The final publication is available at

http://dx.doi.org/10.1016/j.conengprac.2014.03.003

Additional Information

Controller tuning using evolutionary multi-objective optimisation: current trends and applications

Gilberto Reynoso-Meza[a,∗], Xavier Blasco[a], Javier Sanchis[a], Miguel Martínez[a]

[a]*Instituto Universitario de Automática e Informática Industrial, Universitat Politècnica de València, Camino de Vera s/n , Valencia 46022, España*

Abstract

Control engineering problems are generally multi-objective problems; meaning that there are several specifications and requirements that must be fulfilled. A traditional approach to calculate a solution with the desired trade-off is to define an optimisation statement. Multi-objective optimisation techniques deal with this problem from a particular perspective searching for a set of potentially preferable solutions; the designer may then analyse the trade-off among them, and select the best solution according to his/her preferences. In this paper, this design procedure based on evolutionary multiobjective optimisation (EMO) is presented and significant applications on controller tuning are commented on. Through this paper it is noticed that EMO research has been developing towards different optimization statements, but such statements are not commonly used on controller tuning. Therefore gaps between EMO research and EMO applications on controller tuning are detected and suggested as potential trends for research.

Keywords: evolutionary multi-objective optimisation, multi-objective evolutionary algorithms, multi-criteria decision making, multi-objective optimisation design procedure, controller tuning.

1. Introduction

Satisfying a set of specifications and constraints required by real-control engineering problems is often a challenge. For parametric controller tuning for example, these range from time-domain specifications to frequency-domain requirements. Problems in which the designer must deal with the fulfillment of multiple objectives are known as multi-objective problems (MOPs).

It is common to define an optimisation statement to deal with MOPs and calculate a solution with the desired balance among (usually conflictive) objectives. When dealing with a MOP, we usually seek for a Pareto optimal solution [1] in which the objectives have

---

∗Corresponding author. Tel.:+34963877007;fax:+34963879579
Email address: `gilreyme@upv.es` (Gilberto Reynoso-Meza)
URL: `http://cpoh.upv.es/` (Gilberto Reynoso-Meza)

been improved as much as possible without giving anything in exchange. According to [2], there are two main approaches to solve an optimisation statement for a MOP: the *aggregate objective function* (AOF) or the *generate-first choose-later* (GFCL) approach.

In the AOF context a single-index optimisation statement merging the design objectives is defined. In such cases, the decision maker (DM or simply the designer) needs to describe all the trade-offs at once, at the beginning of the optimisation process. In the GFCL approach, the main goal is to generate many potentially desirable Pareto optimal solutions, and then to select the most preferable alternative. This is due to the impossibility to obtain a solution that is good for all objectives, and therefore several solutions with different trade-off levels may appear. The selection takes place in a Multi-Criteria Decision-Making (MCDM) step, where the task of the DM is to analyse the trade-off among objectives, and select the best solution according to his/her preferences.

One way to generate such sets of potential solutions in the GFCL approach is by means of multi-objective optimisation. This optimisation approach seeks for a set of Pareto optimal solutions to approximate what is known as the Pareto set [1, 3]. A Pareto set approximation could provide a preliminary idea of the objective space, and according to [4] it could be helpful when it is necessary to explain and justify the MCDM procedure. As drawbacks, more time and embedment of the DM in the overall process are necessary.

In order to approximate this Pareto set, classic optimisation techniques [1] and Evolutionary Multi-objective Optimisation (EMO) approaches have been used. In the latter case, Multi-objective Evolutionary Algorithms (MOEAs) have become a valuable tool to approximate the Pareto front for non-convex, non-linear and constrained optimisation instances [5, 6]. They have been used with success in several control systems [7] and engineering design [8] areas.

Regarding the GFCL framework, when the multi-objective optimisation process is merged with the MCDM step for a given MOP statement, it is possible to define a multi-objective optimisation design (MOOD) procedure [9]. Despite this MOOD procedure could not be suitable to substitute, in all instances, an AFO approach, it could be helpful in complex design problems, where a closest embedment of the designer is necessary. For example when an analysis on trade-off would be valuable for the DM, before implementing a desired solution.

In this paper, an overview of different applications and examples of MOOD procedures in control system engineering is provided. The paper is focused in this MOOD procedure since from a practical point of view, it is necessary to perform not only the optimisation but also the MCDM stage. Likewise only instances where the EMO is used in the optimisation process are commented. Therefore this means that optimisation statements using AOF approaches for MOPs are outside the scope of this paper. This work is not intended to present an exhaustive review of the literature, but to identify promising and potential areas of EMO in control systems. The rest of this paper is organised as follows: in Section 2 some definitions regarding MOP are given as well as the MOOD procedure. In Section 3, several applications of MOOD for PID, fuzzy, predictive and state space feedback controllers are discussed. Finally, some concluding remarks and possible trends for research are commented.

2

2. Multi-objective optimisation design procedure

A MOP, without loss of generality,[1] can be stated as follows:

$$\min_{\boldsymbol{\theta}} \boldsymbol{J}(\boldsymbol{\theta}) \quad = \quad [J_1(\boldsymbol{\theta}), \ldots, J_m(\boldsymbol{\theta})] \tag{1}$$

subject to:

$$\boldsymbol{g}(\boldsymbol{\theta}) \quad \leq \quad 0 \tag{2}$$
$$\boldsymbol{h}(\boldsymbol{\theta}) \quad = \quad 0 \tag{3}$$
$$\underline{\theta_i} \leq \theta_i \quad \leq \quad \overline{\theta_i}, i = [1, \ldots, n] \tag{4}$$

where $\boldsymbol{\theta} \in \Re^n$ is defined as the decision vector, $\boldsymbol{J}(\boldsymbol{\theta}) \in \Re^m$ as the objective vector, $\boldsymbol{g}(\boldsymbol{\theta})$, $\boldsymbol{h}(\boldsymbol{\theta})$ as the inequality and equality constraint vectors respectively; $\underline{\theta_i}, \overline{\theta_i}$ are the lower and upper bounds in the decision space for θ_i variable.

As remarked previously, there is no single solution because in general there is no solution that is best for all objectives. Therefore, a set of solutions, the Pareto set, is defined. Each solution in the Pareto set defines an objective vector in the Pareto front. All solutions in the Pareto front are said to be a set of Pareto-optimal and non-dominated solutions:

Definition (Pareto optimality [1]): An objective vector $\boldsymbol{J}(\boldsymbol{\theta}^1)$ is Pareto optimal if there is no other objective vector $\boldsymbol{J}(\boldsymbol{\theta}^2)$ such that $J_i(\boldsymbol{\theta}^2) \leq J_i(\boldsymbol{\theta}^1)$ for all $i \in [1, 2, \ldots, m]$ and $J_j(\boldsymbol{\theta}^2) < J_j(\boldsymbol{\theta}^1)$ for at least one $j, j \in [1, 2, \ldots, m]$.

Definition (Dominance [1]): An objective vector $\boldsymbol{J}(\boldsymbol{\theta}^1)$ is dominated by another objective vector $\boldsymbol{J}(\boldsymbol{\theta}^2)$ if $J_i(\boldsymbol{\theta}^2) < J_i(\boldsymbol{\theta}^1)$ for all $i \in [1, 2, \ldots, m]$.

For example, in Figure 1, five different solutions ($\Diamond$) are calculated to approximate a Pareto front (bold line). Solutions A, B, and C are non-dominated solutions, since there are no better solution vectors (in the calculated set) for all the objectives. Solutions B and C are not Pareto optimal, since some solutions (not found in this case) dominate them. Furthermore, solution A is also Pareto optimal, since it lies on the feasible Pareto front. The set of non-dominated solutions (A, B, and C) build the Pareto front approximation. It is important to notice that most of the times the Pareto front is unknown and we shall only rely on approximations.

In [2], an addendum is incorporated into the Pareto front notion to differentiate design concepts. A Pareto front is defined given a design concept (or simply, a concept) which is an idea about how to solve a given MOP. This design concept is built with a family of design alternatives (Pareto-optimal solutions) which are specific solutions in the design concept.

---

[1]A maximisation problem can be converted to a minimisation problem. For each of the objectives that have to be maximised, the transformation: $\max J_i(\boldsymbol{\theta}) = -min(-J_i(\boldsymbol{\theta}))$ could be applied.
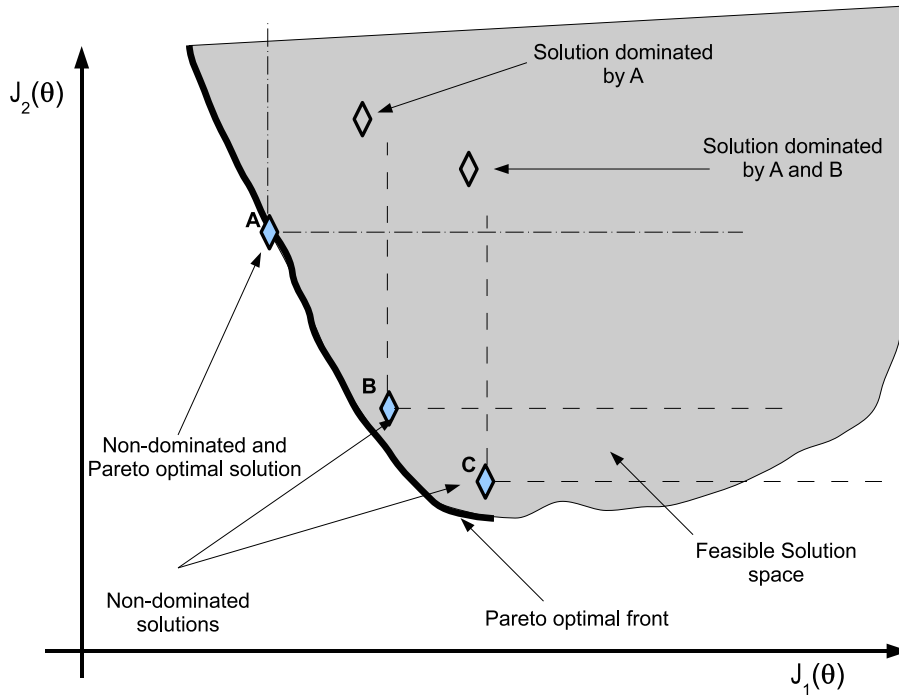
Figure 1: Pareto optimality and dominance concepts.

For example, in Figure 2, a Pareto front approximation (bold line) for a particular design concept is calculated with a set of Pareto-optimal design alternatives ($\Diamond$); we can state, for example, a PID controller for a given MOP as a design concept, where a design alternative is a specific set of values for its parameters.

As remarked in [2], a comparison between design concepts could be useful for the designer, because he will be able to identify the concepts strengths, weaknesses, limitations and drawbacks. It is also important to visualise such comparisons, and to have a quantitative measure to evaluate strengths and weaknesses.

A general framework is required to successfully incorporate this approach into any engineering design process. A multi-objective optimisation design (MOOD) procedure is shown in Figure 3. It consists of (at least) three main steps [5, 10]: the MOP definition (measurement); the multi-objective optimisation process (search); and the MCDM stage (decision making).

