lemma 2. *The set of optimal and non-dominated nearly optimal solutions in its neighborhood $P_n^{\xi^0}$ (see Definition 7) belongs to the set $P_{\epsilon}^{\xi^0}$ ($P_n^{\xi^0} \subset P_{\epsilon}^{\xi^0}$).*

Lemma 3. *If a solution $\boldsymbol{x} \in P_n^{\xi^0}$, there is no other solution $\boldsymbol{x^o} \in P_{\epsilon}^{\xi^0}$ that dominates it in its neighborhood (see Definition 7, if $\boldsymbol{x} \in P_n^{\xi^0} \Rightarrow \nexists \boldsymbol{x}^\circ \in P_{\epsilon}^{\xi^0} : \boldsymbol{x}^\circ \preceq_n \boldsymbol{x}$).*

Lemma 4. *Based on Definition 7 and Lemma 3, if a solution $\boldsymbol{x}$ belongs to the set $P_{\epsilon}^{\xi^0}$, and does not belong to the set $P_n^{\xi^0}$, there exists another solution $\boldsymbol{x^o}$ belonging to the set $P_n^{\xi^0}$ that dominates it in its neighborhood (if $\boldsymbol{x} \in (P_{\epsilon}^{\xi^0} - P_n^{\xi^0}) \Rightarrow \exists \boldsymbol{x}^\circ \in P_n^{\xi^0} : \boldsymbol{x}^\circ \preceq_n \boldsymbol{x}$).*

Theorem 1. *Any lightly robust solution $\boldsymbol{x} \in P_{\mathcal{LR}}^{\mathcal{U}}$ has at least one neighboring solution in the set of optimal and non-dominated nearly optimal solutions in their neighborhood in the nominal scenario $P_n^{\xi^0}$ ($\forall x \in P_{\mathcal{LR}}^{\mathcal{U}}, \exists x^o \in P_n^{\xi^0} | x =_n x^o$).*

Proof. There are two possible cases for solution $\boldsymbol{x} \in P_{\mathcal{LR}}^{\mathcal{U}}$:

- (Case 1) $\boldsymbol{x} \in (P_{\mathcal{LR}}^{\mathcal{U}} \cap P_n^{\xi^0}) \Rightarrow \exists x^o \in P_n^{\xi^0} | x =_n x^o$
- (Case 2) $\boldsymbol{x} \in (P_{\mathcal{LR}}^{\mathcal{U}} - P_n^{\xi^0})$. If $\boldsymbol{x} \in P_{\mathcal{LR}}^{\mathcal{U}}$ then $x \in P_{\epsilon}^{\xi^0}$ (Lemma 1). If $\boldsymbol{x} \in (P_{\epsilon}^{\xi^0} - P_n^{\xi^0})$ then $\exists x^o \in P_n^{\xi^0} | x^o \preceq_n x$ (Lemma 4) $\Rightarrow \exists x^o \in P_n^{\xi^0} | x =_n x^o$.

Therefore, we can affirm that:

$\forall x \in P_{\mathcal{LR}}^{\mathcal{U}}, \exists x^o \in P_n^{\xi^0} | x =_n x^o$ □

Theorem 1 starts from four true lemmas in any MOP. The sets $P_{\mathcal{LR}}^{\mathcal{U}}$ and $P_n^{\xi^0}$ belong to the set of optimal and nearly optimal solutions in the nominal scenario $P_{\epsilon}^{\xi^0}$ (see sets (1), (2) and (4) in Fig. 2). Lemma 3 states that given any optimal or non-dominated nearly optimal solution in its neighborhood in the nominal scenario ($\boldsymbol{x} \in P_n^{\xi^0}$), there is no other optimal or nearly optimal solution in the nominal scenario that dominates it in its neighborhood ($\nexists \boldsymbol{x^o} \in P_{\epsilon}^{\xi^0} : \boldsymbol{x}^\circ \preceq_n \boldsymbol{x}$). Lemma 4 states that given an optimal or nearly optimal solution in the nominal scenario ($\boldsymbol{x} \in P_{\epsilon}^{\xi^0}$), which is not an optimal or nearly optimal solution non-dominated in its neighborhood ($\boldsymbol{x} \notin P_n^{\xi^0}$), there is another solution that dominates it in its neighborhood ($\boldsymbol{x^o} \in P_n^{\xi^0} : \boldsymbol{x}^\circ \preceq_n \boldsymbol{x}$).

Once the previously introduced lemmas are assumed, a solution $\boldsymbol{x} \in P_{\mathcal{LR}}^{\mathcal{U}}$ can: (1) also belong to the set $P_n^{\xi^0}$ (set (3) in Fig. 2), (2) not belong to the set $P_n^{\xi^0}$ (set (1) in Fig. 2). In the first case, the solution $\boldsymbol{x}$ is part of the sets $P_{\mathcal{LR}}^{\mathcal{U}}$ and $P_n^{\xi^0}$. Therefore, there is obviously a solution in $P_n^{\xi^0}$ neighboring $\boldsymbol{x}$ ($\boldsymbol{x}$ itself). In the second case, the conclusion is not so trivial. Firstly, if $\boldsymbol{x} \in P_{\mathcal{LR}}^{\mathcal{U}}$ then $\boldsymbol{x} \in P_{\epsilon}^{\xi^0}$ (Lemma 1). Secondly, if $\boldsymbol{x} \in (P_{\epsilon}^{\xi^0} - P_n^{\xi^0})$ (shaded area with black lines in Fig. 2), there is at least one solution $\boldsymbol{x^o} \in P_n^{\xi^0}$ such that $\boldsymbol{x^o}$ dominates $\boldsymbol{x}$ (Lemma 4) in its neighborhood. Therefore, it can be stated that any solution in $P_{\mathcal{LR}^{\mathcal{U}}}$ has a neighboring solution in $P_n^{\xi^0}$. This theorem can be extrapolated to any robust set $\mathcal{R} \subset P_{\epsilon}^{\xi^0}$.

4. Example 1: Control of a linear system with uncertainties

This section presents a robust controller design for a linear process with uncertainty by applying multi-objective optimization.

4.1. Description of the problem

The model used in this example is:

$$y(s) = G(s)u(s) \tag{6}$$

$$G(s) = \frac{K_{11}e^{-\theta_{11}s}}{s(\tau_{11}s + 1)} \tag{7}$$

where y is the process output to be controlled and u is the process input (control action). The uncertainty is associated with the delay, gain, and time constant of the model. Therefore, the scenarios are defined by the vector $\boldsymbol{\xi}$ as follows:

$$\boldsymbol{\xi} = [K_{11}, \theta_{11}, \tau_{11}] \tag{8}$$

and the nominal scenario $\boldsymbol{\xi^0}$ is defined as:

$$\boldsymbol{\xi^0} = [1, 1, 1] \tag{9}$$

The control system is implemented by using a PI controller:

$$u(s) = Kc\left(e(s) + \frac{1}{Ti \cdot s}e(s)\right) \tag{10}$$

Once the control system is established, the corresponding MOP is defined (see Eq. (11)). Two design objectives are chosen. The first, f_1, evaluates the output performance by applying the IAE (integral absolute error). The second, f_2, evaluates the control effort of the system by applying the IAU (integral absolute control action). The design objectives measure IAE and IAU when a unitary disturbance is introduced at the process input. Considering IAE and IAU as independent objectives in an MOP enables the choice of a controller to be balanced in performance and control effort according to the designer's preferences a posteriori.

$$\min_{\boldsymbol{x} \in \boldsymbol{Q}} \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\xi^0}) \tag{11}$$

$$\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\xi^0}) = [f_1(\boldsymbol{x}, \boldsymbol{\xi^0}) \;\; f_2(\boldsymbol{x}, \boldsymbol{\xi^0})]$$

$$f_1(\boldsymbol{x}, \boldsymbol{\xi^0}) = \int_0^{t_f} |e| dt$$

$$f_2(\boldsymbol{x}, \boldsymbol{\xi^0}) = \int_0^{t_f} |u| dt$$

$$t_f = 1000 \text{ s}$$

subject to:

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

$$f_1(\boldsymbol{x}, \boldsymbol{\xi^0}) < 500 \;;\; f_2(\boldsymbol{x}, \boldsymbol{\xi^0}) < 100$$