2.1. Multi-objective problem definition

At this stage the design concept is defined (how to tackle the problem at hand); the engineering requirements (what is important to optimise); and the constraints (which solutions are not practical/allowed). In [2] it is noticed that the design concept implies the existence of a parametric model that defines the parameter values (the decision space) that lead to a particular design alternative and its performance (objective space).
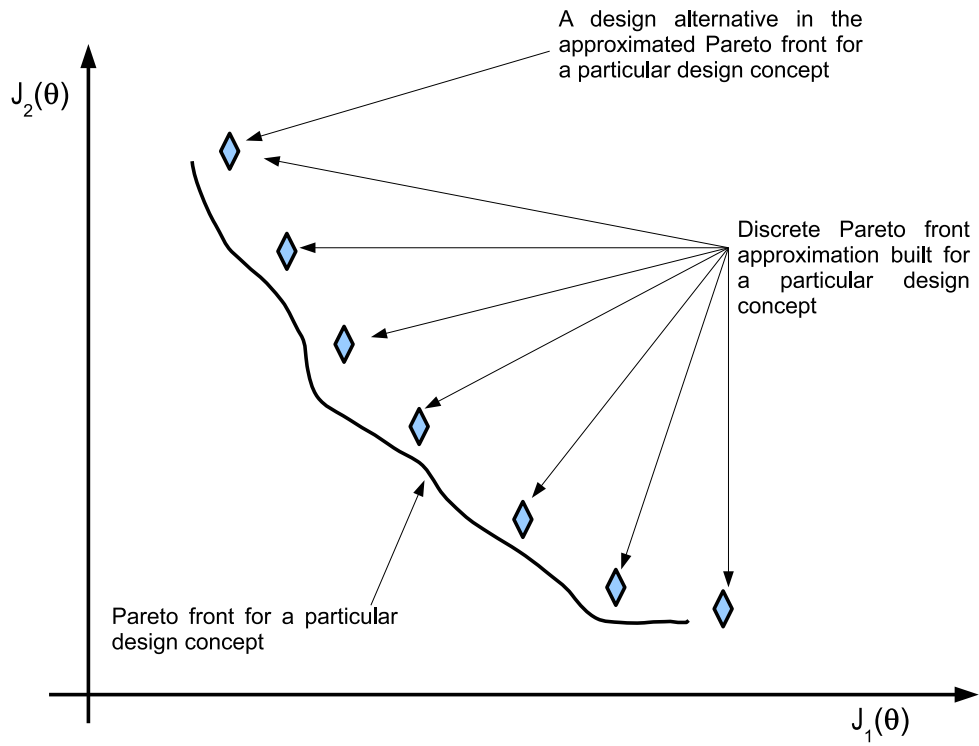
4

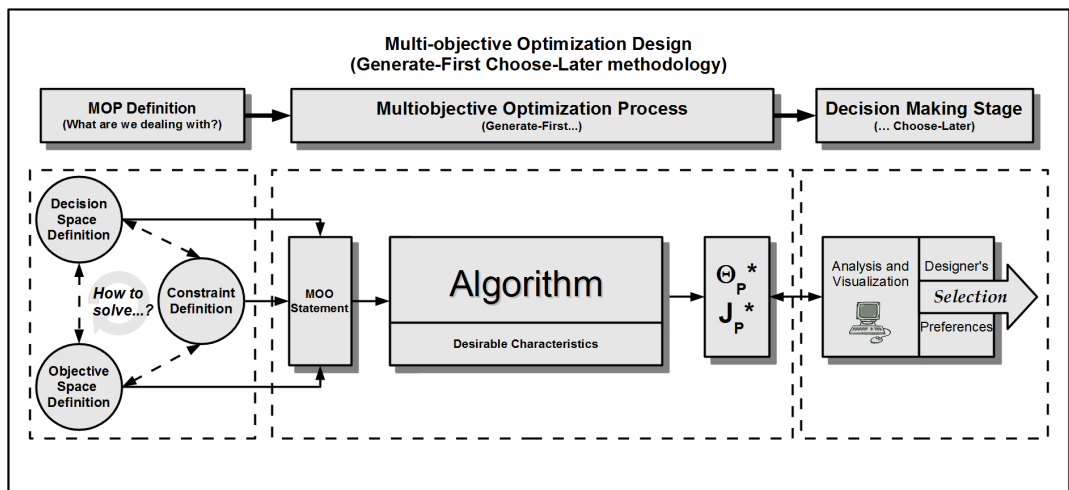Figure 2: Design concept and design alternative.

Figure 3: A multi-objective optimisation design (MOOD) procedure for control systems engineering.

It is worthwhile to notice how the selection of the optimisation objectives to measure the desired performance could be done. A convenient feature of using MOEAs is the flexibility to select interpretable objectives for the designer. That is, the objective selection could be closer to the point of view of the designer. Sometimes, with classical optimisation approaches, a cost function is built satisfying a set of requirements such as convexity and/or continuity; that is, it is built from the point of view of the optimiser, in spite of a possible loss of interpretability for the designer. Therefore, the multiobjective optimisation statement is not a trivial task, since the problem formulation from the point of view of the designer is not that of the optimiser [11].

Given the MOP definition some characteristics for the MOEA to be used could be required. That is, according to the expected design alternatives, the MOEA would need to include certain mechanisms or techniques to deal with the optimisation statement. Some examples are related with robust, multi-modal, dynamic and/or computationally expensive optimisation. Therefore, such instances could lead to certain desirable characteristics for the optimizer, which will be discussed below.

2.2. Evolutionary multi-objective optimisation

MOEA's have been used to approximate the Pareto set [12] due to their flexibility when evolving an entire population towards the Pareto front. A comprehensive review of the early stages of MOEAs is contained in [13]. There are several evolutionary and nature-inspired techniques used by MOEAs. The former are based mainly on the laws of natural selection, where the fittest members (solutions) in a population (set of potential solutions) have the best chance of survival as the population evolves. The latter is based on the natural behaviour of organisms. In both cases, they are used to evolve their populations towards the (unknown) Pareto front. We will refer to both simply as evolutionary techniques.

The most popular techniques seem to include Genetic Algorithms (GA) [14, 15], Particle Swarm Optimisation (PSO) [16, 17], and Differential Evolution (DE) [18, 19, 20]. Nevertheless, evolutionary techniques as Artificial Bee Colony (ABC) [21] or Ant Colony Optimization (ACO) [22] algorithms are becoming popular. No evolutionary technique is better than the others, since each have drawbacks and advantages. These evolutionary/nature-inspired techniques require mechanisms to deal with EMO since they were originally used for single objective optimisation. While the dominance criterion could be used to evolve the population towards a Pareto front, it could be not enough to achieve a minimum degree of satisfaction in other desirable characteristics for a MOEA (diversity for instance). In Algorithm I a general structure for a MOEA is given. Its structure is very similar to that of almost any evolutionary technique ([7]): it builds and evaluates an initial population $P|_0$ (lines 1-2) and begin with the optimization (evolutionary) process (lines 5 to 11). Inside this optimization process, the evolutionary operators (depending on the evolutionary technique) will built and evaluate a new population (line 7-8), and the solutions with better cost function will be selected for the next generation (line 10). The main difference is regarding line 9, where the Pareto set approximation is built; according to the requirements of the designer, such process will incorporate (or not) some desirable features.

Algorithm I: *MOEA.Basic*
1 :   Built initial population $P\|_0$ with N_p individuals.
2 :   Evaluate $P\|_0$.
3 :   Build initial Pareto set approximation $\boldsymbol{X}_P^*\|_0$
4 :   Set generation counter $G = 0$.
5 :   WHILE termination criteria not reached
6 : $G = G + 1$.
7 :   Built population $P^*\|_G$ using $P\|_G$ with an evolutionary or bio-inspired technique.
8 :   Evaluate population $P^*\|_G$.
9 :   Built Pareto set approximation $\boldsymbol{X}_P^*\|_G$ with $\boldsymbol{X}_P^*\|_{G-1} \bigcup P^*\|_G$
10:     Update population $P\|_{G+1}$ with $P^*\|_G \bigcup P\|_G$
11:     END-WHILE
12:     RETURN Pareto set approximation $X_P^*\|_G$

Desirable characteristics for a MOEA could be related to the set of (useful) solutions required by the DM or the MOP statement. Regarding the Pareto set sought, some desirable characteristics include (in no particular order) convergence, diversity and pertinency. Regarding the optimisation statement, some features could be related to deal with constrained, many-objectives, dynamic, multi-modal, robust, computationally expensive or large scale optimisation instances. These desired characteristics are also a guide to appreciate current trends and on going research on EMO and MOEA's development [23], [12]. Some of them are explained below.

2.2.1. Convergence

It refers to the algorithm's capacity to reach the real (usually unknown) Pareto front (Figure 4a). It is known that convergence properties depends on the own evolutionary parameters of a MOEA, modifying its exploitation and exploration capabilities [24]. In this sense, several adaptation mechanisms are provided as well as several ready-to-use MOEAs with a default set of parameters, adjusted according several benchmarks.

2.2.2. Diversity mechanism

Diversity refers to the algorithm's capacity to obtain a set of distributed solutions that provide a useful description of objective trade-off and decision variables (Figure 4b). Popular ideas include pruning mechanisms [25, 26, 27, 28, 29, 30], spreading measures [31] or performance indicators [32] of the approximated front.

2.2.3. Pertinency

Incorporating DM preferences into the MOEA has been suggested to improve the pertinency of solutions (see for example [33, 34]). That is, the capacity to obtain a set of interesting solutions from the DM point of view (Figure 4c). The designer's preferences could be defined in the MOOD procedure in an *a priori, progressive*, or *a posteriori* fashion [35]. A straightforward alternative to improve pertinency of solutions could be done by means of optimisation constraints (besides bound constraints on decision variables).
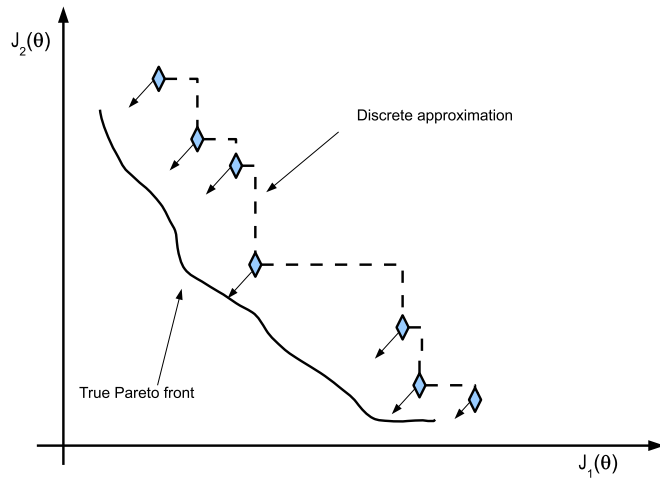
(a) Convergence
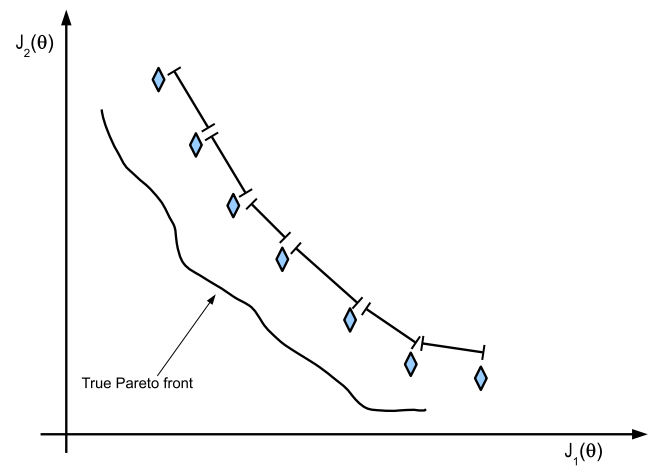
(b) Diversity

(c) Pertinency

Figure 4: Convergence, diversity and pertinency concepts.