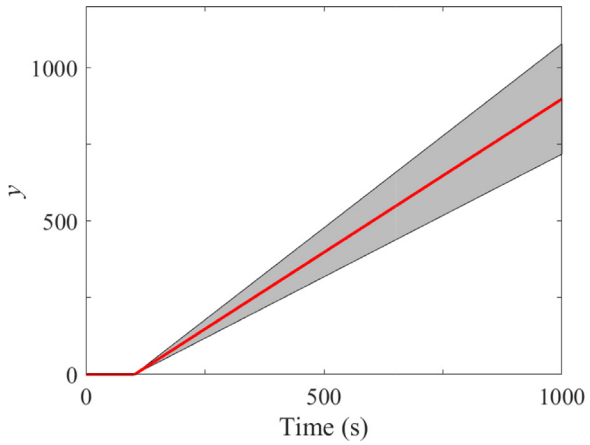

Fig. 3. Envelope response for the 8 models of $\mathcal{U}$ given a unit input $u = 1$ in $t = 100$ s. The response in the nominal model is shown in red. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

These constraints are included in the objectives to define the pertinence zone of the front. The vector of parameters is defined as:

$$\boldsymbol{x} = [Kc,\ Ti]$$

$$\overline{\boldsymbol{x}} = [1,\ 100];\quad \underline{\boldsymbol{x}} = [0.01,\ 1]$$

To take uncertainty into account, eight models are considered. Each scenario considers each extreme case, varying ±20% the nominal value of the parameters $\boldsymbol{\xi^0}$. The open-loop envelope response shown in Fig. 3 is obtained with this set of models. These eight scenarios constitute the set $\mathcal{U}$. Thus, $\boldsymbol{f_{\mathcal{U}}^{max}}(\boldsymbol{x})$ is the supremum value for every objective function separately for the set of eight models (see Eq. (4)). This value will be the function to be minimized by the $\mathcal{RU}$MOP (see Eq. (12)).

$$\min_{\boldsymbol{x}\in Q} \boldsymbol{f_{\mathcal{U}}^{max}}(\boldsymbol{x}) \tag{12}$$

subject to:

Stable in closed-loop

In addition, in (13), $\mathcal{LRU}$MOP is defined and this ensures that the controllers have an acceptable performance in the nominal model.

$$\min_{\boldsymbol{x}\in Q} \boldsymbol{f_{\mathcal{U}}^{max}}(\boldsymbol{x}) \tag{13}$$

subject to:

$$\boldsymbol{x} \in P_{\epsilon}^{\xi^0}$$

$$f_1(\boldsymbol{x}, \boldsymbol{\xi^0}) < 500\ ;\ f_2(\boldsymbol{x}, \boldsymbol{\xi^0}) < 100$$

4.2. Results and discussion

This section shows the results obtained after applying the approaches presented in Section 3. Approach 1 obtains the set $P_{\mathcal{LR}}^{\mathcal{U},*}$ and approach 2 obtains the set $P_n^{\xi^0,*}$. These sets are analyzed for the nominal scenario $\boldsymbol{\xi^0}$ and for the set of scenarios $\mathcal{U}$. The first example, since it is simpler, enables us to analyze in detail many of the sets described in Section 2 (P^{ξ^0}, $P_{\mathcal{R}}^{\mathcal{U}}$, $P_{\mathcal{LR}}^{\mathcal{U}}$ and $P_n^{\xi^0}$). The advantages and disadvantages of each regarding the robustness of their obtained controllers are shown. To make the comparison as fair as possible, the evMOGA and nevMOGA algorithms (to obtain the above-mentioned sets) are used with the same settings: $Nind_P = 400$ (initial population), $Generations = 3075$, $Nind_{GA} = 8$ (evaluations per iteration), $n_{div} = 40$ (number of divisions for each dimension in the objective space grid). The optimization stage is carried out with 25,000 evaluations of the objective function.

Fig. 4 shows the Pareto set $P^{\xi^0,*}$ obtained from solving the MOP defined in (11) for this example. In addition, the set $P_{\mathcal{R}}^{\mathcal{U},*}$ obtained from solving the $\mathcal{RU}$MOP defined in (12) is also shown. The performance of these controllers is plotted in the nominal scenario $\boldsymbol{\xi^0}$ (using $\boldsymbol{f}$) and over the set of scenarios $\mathcal{U}$ (using $\boldsymbol{f_{\mathcal{U}}^{max}}$). Note that several controllers of $P^{\xi^0,*}$ obtain a $\boldsymbol{f_{\mathcal{U}}^{max}}$ which is outside the scale of the figure. As expected, the set $P_{\mathcal{R}}^{\mathcal{U},*}$ is more robust than $P^{\xi^0,*}$ (robust is understood as the set of controllers with the best $\boldsymbol{f_{\mathcal{U}}^{max}}$). However, the set $P^{\xi^0,*}$ performs better on the nominal scenario. From the set $P^{\xi^0,*}$ the compromise controller $\boldsymbol{x^{1,\xi^0}}$ and the controller $\boldsymbol{x^{2,\xi^0}}$ (which will serve as the controller to perform the sensitivity analysis described below) are selected. From the set $P_{\mathcal{R}}^{\mathcal{U},*}$ the compromise controller $\boldsymbol{x^{\mathcal{U},\mathcal{R}}}$ is selected.

In Fig. 5, the output and control action of the system with the three selected controllers is observed in the nominal model. The three controllers obtain significantly different responses. The controller $\boldsymbol{x^{1,\xi^0}}$ obtains the smallest error.

To choose the parameter $\boldsymbol{\epsilon}$ (maximum acceptable degradation in the nominal model), it is recommended to carry out a previous analysis. To perform this analysis is necessary to choose a reference controller (controller with good performance). In this case, taking advantage of the previously selected controllers, we define $\boldsymbol{x^{1,\xi^0}}$ as the reference controller. Subsequently, we select the controller $\boldsymbol{x^{\mathcal{U},\mathcal{R}}}$ as the controller with the worst performance. The controller $\boldsymbol{x^{\mathcal{U},\mathcal{R}}}$ obtains a significantly higher error than $\boldsymbol{x^{1,\xi^0}}$. Therefore, $\boldsymbol{x^{\mathcal{U},\mathcal{R}}}$ could be considered to have unacceptable performance in the nominal model. This analysis facilitates the definition of $\boldsymbol{\epsilon}$.

To guarantee that the obtained controllers have an acceptable performance in the nominal model, the concept of light robustness can be applied. At this point, the Pareto front $P^{\xi^0,*}$ is available. Subsequently, based on the previous analysis, we define $\boldsymbol{\epsilon} = [15,\ 2]$ as the maximum acceptable degradation in the nominal model $\boldsymbol{\xi^0}$ (gray shaded area in Fig. 6). We are now in a position to deal with the $\mathcal{LRU}$MOP problem defined in (13). Fig. 6 shows the obtained set of lightly robust controllers ($P_{\mathcal{LR}}^{\mathcal{U},*}$) (approach 1 is described in Section 3.1). As can be seen, $P_{\mathcal{LR}}^{\mathcal{U},*}$, in contrast to $P_{\mathcal{R}}^{\mathcal{U},*}$, only obtains controllers in the acceptable area $P_{\epsilon}^{\xi^0,*}$ in the nominal model (gray shaded area).

To choose the parameter $\boldsymbol{n}$ (neighborhood), it is recommended to perform a sensitivity analysis on the parameters of the problem (*Kc* and *Ti*). To carry out this analysis it is necessary to choose a reference controller. Taking advantage of the previously selected controllers, we define $\boldsymbol{x^{1,\xi^0}}$ (see Fig. 4) as a reference controller. We also select a controller with a variation in *Kc* and a similar *Ti* with respect to $\boldsymbol{x^{1,\xi^0}}$. This controller is $\boldsymbol{x^{\mathcal{U},\mathcal{R}}}$ (see Fig. 4, controller with a variation of 0.1 in *Kc* with respect to $\boldsymbol{x^{1,\xi^0}}$). These two controllers obtain significantly different responses (see Fig. 5). We also select a controller with a similar *Kc* and a variation in T_i with respect to the reference controller. This controller is $\boldsymbol{x^{2,\xi^0}}$ (see Fig. 4, controller with a variation of 3 in *Ti* with respect to $\boldsymbol{x^{1,\xi^0}}$). Again, $\boldsymbol{x^{1,\xi^0}}$ and $\boldsymbol{x^{2,\xi^0}}$ obtain significantly different responses (see Fig. 5). Therefore, a variation of 0.1 in the *Kc* (variation between $\boldsymbol{x^{1,\xi^0}}$ and $\boldsymbol{x^{\mathcal{U},\mathcal{R}}}$) and 3 in the *Ti* (variation between $\boldsymbol{x^{1,\xi^0}}$ and $\boldsymbol{x^{2,\xi^0}}$) produces significant changes in the output of the process. This simple sensitivity analysis will facilitate the definition of neighborhood $\boldsymbol{n}$.

Finally, we obtain the relevant nearly optimal controllers under the nominal scenario and uncertainty is analyzed (approach

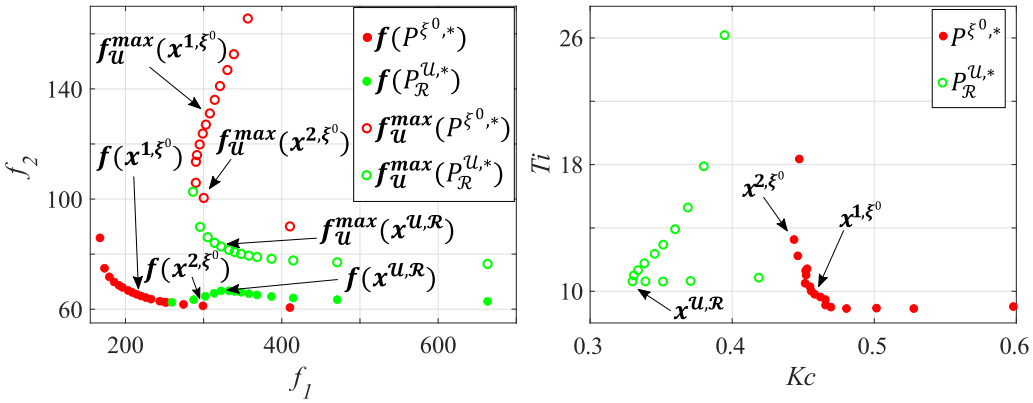

Fig. 4. The set $P^{\xi^0,*}$ is shown in red and the set $P_{\mathcal{R}}^{\mathcal{U},*}$ is shown in green. The decision space is shown on the right. The objective space is shown on the left. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

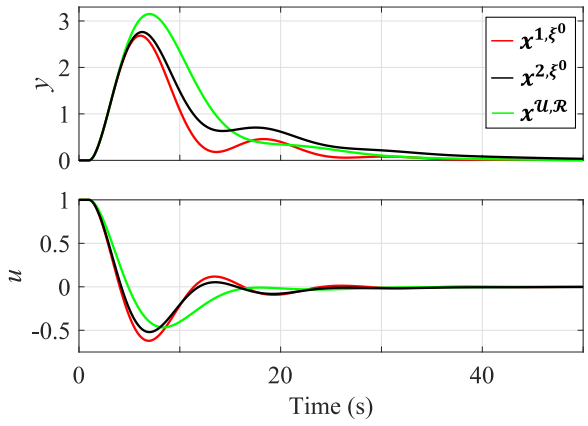

Fig. 5. The response of the controller x^{1,ξ^0} is shown in red and the response of controller x^{2,ξ^0} is shown in black. The response of the controller $x^{\mathcal{U},\mathcal{R}}$ is shown in green. The system output y is at the top. The control action u is below. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

2 described in Section 3.2). First, we define $\epsilon = [15,\ 2]$ as in the previous approach. Subsequently, based on the previous sensitivity analysis on the parameters *Kc* and *Ti*, the neighborhood $n = [0.03,\ 1]$ (a variation that produces significant changes, see Definition 5) is defined. The definition of this parameter conditions the results obtained for the designer's criteria.

In Fig. 7 we observe the set $P_n^{\xi^0,*}$ (approach 2) obtained by solving the MOP defined in (11). Several of the obtained controllers have a $f_{\mathcal{U}}^{max}$ that is outside the scale of the figure, and these are the least robust controllers. The same figure also depicts the previously obtained lightly robust controllers $P_{\mathcal{LR}}^{\mathcal{U},*}$ (approach 1). We can see that $P_n^{\xi^0,*}$ obtains, among others, controllers with robustness to uncertainty similar to that obtained by $P_{\mathcal{LR}}^{\mathcal{U},*}$.

We next focus on the most robust solution set of $P_n^{\xi^0,*}$. This set is defined as:

$$P_n^{\mathcal{U},*} := \{x \in P_n^{\xi^0,*} |\ \nexists x^o \in P_n^{\xi^0,*} : x^o \preceq x\} \tag{14}$$

where dominance is assessed with the function $f_{\mathcal{U}}^{max}(x)$.

In Fig. 8 the sets $P_n^{\mathcal{U},*}$ and $P_{\mathcal{LR}}^{\mathcal{U},*}$ are shown. Additionally, the neighborhood of each controller that constitutes $P_{\mathcal{LR}}^{\mathcal{U},*}$ is shown in yellow. As can be seen, both sets have similar robustness. This is because the controllers which compose $P_{\mathcal{LR}}^{\mathcal{U},*}$ always have a neighboring controller in the set $P_n^{\xi^0,*}$ (see yellow area).

If the designer chooses a smaller neighborhood n, then $P_n^{\xi^0,*}$ and $P_{\mathcal{LR}}^{\mathcal{U},*}$ are expected to be more similar in exchange for a larger number of solutions and a higher computational cost. If the designer chooses a larger neighborhood, predictably $P_n^{\xi^0,*}$ and $P_{\mathcal{LR}}^{\mathcal{U},*}$ will be less similar, in exchange for a smaller number of solutions and lower computational cost. Therefore, in the second approach, it is the designer who has the task of deciding how similar will be the obtained set $P_n^{\xi^0,*}$ to $P_{\mathcal{LR}}^{\mathcal{U},*}$ by defining the neighborhood n.

From the set $P_n^{\xi^0,*}$, the controller $x^{\xi^0,n}$ is selected, since it has a compromise $f_{\mathcal{U}}^{max}$. From the set $P_{\mathcal{LR}}^{\mathcal{U},*}$ we select the controller $x^{\mathcal{U},\mathcal{LR}}$, neighbor of $x^{\xi^0,n}$. $x^{\mathcal{U},\mathcal{LR}}$ is a bit more robust than

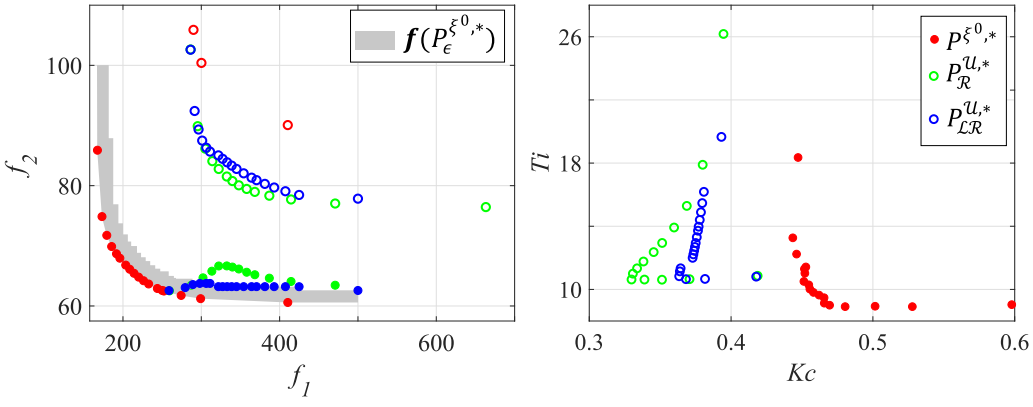

Fig. 6. The decision space is shown on the right and the objective space on the left. f is represented by dots, $f_{\mathcal{U}}^{max}$ by circles.

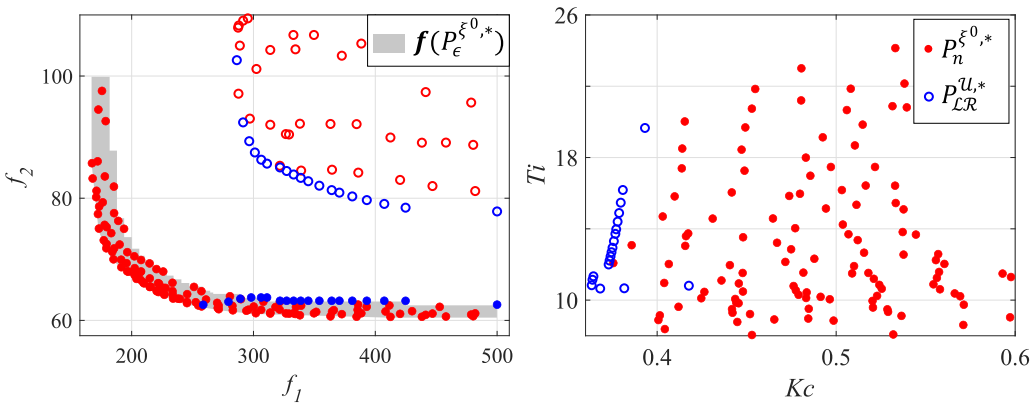

Fig. 7. The decision space is shown on the right. The objective space is shown on the left. $\boldsymbol{f}$ is represented by dots and $\boldsymbol{f}_{\mathcal{U}}^{max}$ is represented by circles.