8

2.2.4. Constrained optimisation

Given that most of the design optimisation problems need to consider constraints, another desirable characteristic could be a constraint handling mechanism. In this sense, various techn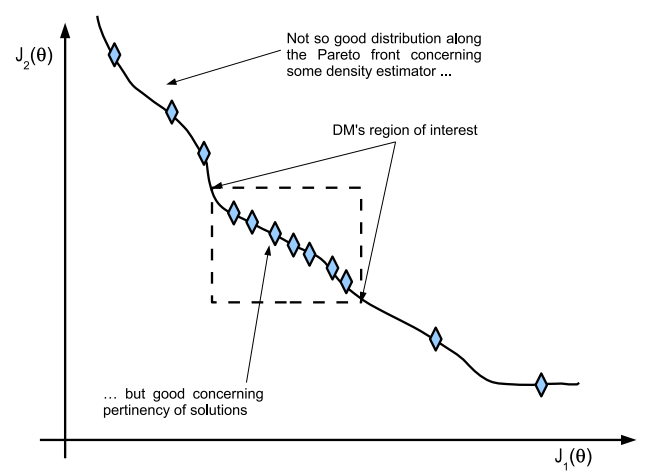iques have been developed over the years [36, 37, 38] for evolutionary optimisation. The latter classifies current approaches as: feasibility rules [39], ϵ-constrained method [40], multiobjective concepts [41] and ensemble of constraint-handling techniques [42].

2.2.5. Many-Objectives optimisation

Algorithms with good diversity preservation mechanisms could face problems if solutions are dominance resistant in an m-dimensional objective space. This situation could lead to waste time and CPU resources searching solutions in non-optimal areas [43]. This is because of the self diverse nature and the large number of objectives (usually, $m \geq 5$). In [44], a review of many-objectives optimisation techniques is given.

2.2.6. Dynamic optimisation

Sometimes the static approach is not enough to find a preferable solution and therefore, a dynamic optimisation statement needs to be solved where the cost function is varying with time. In the case of MOOD procedure, the required task besides tracking the optimal solution, is to select the desired solution at each sampling time. In [45, 46] there are extensive reviews on the topic.

2.2.7. Multi-modal optimisation

In multi-modal optimisation, different decision vectors could bring the same objective vector. In some instances, it could be desirable to mind different solutions even if they have the same objective vector in the MCDM step. This could be important in instances where, for example, the decision variables have a physical meaning and it is convenient to analyse the impact of implementing one over another. For more details on multi-modal optimisation, the interested reader could refer to [47].

2.2.8. Robust optimisation

In a general frame and according to [48], robust optimisation could refer not only to the models used to measure the performance, but also with the sensitivity analysis on the calculated solutions. That is, how much could be degraded the objective vector under the presence of uncertainties. This sensibility analysis could be done by means of deterministic measures and/or with direct search (as Montecarlo methods). This kind of analysis could bring a different level of interpretability of the performance due to uncertainties in the model used in the optimisation. This problem statement is related with reliability optimisation, where a given performance must be assured for a certain process along different scenarios.

2.2.9. Computationally expensive optimisation

Computationally expensive optimisation is related with line 8 in Algorithm I. Sometimes the cost function to evaluate require a huge amount of computational resources. Therefore

stochastic approaches could face a problem, given the complexity to evaluate the fitness (performance) of an individual (design alternative); it could affect their exploration capabilities and hence, slow down its convergence properties. A review on the topic can be consulted in [49].

2.2.10. Large scale optimisation

It refers to the capabilities of a given MOEA to deal with MOP with any number of decision variables, with reasonable computational resources. Sometimes a MOEA could have remarkable convergence properties for a relatively small number of decision variables, but it could be intractable (according to the computational resources available) to solve a problem with a bigger number of decision variables. Whilst in expensive optimisation instances (Section 2.2.9) the complexity is due to the performance measurement (line 8 in Algorithm I), in large scale may be related to the algorithm's mechanism used to approximate a new set of design alternatives (lines 7 and 9 in Algorithm I). In the former the complexity is added by the problem, in the latter by the algorithm. A review on this topic can be consulted in [50].

2.3. Multi criteria decision making

Once the DM has been provided with a Pareto front $\boldsymbol{J}_P^*$, she/he will need to analyse the trade-off between objectives and select the best solution according to her/his preferences. A comprehensive compendium on MCDM techniques (and software) for multi-dimensional data and decision analysis can be consulted in [51]. Assuming that all preferences has been minded as much as possible in the optimisation stage, a final selection step could be done with the approximated Pareto front $\boldsymbol{J}_P^*$.

It is widely accepted that visualisation tools are valuable and provide the DM with a meaningful method to analyse the Pareto front and take decisions [52]. Tools and/or methodologies are required for this final step to successfully embed the DM into the solution refinement and selection process. It is useful for the DM to understand and appreciate the impact that a given trade-off in one sub-space could have on others [4]. Even if an EMO process has been applied to a reduced objective space, sometimes the DM needs to increase the space with additional metrics or measurements to have confidence in her/his own decision [4]. Usually, analysis on the Pareto front may be related with design alternatives comparison and design concepts comparison.

For two-dimensional problems (and sometimes for three-dimensional problems) it is usually straightforward to make an accurate graphical analysis of the Pareto front, but the difficulty increases with the dimension of the problem. In [52], a review on visualisation techniques includes techniques such as decision maps, star diagrams, value paths, GAIA, and heatmap graphs. Possibly the most common choices for Pareto front visualisation and analysis in control systems applications are: scatter diagrams, parallel coordinates [53], and level diagrams [54, 55].

In any case, the characteristics required for such a visualisation were described in [52]: simplicity (must be understandable); persistence (information must be rememberable by the

DM); and completeness (all relevant information must be depicted). Some degree of inter-activity with the visualisation tool is also desirable (during and/or before the optimisation process) to successfully embed the DM into the selection process.

2.4. Conclusions on this section

In this section, some topics on MOP definitions, EMO and MCDM have been covered. The aforementioned steps are important to guarantee the overall performance of a MOOD procedure. With a poor MOP definition, not matter how good the algorithms and MCDM methodology/tools are, the solutions obtained will not fulfill the DM's expectations. If the algorithm is inadequate for the problem at hand (regarding the desirable characteristics from section 2.2.1 to 2.2.10), the DM will not obtain a useful Pareto set to analyse and therefore he/she will not be able to select a solution that meets his/her preferences. Finally, the incorrect use of MCDM tools and methodologies could imply a lower degree of embedment of the DM in the trade-off analysis and the solution selection. The last issue could easily discourage the DM from using a MOOD procedure.

Regarding the MOP, some comments have been made regarding the capacity to reach a different level of interpretability on objective functions. In the MOOD approach there is no need to built a complicated aggregated function to merge the design objectives; therefore the objectives may be minded separately and optimised simultaneously. That is, the objective function statement could be done from the needs of the designer instead of the optimiser. This could facilitate the embedment of the designer into the overall procedure. In the case of EMO, it has been exposed how MOEAs could be useful to face different optimisation instances as well as bring some desirable characteristics to the approximated Pareto front. It is important to remember that the final purpose of any MOEA is to provide the DM with a useful set of solutions (Pareto front approximation) to perform the MCDM procedure [4].

Finally, with regard to the MCDM step, it can be noticed that visualisation of the Pareto front is a desirable tool for the DM to perform his/her selection at the MCDM stage. Below, different design concepts and MOP statements used in control system engineering field will be discussed.

3. Design applications in controller tuning

Since the MOOD procedure provides the opportunity to obtain a set of solutions to describe the objective trade-off for a given MOP, it is worthwhile to use it for controller tuning. Due to the fact that several specifications such as time and frequency requirements need to be fulfilled by the control engineer, a procedure to appreciate the trade-off exchange for complex processes could be useful. Controller design concepts such as PID, fuzzy, state-space feedback and predictive controllers are covered, where by means of the MOOD procedure, the designer is seeking to improve their performance.

In [7], a review on the early stages of MOEAs in control systems is provided. Design applications to be in the scope of this work may include parametric controller tuning on the following 10 years since the aforementioned review (from 2002-2012). Works on parametric model identification are not included since comprehensive reviews on modeling (in broad

sense) using EMO are available for fuzzy systems [56], neural networks, [57, 58], machine learning [59] and support vector machines [60].

We will include relevant applications, where desirable characteristics as convergence and diversity are considered as expected. This means that works where a proof of concept of an EMO for solving a given problem or where a simple comparison between MOEA's is provided are intentionally omitted. Work with depth analysis, bringing usage of new desirable characteristics of MOEA's for controller tuning will be included, commented and discussed. This is because it is intended to offer to the interested reader a useful framework on work *to-be-done* and *already-done* on MOOD procedures for controller tuning.
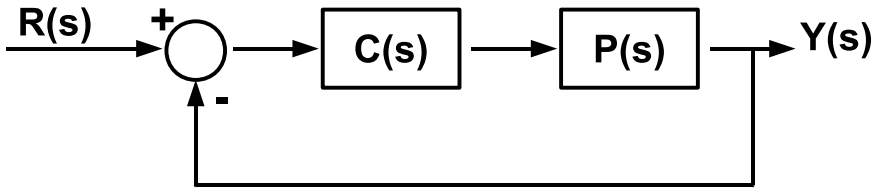
Figure 5: Basic control loop.

According to the basic control loop of Figure 5 and the review, some common choices for objectives in frequency domain are:

- Maximum value of sensitivity function

$$J_{M_s}(\boldsymbol{\theta}) = \left\| (I + P(s)C(s))^{-1} \right\|_\infty \tag{5}$$

- Disturbance attenuation performance

$$J_{W_{M_s}}(\boldsymbol{\theta}) = \left\| W(s) \cdot (I + P(s)C(s))^{-1} \right\|_\infty < 1 \tag{6}$$

- Maximum value of the complementary sensitivity function

$$J_{M_p}(\boldsymbol{\theta}) = \left\| P(s)C(s)(I + P(s)C(s))^{-1} \right\|_\infty \tag{7}$$

- Robust stability performance.

$$J_{W_{M_p}}(\boldsymbol{\theta}) = \left\| W(s) \cdot (P(s)C(s)(I + P(s)C(s))^{-1}) \right\|_\infty < 1 \tag{8}$$

where $W(s)$ are weighting transfer functions commonly used in mixed sensitivity techniques. Meanwhile in time domain:

- Integral of the absolute error value

$$J_{IAE}(\boldsymbol{\theta}) = \int_{t=t_0}^{T_f} |r(t) - y(t)| \, dt \tag{9}$$

12

- Integral of the time weighted absolute error value

$$J_{ITAE}(\boldsymbol{\theta}) = \int_{t=t_0}^{T_f} t\,|r(t) - y(t)|\,dt \tag{10}$$

- Integral of the squared error value

$$J_{ISE}(\boldsymbol{\theta}) = \int_{t=t_0}^{T_f} (r(t) - y(t))^2\,dt \tag{11}$$

- Integral of the time weighted squared error value

$$J_{ITSE}(\boldsymbol{\theta}) = \int_{t=t_0}^{T_f} t\,(r(t) - y(t))^2\,dt \tag{12}$$

- Integral of the control action value

$$J_{ISU}(\boldsymbol{\theta}) = \int_{t=t_0}^{T_f} (u(t))^2\,dt \tag{13}$$

- Total variation of control action

$$J_{TV}(\boldsymbol{\theta}) = \int_{t=t_0}^{T_f} \left|\frac{du}{dt}\right| \tag{14}$$

- Maximum value of control action

$$J_{maxU}(\boldsymbol{\theta}) = \max(u(t)), t \in [t_0, T_f] \tag{15}$$

where $r(t)$, $y(t)$, $u(t)$ are the reference, measured variable and control action in time t. Such objectives, for the sake of simplicity, have been stated in a general sense; details regarding specific implementation issues could be consulted, by the interested reader, in each case.