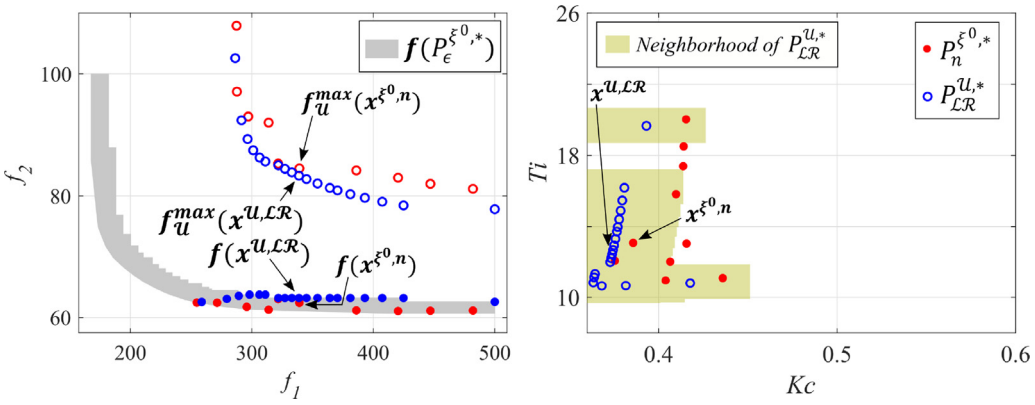

Fig. 8. On the right is the decision space. On the left is the objective space. $\boldsymbol{f}$ is represented by dots, $\boldsymbol{f}_{\mathcal{U}}^{max}$ is represented by circles.

$\boldsymbol{x}^{\boldsymbol{\xi^0},\boldsymbol{n}}$. However, $\boldsymbol{x}^{\boldsymbol{\xi^0},\boldsymbol{n}}$ performs better than $\boldsymbol{x}^{\boldsymbol{\mathcal{U}},\boldsymbol{\mathcal{LR}}}$ in the nominal scenario.

To validate the controllers $\boldsymbol{x}^{\boldsymbol{\xi^0},\boldsymbol{n}}$ and $\boldsymbol{x}^{\boldsymbol{\mathcal{U}},\boldsymbol{\mathcal{LR}}}$, Fig. 9 shows the envelopes of the output and control action (y, u) in a closed-loop for the eight models analyzed. As can be seen, both controllers obtain similar envelopes for both output and control actions. Therefore, both controllers have similar robustness based on the most essential performance criteria. Both controllers obtain an envelope with a peak amplitude between approximately 2.5 and 3.5. In addition, they reject the disturbance in a very similar time. Thus, it is confirmed that the defined neighborhood is correct (two neighboring solutions produce similar responses).

Table 1 shows the computational cost of obtaining each set[6] (before optimization, during the optimization stage, and in the decision stage). The set $P^{\xi^0,*}$ has the lowest computational cost. However, this set does not guarantee robust solutions (or similar). The sets $P_{\mathcal{R}}^{\mathcal{U},*}$ and $P_{\mathcal{LR}}^{\mathcal{U},*}$ have the highest computational cost because they evaluate all scenarios in the optimization process. These sets guarantee the most robust solutions (with any performance in the nominal scenario). The set $P_n^{\xi^0,*}$ has a small computational cost (slightly greater than $P^{\xi^0,*}$). In addition, as previously observed (and proved in Theorem 1), $P_n^{\xi^0,*}$ obtains controllers that are similar/neighboring to those obtained by $P_{\mathcal{LR}}^{\mathcal{U},*}$ (see Fig. 8).

Some final conclusions are as follows:

1. Set $P^{\xi^0,*}$ obtains optimal controllers in the nominal scenario $\boldsymbol{\xi^0}$, but they are not very robust (see Fig. 4).

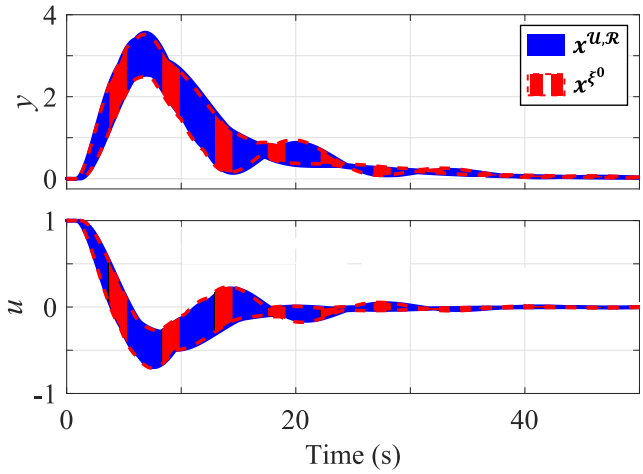

Fig. 9. Envelopes obtained with the set of scenarios $\boldsymbol{\xi} \in \mathcal{U}$ for the controllers $\boldsymbol{x}^{\boldsymbol{\xi^0},\boldsymbol{n}}$ and $\boldsymbol{x}^{\boldsymbol{\mathcal{U}},\boldsymbol{\mathcal{R}}}$. Above is the output of the system y. Below the control action u.

2. Set $P_{\mathcal{R}}^{\mathcal{U},*}$ obtains the most robust controllers, but may result in undesirable performance over $\boldsymbol{\xi^0}$ (see Fig. 6). Moreover, obtaining $P_{\mathcal{R}}^{\mathcal{U},*}$ has a high computational cost (see Table 1).
3. Set $P_{\mathcal{LR}}^{\mathcal{U},*}$ (approach 1) obtains the most robust controllers with acceptable performance over $\boldsymbol{\xi^0}$ (see Fig. 6). However, obtaining $P_{\mathcal{LR}}^{\mathcal{U},*}$ has a high computational cost (see Table 1).
4. $P_n^{\xi^0,*}$ (approach 2) obtains controllers with similar robustness to $P_{\mathcal{LR}}^{\mathcal{U},*}$ (see Fig. 8) and with acceptable performance

[6] CPU processor Intel Core i7, 3.2 GHz with 16 GB RAM.

Table 1
Computational cost of example 1.

Set	Computational cost			
	Before optim.	Optim. stage	Decision stage	Total cost
$P^{\xi^0,*}$	–	3836 s	24 s	3860 s
$P^{\mathcal{U},*}_{\mathcal{R}}$	–	31593 s	2 s	31595 s
$P^{\mathcal{U},*}_{\mathcal{LR}}$	3836 s	25922 s	–	29758 s
$P^{\xi^0,*}_n$	–	4054 s	170 s	4224 s

over $\boldsymbol{\xi^0}$. However, approach 2 has a significantly lower computational cost (see Table 1).

5. Example 2: Control of the boiler–turbine system

This section presents a robust controller design for a non-linear multivariable process with uncertainty and applying multi-objective optimization.

5.1. Description of the problem

The process to be controlled consists of a non-linear MIMO (3x3) boiler–turbine unit for electric power generation. The model was proposed in [24] and corresponds to boiler–turbine plant P16/G16 at the Sydsvenska Kraft AB plant in Malmö, Sweden. This system presents significant non-linearities, and strong coupling between its input–output variables [25,26]. The first-principles model of the system is shown in Eqs. (15)–(19).

$$\dot{x}_1 = \xi_1\, u_2 x_1^{9/8} + \xi_2\, u_1 + \xi_3\, u_3 \tag{15}$$

$$\dot{x}_2 = (\xi_4\, u_2 + \xi_5)\, x_1^{9/8} + \xi_6\, x_2 \tag{16}$$

$$\dot{x}_3 = (\xi_7\, u_3 - (\xi_8\, u_2 - \xi_9)\, x_1) \tag{17}$$

$$y_1 = x_1;\ \ y_2 = x_2 \tag{18}$$

$$y_3 = 0.05\,(0.13073\, x_3 + \xi_{10}\, a_{cs} + \frac{q_e}{9} - 67.975) \tag{19}$$

x_1 (boiler pressure [Kg/cm^2]), x_2 (electrical power output [MW]), and x_3 (fluid density [Kg/m^3]) are the state variables of the system. Inputs u_1, u_2, and u_3 are the valve position for fuel flow, steam control, and feedwater flow control, respectively. The outputs are y_1 (boiler pressure [Kg/cm^2]), y_2 (electrical power output [MW]), and y_3 (water level in the boiler manifold with respect to the reference level for the nominal operating point [m]). The output y_3 can be either positive or negative and depends on the steam quality a_{cs} and the evaporation rate q_e:

$$a_{cs} = \frac{(1 - 0.001538\, x_3)\,(0.8\, x_1 - 25.6)}{x_3\,(1.0394 - 0.0012304\, x_1)} \tag{20}$$

$$q_e = (0.854\, u_2 - 0.147)\, x_1 + 45.59\, u_1 - 2.514\, u_3 - 2.096 \tag{21}$$

The actuators in the system have physical limitations and certain restrictions defined below:

$$\begin{aligned} 0 \le & u_i \le 1;\ i = 1, 2, 3 \\ -0.007 \le & \dot{u}_1 \le 0.007 \\ -2 \le & \dot{u}_2 \le 0.02 \\ -0.05 \le & \dot{u}_3 \le 0.05 \end{aligned} \tag{22}$$

The uncertainty ($\boldsymbol{\xi}$) is associated with the coefficients of the model Eqs. (15), (16) and (17), as well as the parameter that multiplies the term a_{cs} of Eq. (19) (as proposed in [27]). Therefore, the scenarios are defined by the vector $\boldsymbol{\xi}$ as follows:

$$\boldsymbol{\xi} = [\xi_1, \xi_2, \xi_3, \xi_4, \xi_5, \xi_6, \xi_7, \xi_8, \xi_9, \xi_{10}] \tag{23}$$

and the nominal scenario $\boldsymbol{\xi^0}$ is defined as:

$$\begin{aligned} \boldsymbol{\xi^0} = [&-0.0018, 0.9, -0.15, 0.073, -0.016, \\ &-0.1, 141/85, 1.1/85, 0.19/85, 100] \end{aligned} \tag{24}$$

Table 2
Operating point p_4.

Operating point	x_1^0	x_2^0	x_3^0	u_1^0	u_2^0	u_3^0	y_3^0
p_4	108	66.65	428	0.34	0.69	0.433	0

In this example, we use a typical operating point p_4 defined in [24] and shown in Table 2. A multi-loop control structure with controllers 1DOF-PI and antiwindup is proposed. The pairing: $y_1 - u_1$, $y_2 - u_2$ and $y_3 - u_3$ will be used. Therefore:

$$\begin{aligned} u_1(s) =& Kc_1 \left(e_1(s) + \tfrac{1}{Ti_1 \cdot s} e_1(s)\right) \\ u_2(s) =& Kc_2 \left(e_2(s) + \tfrac{1}{Ti_2 \cdot s} e_2(s)\right) \\ u_3(s) =& Kc_3 \left(e_3(s) + \tfrac{1}{Ti_3 \cdot s} e_3(s)\right) \end{aligned} \tag{25}$$

where Kc_i are the proportional gains and Ti_i are the integral times in seconds for each loop i. $e_1 = r_1 - y_1$, $e_2 = r_2 - y_2$ and $e_3 = r_3 - y_3$ are the output errors, being r_1, r_2, and r_3 the set-points for each closed-loop.

The MOP is defined in Eq. (26).

$$\min_{\boldsymbol{x} \in Q} \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\xi^0}) \tag{26}$$

$$\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\xi^0}) = [f_1(\boldsymbol{x}, \boldsymbol{\xi^0})\, f_2(\boldsymbol{x}, \boldsymbol{\xi^0})]$$

$$f_1(\boldsymbol{x}, \boldsymbol{\xi^0}) = \int_0^{t_f} \underbrace{\frac{|e_1|}{64.8} + \frac{|e_2|}{113.13} + \frac{|e_3|}{1.95}}_{r_1 = y_1^0 + \Delta r_1,\ r_2 = y_2^0,\ r_3 = y_3^0} dt + \int_0^{t_f} \underbrace{\frac{|e_1|}{64.8} + \frac{|e_2|}{113.13} + \frac{|e_3|}{1.95}}_{r_1 = y_1^0,\ r_2 = y_2^0 + \Delta r_2,\ r_3 = y_3^0} dt$$

$$f_2(\boldsymbol{x}, \boldsymbol{\xi^0}) = \int_0^{t_f} \underbrace{\left|\frac{du_1}{dt}\right| + \left|\frac{du_2}{dt}\right| + \left|\frac{du_3}{dt}\right|}_{r_1 = y_1^0 + \Delta r_1,\ r_2 = y_2^0,\ r_3 = y_3^0} dt + \int_0^{t_f} \underbrace{\left|\frac{du_1}{dt}\right| + \left|\frac{du_2}{dt}\right| + \left|\frac{du_3}{dt}\right|}_{r_1 = y_1^0,\ r_2 = y_2^0 + \Delta r_2,\ r_3 = y_3^0} dt$$

$t_f = 2500$ s

subject to:

$$\underline{\boldsymbol{x}} \le \boldsymbol{x} \le \overline{\boldsymbol{x}}$$

$$f_1(\boldsymbol{x}, \boldsymbol{\xi^0}) < 5000\ ;\ f_2(\boldsymbol{x}, \boldsymbol{\xi^0}) < 50$$

The constraints on f_1 and f_2 are used to define the pertinence zone of the front. The parameter vector is defined as:

$$\boldsymbol{x} = [Kc_1,\ Ti_1,\ Kc_2,\ Ti_2,\ Kc_3,\ Ti_3]$$

$$\overline{\boldsymbol{x}} = [10,\ 150,\ 2.5,\ 150,\ 15,\ 150];$$
$$\underline{\boldsymbol{x}} = [0.001,\ 1,\ 0.001,\ 1,\ 0.001,\ 1]$$

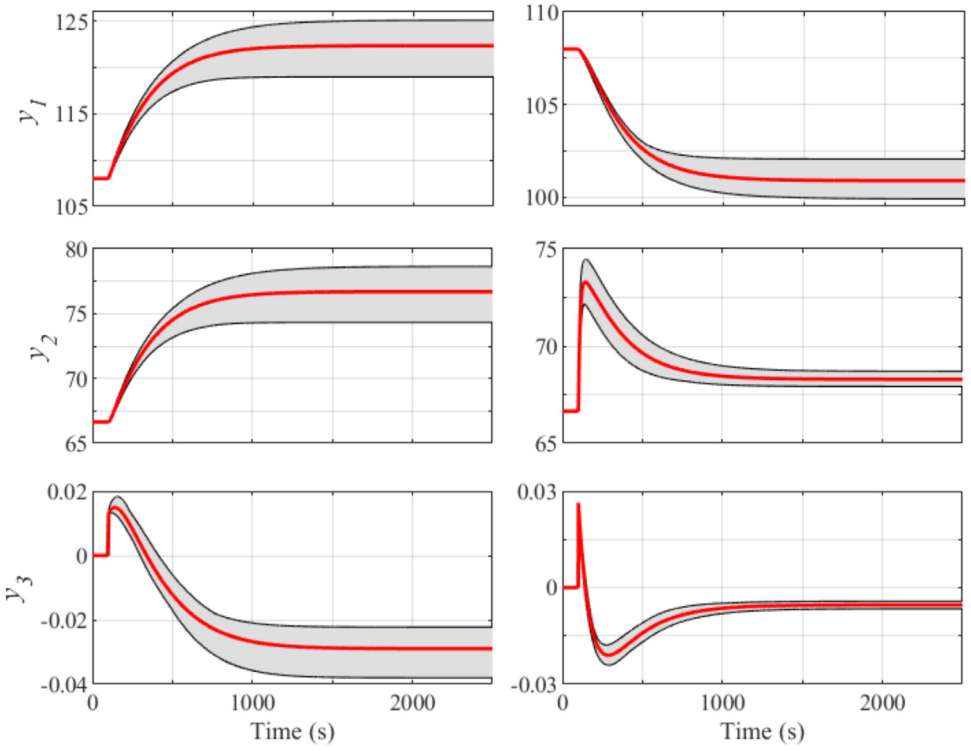

Fig. 10. Envelope response to a model parameter variation of ±10%. On the left with a input variation of $\Delta u = [0.05\ 0\ 0]$ at the operating point p_4. On the right with an input variation of $\Delta u = [0\ 0.05\ 0]$. The nominal model response is shown in red. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

The first objective design, f_1, evaluates the performance of each output by applying the integral absolute error (IAE). These errors have been normalized so that they have the same relative weight. For this, each error is divided by the range of each output, determined from the operating points p_1 to p_7 described in [24]. In this way, we avoid, for example, that output y_3 has a smaller weight on f_1 (because the range of values of this output is much smaller than y_1 and y_2).

The second objective, f_2, measures the control effort of the system by applying the IADU (integral of the absolute value of the derivative control signal). The design objectives measure the IAE and the IADU when a step is introduced in the reference r_1 (boiler pressure) going from $y_1^0 = 108$ Kg/cm^2 to 120 Kg/cm^2 ($\Delta r_1 = 12$ Kg/cm^2) and, in the reference r_2 (the electrical output power) going from $y_2^0 = 66.65$ MW to 86.65 MW ($\Delta r_2 = 20$ MW). Again, considering IAE and IADU as separate objectives allows the designer to choose a controller with a trade-off in performance and control effort.