13

3.1. PI-PID controller design concept

PID controllers are reliable digital control solutions due to their simplicity and efficacy [61, 62]. They represent a common solution for industrial applications and therefore, there is still ongoing research on new techniques for robust PID controller tuning [63]. Any improvement in PID tuning is worthwhile, owing to the minimum number of changes required for their incorporation into already operational control loops [64, 65]. As expected, several works have been focus on the PID's performance improvement.

Given a process $P(s)$, the following general description for a PID controller is used:

$$
\begin{aligned}
C(s) &= k_p \left(a + \frac{1}{T_i s} + b \frac{T_d}{\frac{T_d}{N} s + 1} \right) R(s) \\
&+ k_p \left(1 + \frac{1}{T_i s} + \frac{T_d}{\frac{T_d}{N} s + 1} \right) Y(s)
\end{aligned}
\tag{16}
$$

where k_p is the controller proportional gain, T_i the integral time, T_d the derivative time, N the derivative filter, a, b the setpoint weighting for proportional and derivative actions, $R(s)$ the reference and $Y(s)$ the measured signal. Therefore, the following design concepts (controllers) with their decision variables could be stated:

PI: $\boldsymbol{\theta}_{PI} = [K_p, T_i]$

PD: $\boldsymbol{\theta}_{PD} = [K_p, T_d]$

PID: $\boldsymbol{\theta}_{PID} = [K_p, T_i, T_d]$

PID/N: $\boldsymbol{\theta}_{PID/N} = [K_p, T_i, T_d, N]$

PI^1: $\boldsymbol{\theta}_{PI^1} = [K_p, T_i, a]$

PID^2: $\boldsymbol{\theta}_{PID^2} = [K_p, T_i, T_d, a, b]$

PID^2/N: $\boldsymbol{\theta}_{PID^2/N} = [K_p, T_i, T_d, N, a, b]$

3.1.1. Applications on PI-PID controller design concept

In [66], authors focus on defining a general MOP statement for a wide variety of applications and it is one of the few works using as decision variables all the flexibility degrees of a PID controller (gains, weighting factors and filters). It proposes an MOP using four objectives for a PID^2/N controller. The aforementioned objectives are related with different norms of $J_{IAE}(\boldsymbol{\theta}_{PID^2/N})$, $J_{Ms}(\boldsymbol{\theta}_{PID^2/N})$, $J_{ITAE}(\boldsymbol{\theta}_{PID^2/N})$ (equations 9, 5 and 10 respectively) and $J(\boldsymbol{\theta}_{PID^2/N}) = \|W(s) \cdot (I + P(s)C(s))^{-1}\|_2$ for noise reduction. Constraints to avoid saturation effects are also included. The PI^1 and PID^2/N concepts are compared (using SCp visualization) in order to appreciate the trade-off differences in the MOP given for single input single output and multiple input multiple output processes (SISO and MIMO respectively).

[67] seems to be the first work that provides tuning rules for a MOP statement; that is, defining tuning rules with a certain degree of flexibility for the designer to select a desirable controller according to her/his preferences. It uses an integral MOOD tuning procedure for PI[1] controllers to build a multi-objective-based tuning rule procedure. The methodology begins with an identified first order plus dead time model to perform the optimisation and MCDM step procedure. In the first step, a 3D Pareto front is analysed to select a solution with the desired trade-off among $J_{IAE}(\boldsymbol{\theta}_{PI^1})$ (for load disturbance), $J_{Ms}(\boldsymbol{\theta}_{PI^1})$ and $J_{TV}(\boldsymbol{\theta}_{PI^1})$ (equations 9, 5 and 14 respectively). A second analysis in a 2D Pareto front is then performed to select the proper value of the setpoint weighting for setpoint response. The Pareto fronts are built according a given value of the normalized time delay. The approach was validated with several SISO processes with different characteristics.

In [68], a EMO constrained statement for a PID/N controller is proposed for SISO processes. Three main objectives are defined: a low frequency performance objective based on $J(\boldsymbol{\theta}_{PID/N}) = P(s)(I + P(s)C(s))^{-1}$, a pass-band robust objective using an aggregate function of $J_{M_s}(\boldsymbol{\theta}_{PID/N})$ and $J_{M_p}(\boldsymbol{\theta}_{PID/N})$ (equations 5 and 7 respectively), and a control activity objective based on $J(\boldsymbol{\theta}_{PID/N}) = C(s)(I + P(s)C(s))^{-1}$. Constraints are also incorporated to improve the pertinency of solutions. The MOEA proposed in the paper uses a single-objective optimisation procedure in a subset of its population to improve convergence and speed. A SISO example is provided and after several comparisons (using SCp visualization), the authors noticed that using only frequency domain objectives does not give enough information to the designer regarding pertinency in time domain specifications. Therefore the authors suggest that mixing objectives from both domains would significantly improve the interpretability and pertinency of the design alternative.

In [69], an MOEA using a simulated annealing (SA) based algorithm [70] is proposed for PID[2]/N controller tuning for square MIMO processes. The algorithm uses a generation mechanism based on orthogonal experiment designs; this mechanism is used in order to incorporate a type of *systematic reasoning method* to generate promising Pareto optimal solutions. The design objectives used are the sensitivity and complementary sensitivity functions $J_{W_1}(\boldsymbol{\theta}_{PID^2/N})$, $J_{W_2}(\boldsymbol{\theta}_{PID^2/N})$ and squared error $J_{ISE}(\boldsymbol{\theta}_{PID^2/N})$ (equations 6, 8 and 11 respectively). The algorithm is compared with a state-of-the-art algorithm and is evaluated in an aircraft simulated model. In this proposal, the constraint-handling and pertinency mechanisms are supported by the weighting function approach employed in the mixed sensitivity technique. Comparison with other techniques is shown, where a mixed 2-norm and 1-norm criterion is used to select a preferable solution.

In [71], a PID[2]/N controller was tuned using a MOOD procedure. The controller employed a hybrid constraint approach using mixed sensitivity constraints as objectives. The process under consideration was a black-box non-linear model with three different dynamics (underdamped, damped, and unstable) according to the operational zone. A set of nominal models was identified using a prior EMO statement and a total of 15 objectives (5 for each operational zone) based on $J_{IAE}(\boldsymbol{\theta}_{PID^2/N})$, $J_{ITAE}(\boldsymbol{\theta}_{PID^2/N})$, $J_{TV}(\boldsymbol{\theta}_{PID^2/N})$, $J_{W_1}(\boldsymbol{\theta}_{PID^2/N})$ and $J_{W_2}(\boldsymbol{\theta}_{PID^2/N})$ (equations 9, 10, 14, 6 and 8 respectively) were stated. A subset of design alternatives was selected, according to the 2-norm from the normalised ideal solution in an

LD visualisation framework.

In [72] the optimisation statement developed by [73] for PI[1] controllers is stated as a EMO problem for (square) MIMO processes. The purpose was to analyse the trade-off between integral gain $J_{k_i}(\boldsymbol{\theta}_{PI^1}) = -\frac{k_p}{T_i}$, maximum value of the sensitivity function $J_{M_s}(\boldsymbol{\theta}_{PI^1})$ (equation 5) and maximum value of the complementary sensitivity function $J_{M_p}(\boldsymbol{\theta}_{PI^1})$ (equation 7) for setpoint changes (one triplet for each loop); additionally as overall robust performance the largest log modulus defined in [74] is stated. The trade-off is discussed, and the design alternatives in the Pareto front are compared with other AOF and EMO approaches using a LD visualization.

In [75], a PID controller is tuned for a flexible alternating current transmission system. The overall goal was to find a controller capable of improving the load disturbance rejection performance with the minimum control effort. Two objectives were defined for this purpose: error measurement $J_{ISE}(\boldsymbol{\theta}_{PID})$ and control effort $J_{ISU}(\boldsymbol{\theta}_{PID})$ (equations 11 and 13 respectively). The NSGA-II algorithm was used, and the selection procedure was performed using preference articulation by means of a fuzzy inference system. In this inference system, the minimum and maximum values of each objective in the calculated Pareto front are used to calculate the most preferable solution.

It has been noticed that the MOOD procedure could be more time consuming than an AOF approach. Nevertheless, sometimes is posible to run the optimization for a given nominal case, and use it as basis for a particular (different) statement. A good example is provided in [76] where two design concepts, a PID and an I-PD controller (a PID[2] controller with $a = 0$, $b = 0$) are tuned using a MOEA for an optimal chemotherapy control model for cancer treatment. Both controllers are used to supply drug doses for a given period of time. Three objectives are defined: (maximise) tumour cell killing, (minimise) toxicity, and (minimise) tolerable drug concentration. The MOOD approach is used to observe and evaluate the effects of treatment. The purpose is to develop a *reference case* on the performance of the controllers to be used by the medical staff to determine the most convenient treatment for a particular patient (since it is impractical to generate a Pareto front for *each* patient). To improve the pertinency of solutions, a goal vector approach is used. The visualisation and selection procedure were performed using SCp when comparing both design concepts. The visualisation process was useful, since it was required to perform an in-depth analysis on the characteristics and physical meaning of the solutions.

In [77] a set of four PI controllers is proposed for the ALSTOM gasifier problem [78]. This gasifier is a popular MIMO popular benchmark, [2] and a wide variety of control structures have been developed and evaluated on it. It is a non-linear model of a coal gasifier, with several constraints. The defined MOP consisted of six objectives, each related with the integral of the absolute error $J_{IAE}(\boldsymbol{\theta}_{PI})$ (equation 9) for different load and disturbance scenarios. An SCp visualisation in the decision space is used to analyse the performance of the calculated PI controllers; the MCDM selection was performed by using a filtering process on the Pareto front approximation and a coal quality indicator (*i.e.* a new measurement is

---

[2]Simulink models available at `http://www-staff.lboro.ac.uk/~elrd2/`

16

included in the MCDM process). This shows the flexibility that can be incorporated in the MCDM step, with new indicators included to evaluate the potential sets of solutions, before taking a final selection.

In [79], a double-acting hybrid magnetic thrust bearings and its controller were designed and optimised using a unified MOOD framework. A GA-based MOEA was used to optimise an MOP statement with 5 objectives, 14 constraints and 16 decision variables. Regarding the decision space, two decision variables correspond to a PD controller design concept; regarding objective space, 3 objectives are related to mechanical properties (exerted force, powerloss and weight) and two with control performance: $J_{ISE}(\boldsymbol{\theta}_{PD})$ and $J_{ISU}(\boldsymbol{\theta}_{PD})$ (equations 11 and 13 respectively). Scatter diagram visualisation is used in the MCDM stage and as selection criteria, the closest solutions to the utopia point were selected and compared. In this case, this application could be an example of *holistic mechanical-and-control design*. That is, a multi-disciplinary design approach, where control and geometric decision variables are optimized on the same level.

In [80] a multivariable PI controller structure is tuned for two MIMO processes: a distillation column and an aircraft model. A bi-objective optimisation statement is defined, using an AOF for robust stability and disturbance attenuation by means of $J_{W_1}(\boldsymbol{\theta}_{PI})$ and $J_{W_2}(\boldsymbol{\theta}_{PI})$ (equations 6 and 8 respectively). As a second objective, the integral of the squared error $J_{ISE}(\boldsymbol{\theta}_{PI})$ (equation 11) is stated; a PSO-based MOEA (lb-MOPSO) is proposed and compared with NSGA-II. Norm 1 criteria are used to compare controllers from the Pareto front with other design concepts.

In [81, 82] a controllability analysis of electronic valves using MOEAs is performed. A given valve is evaluated with respect to seven different performance objectives, based on measure of transition time from release to landing, applied forces, landing speed and armature velocity. The main goal is to determine the Pareto-optimal set of candidate actuators fulfilling a set of control requirements. Further analysis on the Pareto front is performed using PAc visualisation. In the light of this visualisation, several remarks are made about the trade-off between objectives. This is a good example of how MOOD methodology can be used to design components with a guaranteed degree of controllability, *i.e.*, a design for controllability.

3.1.2. Conclusions on PI-PID controller design concept

In table 1 a summary on these applications is provided. Brief remarks on MOP, EMO and MCDM for each work were given. From the above the following can be noticed:

Regarding the MOP statement, although works as [73], [83] and [84] noticed the importance of considering decision variables as N, a, b as an integral part of PID tuning procedure, few works focus on using those decision variables within an integral PID tuning methodology.

Also, several works focus mainly in SISO plants and sought for tuning a single PID controller. In this sense, due to the fact that several and well established tuning procedures exists, there is a missing justification on when a MOOD approach could bring better solutions to the designer. Such justification is not related with the optimization problem *per se*; that is, is not about the difficulty to find a solution, but the difficulty to find a solution with a reasonable trade-off on conflicting objectives. In this framework, the MOOD approach for

SISO loops could be justifiable.

With regard to MIMO processes, it seem to be a promising area where the capabilities of MOOD procedure could be evaluated. This due to the fact that MIMO processes could be more complex and a reasonable trade-off on conflicting objectives could be hard to find. In this case, dealing with MIMO processes, mechanisms for many-objectives and expensive optimization could be valuable.

It is also interesting to notice those works were multidisciplinary optimisation is performed, merging for example control and mechanical decision variables. It could lead to interesting optimisation instances, since the shape of the decision space will be different that the one of control parameters. Multimodal optimisation instances could be interesting to apply in such problems, since physical variables need to be minded.

Regarding the optimizer, MOEAs based on GA seem to be more popular for PID controller tuning than their counterparts. This leads to the following question: *are the operators used in GA best suitable for PID controller tuning? is the stochastic search employed by MOEA's based on GA capable of finding better Pareto front approximations, in the case of PID tuning?* There is no work addressing this question, which could be useful for practitioners and designers when selecting an algorithm.

A less considered topic about the stochastic search used in MOEAs for PID controller tuning, is about the proper selection of decision variable bounds. That is, *how to select the $[\underline{\theta}_i, \overline{\theta}_i]$ for each $\theta_i \in \boldsymbol{\theta}$ minding internal closed-loop stability?* While commonly uncommented, this is a primary constraint in controller tuning. In this sense, given the stochastic nature of MOEA's, it could be important to fulfill the following two requirements for the stochastic sampling:

1. Any sampled controller must stabilize the closed loop.
2. Any stabilizing controller $C(s)$ of the process $P(s)$ must be contained in $[\underline{\theta}_i, \overline{\theta}_i]$, $\theta_i \in \boldsymbol{\theta}$.

A common approach for feature 1 is to define bounds on the parameter which avoid all non-stable, but also some stable PID parameters; therefore, feature 2 is not fulfilled. A second alternative, is to bound the search space with all stable PID parameters, but including non-stable parameters which are verified whilst the algorithm is running; this obviously doesn't fulfill feature 1, and could misspent computational (CPU time) resources. Therefore, techniques for stochastic sampling addressing the two requirements could be interesting.

It seems that the MOOD procedure brings a most suitable framework to include constraints in the optimisation problem, in comparison with evolutionary approaches for AOF statements for PID controllers [85]. It seems that the simplicity of the PID and algorithms available have allowed to state many-objectives optimisation problems. Whilst it seems that the optimisers incorporate quite well some constraint handling mechanism, is not usual to find applications with preferences handling incorporated in the algorithm to improve pertinency (in an *a-priori* or *interactive* sense).

Finally, in the MCDM, classical approaches for visualization based on SCp and 3D representation are the most used, despite the number of objectives defined for the MOP. However, guidelines on this procedure, although valuable, are not well covered.

Table 1: Summary of MOOD procedure for PID design concept. $J(\theta)$ refers to the number of objectives; θ to the number of decision variables and $g(\theta)$, $h(\theta)$ to the number of inequality and equality constraints, respectively (according with examples provided in each paper). In some instances, constraints are also stated as objectives.

			MOP			EMO		MCDM	
Concept(s)	Process(es)	Ref	$J(\theta)$	θ	$(g(\theta), h(\theta))$	Algorithm	Related features from section 3	Plot	Selection insights
PID^2/N, PI^1	SISO, MIMO	[66]	4	7	(2,0)	GA	2.2.4	3D, SCp	Concepts comparison.
PI^1	FOPDT	[67]	4	3	(0,0)	GA		3D, 2D	Tuning rule methodology.
PID	Electro-magnetic	[82]	7	27	(2,0)	GA	2.2.5; 2.2.4; 2.2.3	PAc	Iterative controllability analysis for a given design.
PID/N	SISO	[68]	3	4	(4,0)	Ad hoc	2.2.4; 2.2.3	SCp	Incorporate analysis of time domain objectives.
PID	Aeronautical	[69]	3	27	(2,0)	Ad hoc	2.2.4	SCp	Analysis with other techniques.
PD	Mechatro-nical	[79]	5	14	(14,0)	GA	2.2.4; 2.2.5	SCp	Design alternatives comparison.
PID^2/N	SISO	[71]	15	7	(6,0)	GA	2.2.4; 2.2.5	LD	Selection according to preferences.
P + PI (3)	Chemical	[77]	6	8	(1,0)	NSGA-II	2.2.4; 2.2.5	SCp	New indicator included for selection.
PID	Electrical	[75]	2	3	(0,0)	NSGA-II		None	Fuzzy based selection.
PID, I-PD	Bio-medical	[76]	2	3	(3,0)	GA	2.2.4	SCp	Concepts comparison; intended to support specific treatment.
PID	Chemical, Aeronauti-cal	[80]	2	27	(1,0)	PSO	2.2.4	SCp	Norm 1 selection.
PI	Chemical	[72]	7	4	(3,0)	DE	2.2.4; 2.2.5	LD	Trade-off analysis.

3.2. Fuzzy controller design concept

Fuzzy systems have been widely and successfully used in control systems applications as referred in [86]. As in the PID design concept, the MOOD is useful for analysing the trade-off between conflicting objectives. In this case, the fuzzy controller is more complex to tune, given its non-linearity. A comprehensive compendium on the synergy between fuzzy tools and MOEAs is given in [56]. Here, we focus on controller implementations. In general, decision variables regard to $\theta = [\Lambda, \Upsilon, \overline{\overline{\Lambda}}, \overline{\overline{\Upsilon}}, \mu]$ where:

Λ: is the membership function shape.

$\overline{\Lambda}$: is the number of membership functions.

Υ: is the fuzzy rule structure.

$\overline{\overline{\Upsilon}}$: is the number of fuzzy rules.

μ: are the weights of the fuzzy inference system.

3.2.1. Applications on fuzzy controller design concept

In [87] a fuzzy logic controller was adjusted using NSGA-II for a base-isolation system using a magnetorheological damper. A total of four objectives were stated, based on the maximum displacement, maximum acceleration and their root mean squared values for different earthquakes profiles using as decision variables $\boldsymbol{\theta} = (\Lambda, \Upsilon, \mu)$. Afterwards, a design concept comparison with a skyhook controller was performed using an SCp visualisation. It was remarked that with regard to a pair of objectives, there was practically no difference among concepts (using as a basis the number of dominated solutions). Nevertheless when four objectives were considered, the MOOD tuning procedure using NSGA-II outperformed the other design concept. This is a good example of how we can ask more from an existing tuning rule by adding new objectives and indexes to discriminate solutions; nevertheless, in this work it was remarked that the more objectives, the greater the ratio of non-dominated solutions. This issue is typical in many-objectives optimisation problems, as it was discussed above.

In [88], a fuzzy scheduling controller is adjusted using an MOEA for a gas turbine engine model. This is a complex aerospace system where several considerations (such as safety, reliability, and maintainability) need to be minded as control requirements. A MOGA algorithm is used to tune the membership functions Λ , and the scaling factors μ of the fuzzy controller. A total of nine different objectives were stated: rise time and settling time of the compressor, engine thrust and its rise time, maximum nozzle and turbine temperatures, the fan pressure ratio, the low pressure surge margin and thermodynamic stress. A PAc visualisation is used to analyse the trade-off among design alternatives. In this case, no controller was capable of fulfilling all the requirements, and an extensive analysis of the conflicting objectives was made using a trade-off matrix to find suitable controllers. This is a good example of the designer interacting with the optimiser to improve the pertinency of the solutions (progressive preferences articulation).

In [89], a MOOD is used for parameter tuning of a fuzzy controller for vibration suppression on smart structures. Objectives based on $J_{ISE}(\Lambda, \mu, \Upsilon)$ (equation 11) for nodal displacements, velocities and accelerations are stated. Several tests on the behaviour of the design alternatives were made to guide the final controller selection by means of a multi-objective PSO algorithm. The main objective of this work is to focus on the EMO stage by evaluating the best PSO strategy based on a number of non-dominated solutions criteria. Several design alternatives are evaluated to analyse the trade-off achieved by each MOEA alternative. A work like this could be used as reference to select the most adequate evolutionary strategy (regarding its exploitation and exploration capabilities) for a given MOP.

In [90] a fuzzy controller is tuned by optimising a quality function and the number of rules in the fuzzy logic controller. The example provided is a heat, ventilation, and air conditioning (HVAC) system with a fuzzy controller with nine inputs and three outputs. It is a very complex process given the considerable time requirements and computation time needed to run a simulation to measure the performance. The quality function used comprises an aggregate objective function $J_{AFO}(\Lambda, \overline{\overline{\Upsilon}})$ of thermal comfort, air quality requirements, energy consumption, and system stability. The aim of this work was to obtain a controller to accomplish the quality function cost with a reduced (tractable) number of rules and therefore $J(\Lambda, \overline{\overline{\Upsilon}}) = \overline{\overline{\Upsilon}}$ is considered as second objective. A design concept analysis (2D visualisation) is presented at two different levels: among different MOEAs and among controllers. The advantages of considering the number of rules was demonstrated since it was possible to reduce them, and still fulfilling the requirements with better results in the cost function.

In [91], a fuzzy logic controller for drug infusion for anesthesia was proposed. Traditionally, this administration is performed by an expert; therefore, a fuzzy controller is a good candidate to incorporate such expertise into an automatic control scheme. Nevertheless, when trying to optimise the performance using membership functions and rule structure as decision variables, the controller's interpretability could be lost. This is a non-desirable situation, given that for the medical staff it is fundamental to understand the logic behind the controller for the right anesthesia infusion. Due to this, two objectives were stated: control action quality by means of and aggregate objective function $J_{AFO}(\Lambda, \Upsilon)$ based on $J_{ISE}(\Lambda, \Upsilon)$, $J_{ISU}(\Lambda, \Upsilon)$ and an interpretability index $J_{II}(\Lambda, \Upsilon)$. In the MCDM step, several design concepts are compared with an alternative solution that is closer to the ideal solution in the calculated Pareto front.