To analyze the uncertainty, 50 models are considered in which the parameters are randomly varied ±10% concerning the nominal scenario $\boldsymbol{\xi^0}$. In this way, the set of scenarios $\mathcal{U}$ is generated. With this set of models, we obtain the open-loop envelope response seen in Fig. 10. Since the output y_3 is unstable in open loop, the envelope on this output has been obtained by a proportional controller with gain 2. Thus, $\boldsymbol{f_{\mathcal{U}}^{max}}(\boldsymbol{x})$ is the supremum value for every objective function separately for the set of 50 scenarios (see Eq. (4)). In addition, the obtained controllers should have an acceptable performance in the nominal model ($\boldsymbol{x} \in P_\epsilon^{\xi^0}$), i.e. they should be lightly robust (see $\mathcal{LRU}$MOP defined in (27)).

$$\min_{\boldsymbol{x}\in Q} \boldsymbol{f_{\mathcal{U}}^{max}}(\boldsymbol{x}) \tag{27}$$

subject to:

$$\boldsymbol{x} \in P_\epsilon^{\xi^0}$$

$$f_1(\boldsymbol{x}, \boldsymbol{\xi^0}) < 5000\ ;\ f_2(\boldsymbol{x}, \boldsymbol{\xi^0}) < 50$$

5.2. Results and discussion

This section shows the results obtained after applying the approaches presented in Section 3. To make the comparison as fair as possible, both approaches use the algorithms with the same settings: $Nind_P = 200$ (initial population), $Generations = 1000$, $Nind_{GA} = 8$ (evaluations per iteration), $n_{div} = 20$ (number of divisions for each dimension in the objective space grid). The optimization is carried out with 10,000 evaluations of the objective function. No further evaluations have been performed due to the high computational cost of taking uncertainty into account in the optimization stage.

For both examples, we define $\boldsymbol{\epsilon} = [25,\ 0.2]$ as the maximum acceptable degradation in the nominal scenario. Under the first approach, the set $P_{\mathcal{LRU}}^{\mathcal{U},*}$ ($\mathcal{LRU}$MOP defined in (27)) is obtained. For approach 2, as in example 1, a sensitivity analysis is performed on the parameters Kc_i and Ti_i. Thus, we define the neighborhood $\boldsymbol{n} = [0.5,\ 50,\ 0.5,\ 50,\ 3,\ 50]$ (see Definition 5), and obtain the set $P_n^{\xi^0,*}$ (MOP defined in (26)). Fig. 11 shows the obtained sets. The decision space is shown on the right. To show the decision variables, we use the level diagram (LD[7] [28,29]) visualization tool and using 2-norm ($\|\cdot\|_2$). The LD tool is based on level diagrams, where each design objective and parameter is represented on a separate diagram. In this way, each level diagram represents the objective value or parameter on the abscissa axis and its norm value on the ordinate axis. The objective space is shown on the left. These sets are analyzed under the nominal scenario using

[7] Available in Matlab Central: https://es.mathworks.com/matlabcentral/fileexchange/62224-interactive-tool-for-decision-making-in-multiobjective-optimization-with-level-diagrams.

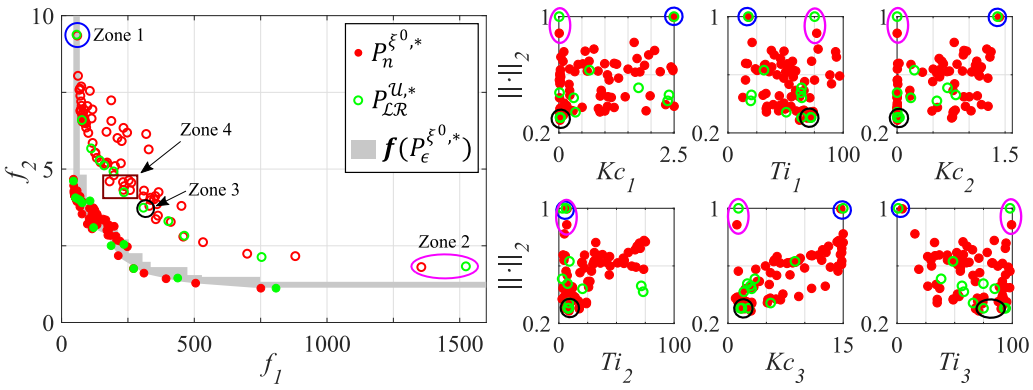

Fig. 11. The decision space by level diagram is shown on the right. The objective space is shown on the left. $\boldsymbol{f}$ is represented by dots, $\boldsymbol{f}_{\mathcal{U}}^{max}$ is represented by circles.

Table 3
Controllers in Zone 4 of Fig. 11.

	Set	Kc_1	Ti_1	Kc_2	Ti_2	Kc_3	Ti_3	$\boldsymbol{f}$	$\boldsymbol{f}_{\mathcal{U}}^{max}$
$\boldsymbol{x}^{1,\mathcal{U}}$	$P_{LR}^{U,*}$	0.36	50.5	0.002	8.5	5.6	66.0	[120.5 3.1]	[234.8 4.2]
$\boldsymbol{x}^{2,\xi^0,n}$	$P_n^{\xi^0,*}$	0.50	47.7	0.003	6.9	5.5	69.6	[97.2 3.2]	[180.9 4.6]
$\boldsymbol{x}^{3,\xi^0,n}$	$P_n^{\xi^0,*}$	0.17	44.9	0.001	4.8	9.2	53.8	[133.9 3.2]	[262.7 4.6]
$\boldsymbol{x}^{4,\xi^0,n}$	$P_n^{\xi^0,*}$	2.32	39.3	1.171	12.3	1.8	91.5	[125.3 3.7]	[253.3 4.6]
$\boldsymbol{x}^{5,\xi^0,n}$	$P_n^{\xi^0,*}$	0.31	64.2	0.004	6.4	2.5	74.4	[115.0 3.1]	[233.0 4.3]
$\boldsymbol{x}^{6,\xi^0,n}$	$P_n^{\xi^0,*}$	1.05	54.7	0.950	14.1	1.8	75.3	[120.3 3.7]	[231.5 4.6]
$\boldsymbol{x}^{7,\xi^0,n}$	$P_n^{\xi^0,*}$	0.86	33.2	0.001	4.0	5.2	34.9	[121.6 3.2]	[236.9 4.4]
$\boldsymbol{x}^{8,\xi^0,n}$	$P_n^{\xi^0,*}$	0.25	65.9	0.807	16.9	1.6	69.9	[133.6 3.5]	[257.8 4.3]

$\boldsymbol{f}$ or under uncertainty using $\boldsymbol{f}_{\mathcal{U}}^{max}$ (several controllers of $P_n^{\xi^0,*}$ obtain a $\boldsymbol{f}_{\mathcal{U}}^{max}$ which is outside the scale of the figure).

As can be seen, in the objective space of Fig. 11, the set $\boldsymbol{f}_{\mathcal{U}}^{max}(P_n^{\xi^0,*})$ adequately characterizes the set $\boldsymbol{f}_{\mathcal{U}}^{max}(P_{\mathcal{LR}}^{\mathcal{U},*})$. This figure shows the zones 1 (blue), 2 (purple), and 3 (black). There are two neighboring controllers in these zones, one obtained by approach 1, and another obtained by approach 2. The same could be extrapolated to all the controllers in $P_{\mathcal{LR}}^{\mathcal{U},*}$, since each of them has at least one similar (neighboring) controller in $P_n^{\xi^0,*}$. Therefore, it can be concluded that approach 2 has found neighboring controllers (neighborhood $\boldsymbol{n}$ defined by the designer) to controllers of approach 1.

Furthermore, in this example, there is multimodality due to aggregation in the design objectives (see Eq. (26)). Therefore, not only the most robust controllers are helpful. Some significantly different controllers perform similarly in the objective space due to aggregation [20]. An example of this is seen in the selected Zone 4 in Fig. 11 (brown square). The parameters and performance of the controllers in this zone are shown in Table 3. The controller $\boldsymbol{x}^{1,\mathcal{U}}$, is the only controller obtained by $P_{\mathcal{LR}}^{\mathcal{U},*}$ in this area. However, $P_n^{\xi^0,*}$ is able to obtain seven different controllers with similar robustness to $\boldsymbol{x}^{1,\mathcal{U}}$ ($\boldsymbol{x}^{2,\xi^0,n}$ to $\boldsymbol{x}^{8,\xi^0,n}$). From these seven controllers, $\boldsymbol{x}^{2,\xi^0,n}$ is a neighboring controller to $\boldsymbol{x}^{1,\mathcal{U}}$ (neighborhood $\boldsymbol{n} = [0.5,\ 50,\ 0.5,\ 50,\ 3,\ 50]$). However, all the other controllers, in particular $\boldsymbol{x}^{3,\xi^0,n}$ and $\boldsymbol{x}^{4,\xi^0,n}$, are controllers significantly different to $\boldsymbol{x}^{1,\mathcal{U}}$ and $\boldsymbol{x}^{2,\xi^0,n}$ (non-neighbors). This diversity makes approach 2 even more interesting.

To validate these controllers, in Fig. 12 we observe the envelope obtained for the 50 scenarios $\boldsymbol{\xi} \in \mathcal{U}$ when a step is introduced in r_1. The controller $\boldsymbol{x}^{3,\xi^0,n}$ obtains an envelope with higher overshoot and settling time on y_1 and y_2. The controller $\boldsymbol{x}^{4,\xi^0,n}$ has significantly lower overshoot on y_2. The controller $\boldsymbol{x}^{4,\xi^0,n}$ has a significantly larger settling time than the other controllers on y_3. The controller $\boldsymbol{x}^{2,\xi^0,n}$ obtains an envelope similar to $\boldsymbol{x}^{1,\mathcal{U}}$.

Fig. 13 shows this envelope when the step is in r_2. The controller $\boldsymbol{x}^{3,\xi^0,n}$ obtains an envelope with a greater overshoot on y_1. The controller $\boldsymbol{x}^{4,\xi^0,n}$ has significantly smaller overshoot on y_1 and y_2. The controller $\boldsymbol{x}^{4,\xi^0,n}$ has a significantly higher oscillation and settling time than the other controllers on y_3. The controller $\boldsymbol{x}^{2,\xi^0,n}$ obtains an envelope similar to $\boldsymbol{x}^{1,\mathcal{U}}$.

As can be seen, controllers $\boldsymbol{x}^{1,\mathcal{U}}$ and $\boldsymbol{x}^{2,\xi^0,n}$ obtain a similar envelope. However, the controllers $\boldsymbol{x}^{3,\xi^0,n}$ and $\boldsymbol{x}^{4,\xi^0,n}$ obtain significantly different envelopes. $\boldsymbol{x}^{4,\xi^0,n}$ performs better on y_1 and y_2 than its competitors but worse on y_3.

This results confirms the good choice of the neighborhood $\boldsymbol{n}$. The neighboring controllers $\boldsymbol{x}^{1,\mathcal{U}}$ and $\boldsymbol{x}^{2,\xi^0,n}$ obtain similar responses. The non-neighboring controllers ($\boldsymbol{x}^{1,\mathcal{U}}$ and $\boldsymbol{x}^{3,\xi^0,n}$, $\boldsymbol{x}^{1,\mathcal{U}}$ and $\boldsymbol{x}^{4,\xi^0,n}$, etc.) obtain significantly different responses. $\boldsymbol{x}^{3,\xi^0,n}$ and $\boldsymbol{x}^{4,\xi^0,n}$ are characterized by $P_n^{\xi^0,*}$ due to the fact that despite not being optimal under the nominal scenario (only $\boldsymbol{x}^{2,\xi^0,n}$ is optimal), they are nearly optimal and nondominated in their neighborhood. These controllers are omitted by $P_{\mathcal{LR}}^{\mathcal{U},*}$ (because they obtain worse robustness than $\boldsymbol{x}^{1,\mathcal{U}}$). However, these controllers are also relevant and provide interesting alternatives to the designer before making the final decision. The same phenomenon occurs at other points on the Pareto front.

Table 4 shows the computational cost[8] required by each approach. The set $P_{\mathcal{LR}}^{\mathcal{U},*}$ obtains the most robust controllers (see Fig. 11). However, obtaining this set has a high computational cost as it must evaluate the controllers over the 50 scenarios $\mathcal{U}$ in the optimization process. Moreover, before the optimization,

[8] CPU processor Intel Core i7, 3.2 GHz with 16 GB RAM.

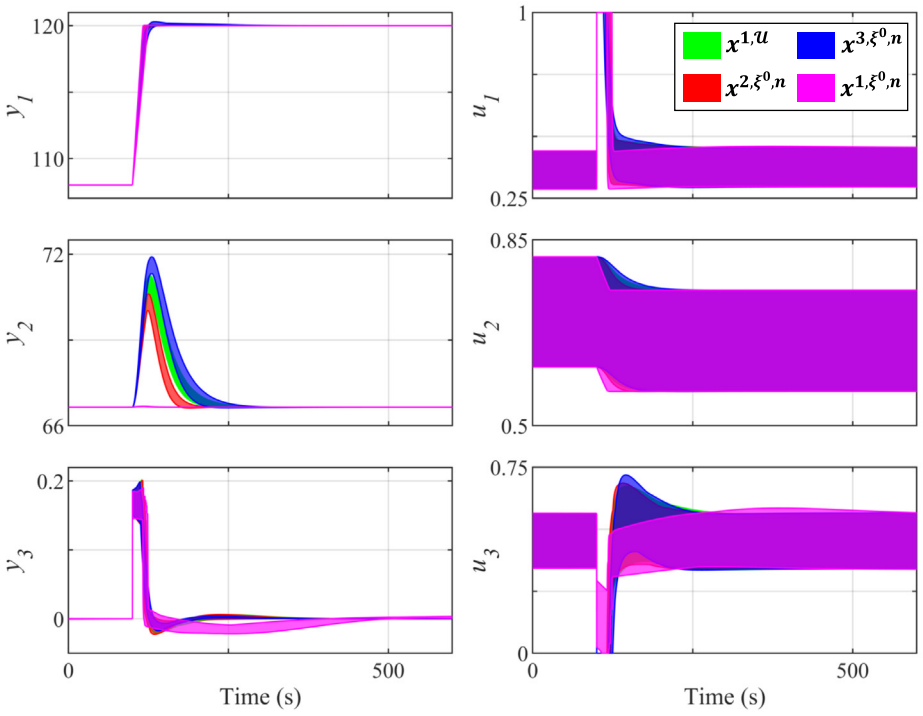

Fig. 12. Envelope response of controllers $x^{1,\mathcal{U}}$, $x^{2,\xi^0,n}$, $x^{3,\xi^0,n}$ and $x^{4,\xi^0,n}$ for the 50 scenarios $\xi \in \mathcal{U}$ when the step is set in r_1.

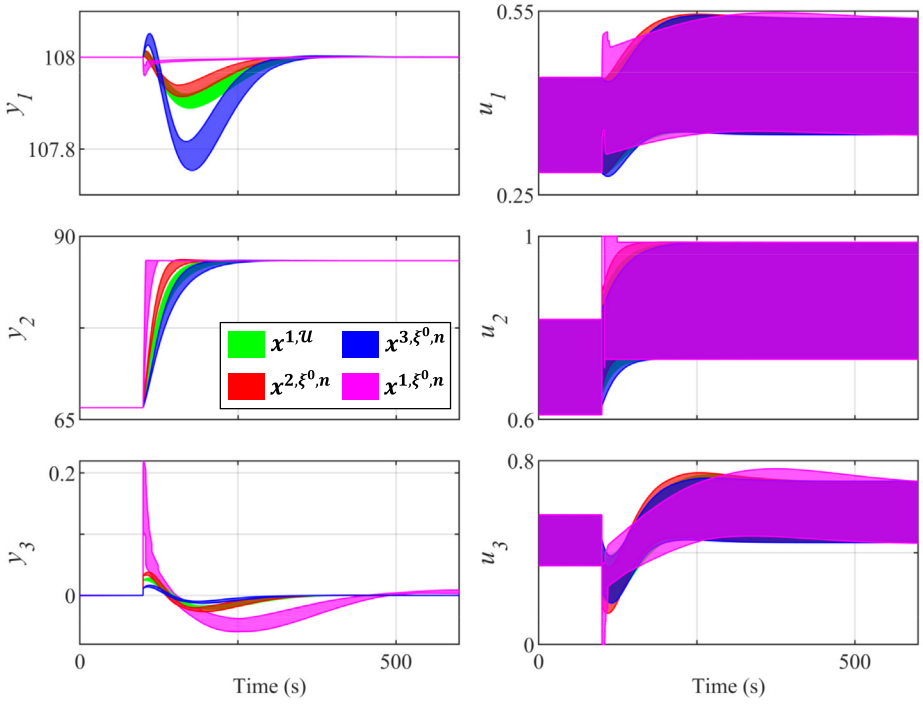

Fig. 13. Envelope response of controllers $x^{1,\mathcal{U}}$, $x^{2,\xi^0,n}$, $x^{3,\xi^0,n}$ and $x^{4,\xi^0,n}$ for the 50 scenarios $\xi \in \mathcal{U}$ when the step is set in r_2.