Finally, in [92], fuzzy logic controllers are adjusted on-line using MOEAs. The main purpose of this work was to investigate the applicability of MOEA for control design under a hardware-in-the-loop context. The selected process was a sealed pump running on magnetic bearings. Four objectives were minimized: rise-time, steady-state error, power utilisation, and control complexity, using as decision variables $\boldsymbol{\theta} = (\overline{\overline{\Upsilon}}, \overline{\Lambda}, \Upsilon)$. Whereas several controllers fulfill the required specifications, those with the least complexity were selected. As noticed by the authors, lower controller complexity offers several computational advantages when working on-line.

3.2.2. Conclusions on fuzzy controller design concept

In table 2 a summary on these applications is provided. It is possible to notice the difference on the quantity of works dedicated to fuzzy controllers with the ones dedicated to PID controllers.

In regard to MOP definition, it seems that EMO has been popular to optimise simultaneously objectives related with performance and interpretability of the fuzzy inference system. Also, it can be noticed that few works incorporate constraints into the MOP statement; in the same way, is not usual to find many-objectives optimisations instances. This could be due to the fact that such kind of design concepts needs to handle with large scale optimisation instances (several decision variables). This is justifiable if we considerer that the first

Table 2: Summary of MOOD methodology for Fuzzy Controller design concept. $\boldsymbol{J(\theta)}$ refers to the number of objectives; $\boldsymbol{\theta}$ to the number of decision variables and $\boldsymbol{g(\theta)}$, $\boldsymbol{h(\theta)}$ to the number of inequality and equality constraints, respectively (according with examples provided in each paper). In some instances, constraints are also stated as objectives.

			MOP			EMO		MCDM	
Concept(s)	Process(es)	Ref	$\boldsymbol{J(\theta)}$	$\boldsymbol{\theta}$	$(\boldsymbol{g(\theta)}, \boldsymbol{h(\theta)})$	Algorithm	Related features from section 3	Plot	Selection insights
Fuzzy scheduling	Aeronautical	[88]	9	100	(9, 0)	GA	2.2.3,2.2.5	PAc	Constraint violation analysis; fine tuning.
PID, Fuzzy Controller	DC motor (HiL)	[92]	4	34	(0,0)	GA	2.2.6	None	According performance.
Fuzzy controller	Geological	[87]	4	160	(0,0)	NSGA-II	2.2.10	SCp	Design alternatives comparison.
Fuzzy controller	Bio-medical	[91]	2	40	(0,0)	SPEA based	2.2.10	2D	Design alternatives/concepts comparison with other controllers. Selection by norm-2 criteria.
Fuzzy controller	Mechanical	[89]	3	30	(0, 0)	PSO		3D	Design alternatives comparison.
Fuzzy controller	HVAC system	[90]	2	3096	(0, 0)	SPEA based	2.2.10,2.2.9	2D	Design alternatives comparison at two levels: different controllers and different MOEAs.

issue to handle are the parameters of the controller before dealing with additional objectives and/or constraints. The reason because constraints are not fully incorporated, could be for incompatibilities with the large-scale optimisation and the constraints handling; therefore the question here is: *is there a gap on MOEAs to satisfy such need?* This scalability issue is an open problem also noticed in [56]. Finally, in the MCDM step SCp tools have been enough for Pareto front visualization and analysis, due to the few objectives stated in the MOP; nevertheless, as more objectives are included in the MOP, some mechanism to deal with scalability and many-objectives optimisation will be required, with integration of decision maker preferences, as stated in the same review.

3.3. State space feedback controller design concept

State space representation has shown to be a remarkable tool for controller design. Several advanced control techniques use this representation to calculate a controller (in the same representation) with a desired performance. The decision variables stated are the gains of the matrix $\boldsymbol{G}$. Classical optimisation approaches in a MOOD framework has been used [93] with good results. In several instances, it seems that the MOOD procedure has been used

to compare classical approaches with the one provided by the EMO approach, as presented below.

3.3.1. Applications on state space feedback controller design concept

In [94], a new algorithm is developed for state space feedback controller tuning. The algorithm is based on an ϵ-elimination procedure, which eliminates similar individuals in the objective space sense, and in the decision variable space. This could be helpful to improve the diversity of solutions when the decision vector could be crucial in discriminating or accepting solutions. The algorithm is evaluated in a single inverted pendulum model using two objectives in two different instances; in the first one, an aggregate function objective of the settling time and overshoot of the cart $J_{AFO_1}(\boldsymbol{G})$ and pendulum $J_{AFO_2}(\boldsymbol{G})$ were stated. In the second the probabilities of failure $J_{Pr_1}(\boldsymbol{G})$, $J_{Pr_2}(\boldsymbol{G})$ measured in a probabilistic set of models were used as objectives. As two objectives were stated for each instance, a classical SCp visualisation and decision analysis were performed.

In [95], the algorithm developed in [66] (covered in the PID design concept) is used for n-order controller tuning for the SISO and MIMO processes. Initially, the algorithm's capacity to calculate controllers under an H_2/H_∞ framework was compared with LMI techniques using a 2D representation. One of the advantages of the MOOD approach is flexibility in fixing the controller order size. Several controllers with different order sizes (to define different design concepts) were proposed for MIMO controller tuning. This last statement enabled the performance improvement of the controller to be analysed as the controller order size variate. Similarly, in [96], the MOOD approach is compared with the LMI concept for an LQR and H_2/H_∞ controllers using as objectives noise sensitivities for output and actuator.

3.3.2. Conclusions on state space feedback controller design concept

In table 3 a summary on these applications is provided. There are still few works focusing in this design concept and therefore, it is difficult to extrapolate conclusions as in the PID and fuzzy cases. It can be noticed that the MOOD procedure has been compared with modern tuning techniques as H_2/H_∞ and LQC techniques. The stochastic sampling used by evolutionary techniques to search the gains $\boldsymbol{G}$ on the control matrix could lead to the same issues noticed in PID control (section 3.1.2). Given that H_2/H_∞ and LQC techniques are also based on optimization, it could be interesting to hybridize both approaches for this design concept.

3.4. Predictive control design concept

Online applications for MOOD are not straightforward, since the MCDM stage must be carried out, in some instances, automatically. As a result, analysis that relies on the DM must be codified to become an automatic process. Predictive control techniques have been incorporating the MOOD framework and the MCDM procedure in their optimisation stages. Good examples using deterministic approaches are presented in [97, 98, 99]. Approaches using EMO in the MOOD procedure are presented below; decision variables regard to $\boldsymbol{\theta} = \boldsymbol{U}$ (control action through the control horizon) or $\boldsymbol{\theta} = \boldsymbol{R}$ (references given to the controllers through the control horizon).

Table 3: Summary of MOOD methodology for state space feedback controller design concept. $\boldsymbol{J(\theta)}$ refers to the number of objectives; $\boldsymbol{\theta}$ to the number of decision variables and $\boldsymbol{g(\theta)}$, $\boldsymbol{h(\theta)}$ to the number of inequality and equality constraints, respectively (according with the examples provided in the papers). In some instances, constraints are also counted as objectives.

			MOP			EMO		MCDM	
Concept(s)	Process(es)	Ref	$\boldsymbol{J(\theta)}$	$\boldsymbol{\theta}$	$(\boldsymbol{g(\theta)}, \boldsymbol{h(\theta)})$	Algorithm	Related features from section 3	Plot	Selection insights
H_2/H_∞	SISO, MIMO	[95]	3	13	(1,0)	GA	2.2.4	SCp	Concepts comparison with LMI design
LQG, H_2/H_∞	SISO	[96]	3	6	(1,0)	GA	2.2.4, 2.2.3	2D	Concepts comparison with LMI.
State space controller	Mechanical	[94]	4	4	(0,0)	GA	2.2.7, 2.2.8	SCp	Design alternatives comparison

3.4.1. Applications on predictive control design concept

Recently, [100] proposed a MOOD procedure, where an evolutionary approach and an inference system are used for non-linear systems. Firstly, a NARX dynamic neural network is used to built a model of the plant. Then a multi-objective genetic algorithm is used to approximate a Pareto front on the selected prediction horizon $J_{ISE}(\boldsymbol{U})$; afterwards a fuzzy inference system is used to select the control action to be applied. This is an interesting example where several computational intelligence tools are used for modeling, optimizing and selecting the most suitable control action.

In [101] NSGA-II is used to generate the Pareto set of optimal trajectories $\theta = \boldsymbol{R}$ for set-point changes for supervisory control of flotation columns. The MOOD is used to meet the engineering requirements, as well as the market requirements that are specifically stated for the control problem. Stated objectives include reducing the change of the references upon the control horizon, economic profit and hydraulic stability, subject to constraints related with froth overloading, quality of the concentrate and cleaning efficiency. After the optimisation, an automatic decision making procedure, based on normalised distances to the ideal solution and constraint fulfillment, is defined to select the best trajectory.

3.4.2. Conclusions on predictive control design concept

In table 4 a summary on these applications is provided. As it can be noticed, in Predictive control seem to be an opportunity to apply the MOOD approach, due to the few works dedicated to this control design alternative. Nevertheless, as it can be noticed also, the problem relies in, besides tracking the Pareto front each sampling time, to perform the selection procedure *on the fly* (interactive approach). An alternative, not exploited here, is to use the MOOD approach in an *upper* layer where the DM may analyse the Pareto set approximation and change the control objectives accordingly (as employed in [99] with multiobjective deterministic algorithms).

Table 4: Summary of MOOD methodology for predictive control concept. $J(\boldsymbol{\theta})$ refers to the number of objectives; $\boldsymbol{\theta}$ to the number of decision variables and $g(\boldsymbol{\theta})$, $h(\boldsymbol{\theta})$ to the number of inequality and equality constraints, respectively (according with the examples provided in the papers). In some instances, constraints are also counted as objectives.

			MOP			EMO		MCDM	
Concept(s)	Process(es)	Ref	$J(\boldsymbol{\theta})$	$\boldsymbol{\theta}$	$(g(\boldsymbol{\theta}), h(\boldsymbol{\theta}))$	Algorithm	Related features from section 3	Plot	Selection insights
Predictive control	Mechanical	[100]	2	8	(0,0)	GA	2.2.6	None	Fuzzy inference system is used.
Predictive control	Chemical	[101]	8	-	(4,0)	NSGA-II	2.2.6; 2.2.4	None	Successive ordering according to feasibility.

3.5. Other design concepts approaches

The MOOD methodology has also been used with ad-hoc controller structures to address the performance of particular and complex processes. That is, specific controllers structures for specific processes. They are commented below.

3.5.1. Applications

In [102] a disk drive servo controller was tuned using an MOEA. An ad-hoc algorithm was proposed that assigns greater priority to constraint objectives than optimisation objectives. These constraints are bounds on certain design objectives that avoid non-pertinent solutions. A goal objective vector is defined with a prioritised ranking approach, for two different tracks that seek values where constraints (3) are included as objectives (for a total of 10 objectives). Objectives are based on $J_{ISU}(\boldsymbol{\theta})$ and $J_{IADU}(\boldsymbol{\theta})$ constrained to desired values of steady state error, standard deviation of following error and seek time. To deal with the many objective optimization instance, a preference articulation is included in the algorithm to focus the search towards a pertinent Pareto front. In this work, the designer identifies the importance of having a solution reference (goal vector) and uses it in the evolution process to improve the pertinency of solutions. Pareto front visualisation and analysis were performed using a PAc visualisation. After a validation in a simulated model, the design alternatives selected are evaluated in the physical process.