Table 4
Computational cost of example 2.

Set	Computational cost			
	Before optim.	Optim. stage	Decision stage	Total cost
$P_{\mathcal{LR}}^{\mathcal{U},*}$	428 min.	17333 min.	–	17761 min.
$P_n^{\xi^0,*}$	–	454 min.	167 min.	621 min.

the computation of $P^{\xi^0,*}$ is required. The set $P_n^{\xi^0,*}$ has a similar robustness to $P_{\mathcal{LR}}^{\mathcal{U},*}$ (see Fig. 11) with a significantly lower computational cost (28 times less). Moreover, this set can obtain significantly different controllers with similar robustness to that obtained by $P_{\mathcal{LR}}^{\mathcal{U},*}$. Therefore, approach 2: (1) obtains controllers with similar robustness to approach 1 and with a significantly lower computational cost; and (2) provides a greater diversity of solutions to the designer. This greater diversity can modify the final decision of the designer.

6. Conclusions

In this work, we have shown a computationally efficient methodology to characterize lightly robust controllers for linear and nonlinear systems. This has been done under a multiobjective optimal design approach and considering stochastic parametric uncertainty. It has been shown mathematically that the proposed approach characterizes lightly robust solutions. The proposed methodology addresses the uncertainty by analyzing the robustness of optimal and nearly optimal controllers in the nominal scenario.

Two examples of robust controllers design have been used to compare the traditional approach and the one proposed in this paper, the first example in a linear single-input single-output system and the second example for a multiple-inputs multiple-outputs system.

The first example has only one proportional integral controller to tune and eight scenarios, and it has been used to clearly compare the robust controllers obtained with both approaches. The choice of the maximum allowable degradation ($\boldsymbol{\epsilon}$) and the neighborhood ($\boldsymbol{n}$) are parameters that condition the lightly robust controllers and their characterization, respectively. Both parameters make physical sense, and the designer's preferences will be used to set them. In this example, with the approach proposed in this paper, the computational cost is seven times lower than with the traditional approach. In addition, it has been shown that all lightly robust controllers found with the traditional approach have been correctly characterized with the approach proposed in this paper as neighbor/similar controllers have been found for each of them.

The second example produces similar conclusions. In this case, 50 scenarios have been considered with a model with ten uncertainty parameters. The greater the number of scenarios, the more interesting the computational cost reduction offered by the proposed approach; in this example, it was 39 times lower. There is also another advantage of the proposed approach. A multimodality issue arises when the design objectives to be optimized are formed based on the aggregation of partial objectives. Multimodality causes the apparition of significantly different controllers that obtain similar performance in the space of the optimized objectives. These controllers are also relevant and provide interesting alternatives to the designer as they offer different values on the partial objectives. These controllers are also characterized in the approach presented in this paper, and this cannot be done with the traditional approach.

Declaration of competing interest

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

Acknowledgments

This work was supported in part by the grant PID2021-124908NB-I00 founded by MCIN/AEI/ 10.13039/501100011033/ and by "ERDF A way of making Europe"; by the Universitat Politècnica de València through the grant SP20200109 (PAID-10-20); by grant PRE2019-087579 funded by MCIN/AEI/ 10.13039/501100011033 and by "ESF Investing in your future"; and by the Generalitat Valenciana regional government through project CIAICO/2021/064. Funding for open access charge: CRUE-Universitat Politècnica de València.

References

[1] Reynoso-Meza G, Blasco X, Sanchis J, Herrero JM. Controller tuning with evolutionary multiobjective optimization: A holistic multiobjective optimization design procedure. Springer; 2017.
[2] Deb K. Multi-objective optimisation using evolutionary algorithms: An introduction. In: Multi-objective evolutionary optimisation for product design and manufacturing. Springer; 2011, p. 3–34.
[3] Miettinen K. Nonlinear multiobjective optimization. Vol. 12. Springer Science & Business Media; 2012.
[4] Gunantara N. A review of multi-objective optimization: Methods and its applications. Cogent Eng 2018;5:1502242.
[5] de la Penad DM, Bemporad A, Alamo T. Stochastic programming applied to model predictive control. In: Proceedings of the 44th IEEE Conference on Decision and Control. IEEE; 2005, p. 1361–6.
[6] Tempo R, Calafiore G, Dabbene F. Randomized algorithms for analysis and control of uncertain systems: With applications. Springer; 2013.
[7] Ide J, Schöbel A. Robustness for uncertain multi-objective optimization: A survey and analysis of different concepts. OR Spectrum 2016;38:235–71.
[8] Ehrgott M, Ide J, Schöbel A. Minmax robustness for multi-objective optimization problems. European J Oper Res 2014;239:17–31.
[9] Hernández Castellanos CI, Schütze O, Sun JQ, Ober-Blöbaum S. Non-epsilon dominated evolutionary algorithm for the set of approximate solutions. Math Comput Appl 2020;25:3.
[10] Fischetti M, Monaci M. Light robustness. In: Robust and online large-scale optimization. Springer; 2009, p. 61–84.
[11] Schöbel A. Generalized light robustness and the trade-off between robustness and nominal quality. Math Methods Oper Res 2014;80:161–91.
[12] Engau A, Wiecek MM. Generating ε-efficient solutions in multiobjective programming. European J Oper Res 2007;177:1566–79.
[13] Loridan P. ε-Solutions in vector minimization problems. J Optim Theory Appl 1984;43:265–76.
[14] White DJ. Epsilon efficiency. J Optim Theory Appl 1986;49:319–37.
[15] Pajares A, Blasco X, Herrero JM, Reynoso-Meza G. A new point of view in multivariable controller tuning under multiobjetive optimization by considering nearly optimal solutions. IEEE Access 2019;7:66435–52.
[16] Pajares Ferrando A. Development of a multi-objective optimization methodology considering nearly optimal solutions. Application to problems in control engineering [Ph.D. thesis], 2019.
[17] Pareto V, et al. Manual of political economy. 1971.
[18] Schütze O, Coello CAC, Talbi EG. Approximating the ε-efficient set of an mop with stochastic search algorithms. In: Mexican international conference on artificial intelligence. Springer; 2007, p. 128–38.
[19] Schütze O, Vasile M, Coello CAC. Computing the set of epsilon-efficient solutions in multiobjective space mission design. J Aerosp Comput, Inf Commun 2011;8:53–70.
[20] Pajares A, Blasco X, Herrero JM, Reynoso-Meza G. A multiobjective genetic algorithm for the localization of optimal and nearly optimal solutions which are potentially useful: NevMOGA. Complexity 2018;2018.
[21] Martínez-Iranzo M, Herrero JM, Sanchis J, Blasco X, García-Nieto S. Applied pareto multi-objective optimization by stochastic solvers. Eng Appl Artif Intell 2009;22:455–65.
[22] Pajares A, Blasco X, Herrero JM, Martínez MA. A comparison of archiving strategies for characterization of nearly optimal solutions under multi-objective optimization. Mathematics 2021;9:999.
[23] Venn J. I. on the diagrammatic and mechanical representation of propositions and reasonings. Lond, Edinb, Dublin Philos Mag J Sci 10:1–18.
[24] Åström K, Bell R. Dynamic models for boiler-turbine alternator units: Data logs and parameter estimation for a 160 MW unit. Research reports TFRT-3192, Department of Automatic Control, Lund Institute of Technology (LTH); 1987.
[25] Garduno-Ramirez R, Lee KY. Multiobjective optimal power plant operation through coordinate control with pressure set point scheduling. IEEE Trans Energy Convers 2001;16:115–22.
[26] Tan W, Marquez HJ, Chen T, Liu J. Analysis and control of a nonlinear boiler-turbine unit. J Process Control 2005;15:883–91.
[27] Moon UC, Lee KY. A boiler-turbine system control using a fuzzy auto-regressive moving average (Farma) model. IEEE Trans Energy Convers 2003;18:142–8.
[28] Blasco X, Herrero JM, Sanchis J, Martínez M. A new graphical visualization of n-dimensional pareto front for decision-making in multiobjective optimization. Inform Sci 2008;178:3908–24.
[29] Blasco X, Herrero JM, Reynoso-Meza G, Martínez-Iranzo MA. Interactive tool for analyzing multiobjective optimization results with level diagrams. In: Proceedings of the genetic and evolutionary computation conference companion. 2017, p. 1689–96.