In [103] a pole placement controller for an electronically operated throttle system is adjusted using a MOOD methodology. The process is a highly non-linear system, with several time domain specifications and requirements to fulfil. Firstly, a low-order model is used to identify the tractable number of poles in the controller (nine in this case). A high-order non-linear model (experimentally verified) is then employed to select the pole location using a MOEA. A total of five objectives are defined: rise time, overshoot, settling time, steady-state error, and system delay. The final selection was used by the designer in accordance with his expertise. It was remarked that the extra design work was worthwhile since a controller with a better response lag was calculated.

In [104] a MOGA algorithm is used to generate *command shapers* for the control of a flexible manipulator; this kind of manipulator is light and fast but vibration control is difficult to achieve. A command shaper control is a good alternative for control, but several design objectives could be in conflict. For this reason, the MOOD approach is used to generate a set of potentially preferable controllers. Six objectives are defined and a PAc visualisation is used to compare design alternatives. Objectives are based on settling time (for hub and end-point), rise time, peak to peak oscillation, infinity norm and RMS value of end-point acceleration. The main advantage identified by the authors is that the MOEA does not require a priori knowledge of the natural frequency to calculate the command shaper parameters, as required by the original design methodology. The methodology process includes a preliminary analysis of three different command shapers (design concepts) in 2D; the most promising concept is then used to deal with a six objective space.

In [105], a design concept comparison for different controllers was performed using the MOOD approach. A MOEA was used to calculate the best control action that cancels vibrations in a magneto-rheological damper (MRD). Since this is not possible (in the practical sense) because it requires a perfect knowledge of the current state of the damper, such control action and its performance were used to bound the performance for a set of controllers. A Skyhook control, feedback linearisation, and a sliding mode control were compared using this framework and two objectives were used: mean dissipated power and absorbed power. A design concept comparison was commented on that led to the selection of the best control solution according to the current state of the MRD. This is a good example of how a design concept comparison can assist the DM select a desired controller according to his/her preferences.

Finally, in [106], a MOEA is used to tune the parameters of a fractional PID controller $(\text{PI}^\lambda\text{D}^\nu)$ [107] using a reliability based optimization approach with stochastic sampling. This type of controller has some advantages over the classical PID controllers, but tuning techniques are more difficult. The authors chose a MOOD approach to select the parameters of this type of controller in order to fully appreciate its performance. A design concept comparison with a classical PID controller was made using an LD visualisation framework with five design objectives. Objectives stated include the probabilities of failure of $J_S(\theta_{PI^\lambda D^\nu})$, $J_T(\theta_{PI^\lambda D^\nu})$, $J_{ITSE}(\theta_{PI^\lambda D^\nu})$ and $J_{U_{max}}(\theta_{PI^\lambda D^\nu})$ (equations 5, 7, 12 and 15) mixing time domain and frequency domain objectives. Using such an approach, it was possible to appreciate the drawbacks and advantages of using a complex PID controller. As mentioned previously, this could be useful for the designer in justifying his/her selection.

3.5.2. Conclusions

In table 5 a summary on these applications is provided. An interesting fact is that GA seems to be the first option as a evolutionary technique for ad-hoc controllers. Also it seems that the option with specific controllers is to evaluate their performance in many-objectives optimisation instances.

Table 5: Summary of MOOD methodology for control systems engineering. $J(\boldsymbol{\theta})$ refers to the number of objectives; $\boldsymbol{\theta}$ to the number of decision variables and $g(\boldsymbol{\theta})$, $h(\boldsymbol{\theta})$ to the number of inequality and equality constraints, respectively (according with the examples provided in the papers). In some instances, constraints are also counted as objectives.

			MOP			EMO		MCDM	
Concept(s)	Process(es)	Ref	$J(\boldsymbol{\theta})$	$\boldsymbol{\theta}$	$(g(\boldsymbol{\theta}), h(\boldsymbol{\theta}))$	Algorithm	Related features from section 3	Plot	Selection insights
State space controller	Electronic	[102]	10	7	(6,0)	GA	2.2.4, 2.2.5,2.2.3	PAc	Design alternatives comparison
RST controller	Mechanical	[103]	5	9	(0,0)	GA	2.2.5	None	Design alternative evaluation
Command Shapers	Robotics	[104]	2,6	8	(0,0)	GA	2.2.5	PAc	Concepts comparison in 2D. Selection of a flexible controller for 6D objectives.
Several	Mechanical	[105]	2	–	(0,0)	GA		2D	Concepts comparison.
PID, PI$^\lambda$D$^\mu$	SISO	[106]	5	5	(0,0)	NSGA-II	2.2.5; 2.2.8	LD	Design alternatives comparison.

4. Conclusions

A review of MOOD methodologies in control systems engineering field was presented. The MOOD procedure is a GFCL approach that includes a MOP statement, a multi-objective optimisation process, and an MCDM step. All of them are important for a successful implementation of the MOOD approach and embedment of the DM into the design process. This review has focused on EMO techniques for the optimisation stages for MOPs dealing with controller tuning.

The MOOD procedure has been shown to be a useful tool for parametric controller tuning. Such approach allows the designer to have a different insight on design alternatives and their trade-off, in order to select the most convenient or preferable solution for the DM. The MOOD procedure requires a closest embedment of the designer and it is more time consuming than an AOF approach, due to the multi-objective optimisation stage and the MCDM step. For this reason, this approach could be reserved for complex MOP instances, where it is worthwhile to expend more time in the design phase to analyze the objective exchange among design alternatives.

Several applications on controller tuning for different design concepts have been presented and discussed. The MOOD procedure has been used for different kind on controllers, from simple to complex architectures in a wide variety of applications. Next, some possible trends will be commented, attending to each one of the steps of the MOOD procedure.

Table 6: Summary of EMO features on works presented

Concept(s)	Ref	Pertinency	Constrained	Many Objectives	Dynamic	Multi-modal	Robust	Expensive	Large Scale
PID Control	[66]		X						
	[67]								
	[82]	X	X	X					
	[68]	X	X						
	[69]		X						
	[79]		X	X					
	[71]		X	X					
	[77]		X	X					
	[75]								
	[76]		X						
	[80]		X						
	[72]		X	X					
Fuzzy Control	[88]	X	X	X					
	[92]				X				
	[87]								X
	[91]								X
	[89]								
	[90]							X	X
Space	[95]		X						X
	[96]	X	X						
	[94]					X	X		
Predictive Control	[100]				X				
	[101]		X		X				
State	[102]	X	X	X					
Others	[104]	X		X					
	[103]			X					
	[105]								
	[106]			X			X		

4.1. The multi-objective problem statement

Perhaps the first question to answer regarding the MOP is: *What kind of problems are worthwhile to address with MOOD?* As noticed in [108], more activity should be focused on identifying which problems are really *real world problems* affordable by MOEAs. There is no doubt that controller tuning is a real problem with practical applications. Nevertheless, for the same reason there are several techniques for controller tuning. Therefore, some research should be focus on identifying what kind of MOP are best suitable to be solved by the MOOD procedure in controller tuning. The following two questions could be helpful:

- *Is it difficult to find a controller with a reasonable balance among design objectives?*

28

- *Is it worthwhile to analyse the trade-off among controllers (design alternatives)?*

If the answer is yes to both of them, then the MOOD procedure could be an appropriate tool for the problem at hand. Otherwise, tuning techniques or AFO approaches could be enough.

For complex controllers (Fuzzy for instance), the MOOD could be a valuable alternative. But in the specific case of PID controller, we have noticed that several works have focused on tuning a single loop. In such case, the MOOD has to compete with several and well established tuning procedures and techniques. Perhaps efforts should be focused in MIMO processes, that could be more complex to tune and therefore, to find a solution with a reasonable trade-off.

An alternative to the MOOD procedure is to be used at the beginning of the design phase. The multi-disciplinary optimization approach, where mechanical and control design could be merged, is recognized as a mathematical challenge in optimization [109] and a promising optimization approach for design [110]. This integration is exploited by commercial products as modeFRONTIER[3] which could bring an interesting possibility for holistic analyses in multi-disciplinary design.

According with table 6, some gaps among controller design concepts and MOEA's are detected. Such gaps could be due to:

- it has not been proposed a MOP where such desirable characteristics could be required;

- the available algorithms don't reach the specifications required by the designer to provide useful Pareto front approximations.

Regarding the former possibility, if no MOP are defining requiring such features, this is because such MOPs: *are not interesting from the point of view of the designer?* or because *is the designer unaware of the possibility to deal with them using a EMO approach?* In regard to the latter possibility, if MOEAs are not good enough, *is it due to a lack of understanding of the MOP related with controller tuning?* In any case, this questions could provide a starting point for new MOP statements.

Regarding the objectives definition, frequency and time domain performance measures have been used to identify preferable solutions. Nevertheless, few works use both kind of objectives in the same MOP statement. Therefore, it could be a possible direction to merge both, improving pertinency of solutions.

4.2. The evolutionary multi-objective process

With regard to the optimiser, its selection should be made according to the MOP statement in order to guarantee the designer's desired performance. Such selection should be made according to the different features that could be desired for the algorithm. Those characteristics could be related to the quality of the Pareto front (convergence, diversity,

---

[3]http://www.esteco.com/modefrontier

pertinency) or to the MOP stated (constrained, multi-modal, many-objectives, computationally expensive, dynamic or robust optimisation for example).

As noticed in [111], there are several works focusing on *new* evolutionary or bio-inspired techniques instead of focusing in other more *practical* aspects of the optimization problem. The question is *do we need so many MOEA's to address the controller tuning problem?*

Several MOEA's are available for the designer offering such capabilities, and research on new algorithms for controller tuning should be oriented to amend any existing gap (according Table 6) identified by the designer's requirements. Research should be focus on mechanisms to support (if required) the different optimization instances from section 2.2.1 to 2.2.10, instead of defining new evolutionary techniques. Given that convergence and diversity are expected properties, and almost every MOEAs include mechanisms for those purposes, the efforts could be oriented to preferences inclusion. As commented before, we could use the MOOD in the controller tuning framework for those problems where it is difficult to find a reasonable trade-off. That means that the designer has some idea concerning what does he/she needs, and this information could be merged into the evolutionary process. This could be a valuable mechanism to lead the evolutionary search efficiently towards a pertinent Pareto front approximation. This could facilitate the optimisation in many-objectives and large-scale optimization instances.

Also, the following questions should be answered: *which evolutionary techniques fits better for a given design concept?* That is, *which kind of exploitation and exploration capabilities are better for approximating a Pareto front? What operators fits better on the objective space, to approximate a Pareto font given a controller? Which problems are most suitable to be addressed by a particular evolutionary technique? Is there any difference? how to address problem which are separable, non separable or a mixture of both?* One possibility to answer those questions is to define a proper control benchmark, suitable to be solved by using the MOOD procedure. Although there are several works using this procedure for controller tuning, a common benchmark problem has not been stated to compare performance on MOEA's and decision making procedures. The Alstom gasifier could represent an excellent benchmark platform for this purpose.

Finally, more basic research would be helpful, as the one required to guarantee the two characteristics for stochastic sampling referred in section 3.1.2 as well as work focusing on optimization hybrid techniques.

4.3. The multi-criteria decision making step

Concerning the MCDM step several methodologies and visualizations have been presented that could help the DM take his/her final resolution. Nevertheless, more insights on the MCDM procedure needs to be documented, since it is not always an easy task to perform. This gap could be amended by bringing more tools from the multi-dimensional data and decision analysis field.

In the visualization case, *could it be possible to define different visualization approaches for analysing multidimensional data and multi-dimensional Pareto front which fits better for controller tuning?* There are several GUI developed for algorithms tuning, but merging such

capabilities with an analysis of Pareto front (minding simplicity, persistence and completeness) could be useful for the designer. Also and interesting feature to develop would be design concepts comparison of different controller in multi-dimensional Pareto fronts. Such analysis could bring conclusions about when it is worthwhile to use a complicated control technique over a simpler one.

Finally, there is an uncommented issue regarding the quantity of Pareto optimal solutions that a MOEA needs to approximate for the MCDM stage. Several algorithms evaluate their performance by approximating a very dense Pareto front approximation with several solutions. Nevertheless, it is needed to mind that the designer will analyse the trade-off of those solutions, and a big quantity of solutions could be more confusing than helpful. Therefore, MOEA's should focus on bringing the quantity of solutions required with the highest pertinency possible.

Acknowledgment

References

[1] K. M. Miettinen, Nonlinear multiobjective optimization, Kluwer Academic Publishers, 1998.

[2] C. A. Mattson, A. Messac, Pareto frontier based concept selection under uncertainty, with visualization, Optimization and Engineering 6 (2005) 85–115, 10.1023/B:OPTE.0000048538.35456.45.

[3] R. Marler, J. Arora, Survey of multi-objective optimization methods for engineering, Structural and Multidisciplinary Optimization (26) (2004) 369 – 395.

[4] P. Bonissone, R. Subbu, J. Lizzi, Multicriteria decision making (mcdm): a framework for research and applications, IEEE Computational Intelligence Magazine.

[5] C. A. C. Coello, D. V. Veldhuizen, G. Lamont, Evolutionary algorithms for solving multi-objective problems, Kluwer Academic press, 2002.

[6] C. A. C. Coello, G. B. Lamont, Applications of Multi-Objective evolutionary algorithms, advances in natural computation vol. 1 Edition, World scientific publishing, 2004.

[7] P. Fleming, R. Purshouse, Evolutionary algorithms in control systems engineering: a survey, Control Engineering Practice (10) (2002) 1223 – 1241.

[8] K. Saridakis, A. Dentsoras, Soft computing in engineering design - a review, Advanced Engineering Informatics 22 (2) (2008) 202 – 221, network methods in engineering. `doi:10.1016/j.aei.2007.10.001`.

[9] G. Reynoso-Meza, X. Blasco, J. Sanchis, Optimización evolutiva multi-objetivo y selección multi-criterio para la ingeniería de control, in: X Simposio CEA de Ingeniería de Control, 2012.

[10] C. A. C. Coello, G. B. Lamont, D. A. V. Veldhuizen, Multi-criteria decision making, in: Evolutionary Algorithms for Solving Multi-Objective Problems, Genetic and Evolutionary Computation Series, Springer US, 2007, pp. 515–545. `doi:10.1007/978-0-387-36797-2\_9`.

[11] C. Fonseca, P. Fleming, Multiobjective optimization and multiple constraint handling with evolutionary algorithms-II: Application example, IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans.

[12] A. Zhou, B.-Y. Qu, H. Li, S.-Z. Zhao, P. N. Suganthan, Q. Zhang, Multiobjective evolutionary algorithms: A survey of the state of the art, Swarm and Evolutionary Computation 1 (1) (2011) 32 – 49. `doi:DOI:10.1016/j.swevo.2011.03.001`.

[13] C. Coello Coello, Evolutionary multi-objective optimization: A historical view of the field, IEEE Computational Intelligence Magazine.

[14] M. Srinivas, L. Patnaik, Genetic algorithms: a survey, Computer.

[15] A. Konak, D. W. Coit, A. E. Smith, Multi-objective optimization using genetic algorithms: A tutorial, Reliability Engineering & System Safety 91 (9) (2006) 992 – 1007, special Issue - Genetic Algorithms and Reliability. `doi:DOI:10.1016/j.ress.2005.11.018`.

[16] J. Kennedy, R. Eberhart, Particle swarm optimization, in: Neural Networks, 1995. Proceedings., IEEE International Conference on, 1995.

[17] C. Coello, An introduction to multi-objective particle swarm optimizers, in: A. Gaspar-Cunha, R. Takahashi, G. Schaefer, L. Costa (Eds.), Soft Computing in Industrial Applications, Vol. 96 of Advances in Intelligent and Soft Computing, Springer Berlin / Heidelberg, 2011, pp. 3–12, $10.1007/978 - 3 - 642 - 20505 - 7_1$.

[18] R. Storn, K. Price, Differential evolution: A simple and efficient heuristic for global optimization over continuous spaces, Journal of Global Optimization 11 (1997) 341 – 359.

[19] E. Mezura-Montes, M. Reyes-Sierra, C. Coello, Multi-objective optimization using differential evolution: A survey of the state-of-the-art, Advances in Differential Evolution (SCI 143) (2008) 173 – 196.

[20] S. Das, P. N. Suganthan, Differential evolution: A survey of the state-of-the-art, IEEE Transactions on Evolutionary Computation PP (99) (2010) 1 –28. `doi:10.1109/TEVC.2010.2059031`.

[21] D. Karaboga, B. Gorkemli, C. Ozturk, N. Karaboga, A comprehensive survey: artificial bee colony (ABC) algorithm and applications, Artificial Intelligence Review (2012) 1–37`doi:10.1007/s10462-012-9328-0`.

[22] M. Dorigo, T. Stützle, Ant colony optimization: Overview and recent advances, in: M. Gendreau, J.-Y. Potvin (Eds.), Handbook of Metaheuristics, Vol. 146 of International Series in Operations Research & Management Science, Springer US, 2010, pp. 227–263. `doi:10.1007/978-1-4419-1665-5\_8`.

[23] K. Deb, Advances in evolutionary multi-objective optimization, in: G. Fraser, J. Teixeira de Souza (Eds.), Search Based Software Engineering, Vol. 7515 of Lecture Notes in Computer Science, Springer Berlin / Heidelberg, 2012, pp. 1–26, 10.1007/978-3-642-33119-0_1.

[24] A. Eiben, C. Schippers, On evolutionary exploration and exploitation, Fundamenta Informaticae 35 (1) (1998) 35 – 50. `doi:10.3233/FI-1998-35123403`.

[25] J. Herrero, M. Martínez, J. Sanchis, X. Blasco, Well-distributed Pareto front by using the ϵ-MOGA evolutionary algorithm., in: F. S. et al. (Ed.), Computational and Ambient Intelligence, Vol. LNCS 4507, Springer-Verlag, 2007, pp. 292 – 299.

[26] A. G. Hernández-Díaz, L. V. Santana-Quintero, C. A. C. Coello, J. Molina, Pareto-adaptive ϵ-dominance, Evolutionary Computation (4) (2007) 493 – 517.

[27] W. Gong, Z. Cai, L. Zhu, An efficient multiobjective Differential Evolution algorithm for engineering design, Structural and Multidisciplinary Optimization 38 (2009) 137 – 157, 10.1007/s00158-008-0269-9.

[28] J. Branke, H. Schmeck, K. Deb, M. Reddy S, Parallelizing multi-objective evolutionary algorithms: cone separation, in: Evolutionary Computation, 2004. CEC2004. Congress on, 2004.

[29] G. Reynoso-Meza, J. Sanchis, X. Blasco, M. Martínez, Multiobjective design of continuous controllers using differential evolution and spherical pruning, in: C. D. Chio, S. Cagnoni, C. Cotta, M. Eber, A. Ekárt, A. I.Esparcia-Alcaráz, C.-K. Goh, J. J.Merelo, F. Neri, M. Preuss, J. Togelius, G. N.Yannakakis (Eds.), Applications of Evolutionary Computation, Part I, Vol. LNCS 6024, Springer-Verlag, 2010, pp. 532–541.

[30] L. Batista, F. Campelo, F. Guimarães, J. Ramírez, Pareto cone ϵ-dominance: Improving convergence and diversity in multiobjective evolutionary algorithms, in: R. Takahashi, K. Deb, E. Wanner, S. Greco (Eds.), Evolutionary Multi-Criterion Optimization, Vol. 6576 of Lecture Notes in Computer Science, Springer Berlin / Heidelberg, 2011, pp. 76–90, 10.1007/978-3-642-19893-9-6.

[31] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, IEEE Transactions on Evolutionary Computation 6 (2) (2002) 124 – 141.

[32] E. Zitzler, S. Künzli, Indicator-based selection in multiobjective search, in: X. Yao, E. Burke, J. Lozano, J. Smith, J. Merelo-Guervós, J. Bullinaria, J. Rowe, P. Tino, A. Kabán, H.-P. Schwefel (Eds.), Parallel Problem Solving from Nature - PPSN VIII, Vol. 3242 of Lecture Notes in Computer Science, Springer Berlin / Heidelberg, 2004, pp. 832–842, 10.1007/978-3-540-30217-9 84.

[33] C. Coello, Handling preferences in evolutionary multiobjective optimization: a survey, in: Evolutionary Computation, 2000. Proceedings of the 2000 Congress on, Vol. 1, 2000, pp. 30 –37 vol.1. `doi:10.1109/CEC.2000.870272`.

[34] D. Cvetkovic, I. Parmee, Preferences and their application in evolutionary multiobjective optimization, IEEE Transactions on Evolutionary Computation.

[35] M. Munro, B. Aouni, Group decision makers' preferences modelling within the goal programming model: An overview and a typology, Journal of Multi-Criteria Decision Analysis 19 (3-4) (2012) 169–184. `doi:10.1002/mcda.492`.

[36] C. Fonseca, P. Fleming, Multiobjective optimization and multiple constraint handling with evolutionary algorithms-I: A unified formulation, IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans.

[37] C. A. C. Coello, Theorical and numerical constraint-handling techniques used with evolutionary algorithms: a survey of the state of the art, Computer methods in applied mechanics and engineering (191) (2002) 1245 – 1287.

[38] E. Mezura-Montes, C. A. C. Coello, Constraint-handling in nature-inspired numerical optimization: Past, present and future, Swarm and Evolutionary Computation 1 (4) (2011) 173 – 194. `doi:10.1016/j.swevo.2011.10.001`.

[39] K. Deb, An efficient constraint handling method for genetic algorithms, Computer Methods in Applied Mechanics and Engineering 186 (2-4) (2000) 311 – 338. `doi:DOI:10.1016/S0045-7825(99)00389-8`.

[40] R. Mallipeddi, P. Suganthan, Problem definitions and evaluation criteria for the CEC 2010 competition on constrained real-parameter optimization, Tech. rep., Nanyang Technological University, Singapore (2009).

[41] G. Reynoso-Meza, X. Blasco, J. Sanchis, M. Martínez, Multiobjective optimization algorithm for solving constrained single objective problems, in: Proceedings of the IEEE congress on evolutionary computation (CEC2010), 2010.

[42] R. Mallipeddi, P. Suganthan, Ensemble of constraint handling techniques, IEEE Transactions on Evolutionary Computation.

[43] R. Purshouse, P. Fleming, On the evolutionary optimization of many conflicting objectives, IEEE Transactions on Evolutionary Computation.

[44] H. Ishibuchi, N. Tsukamoto, Y. Nojima, Evolutionary many-objective optimization: A short review, in: Evolutionary Computation, 2008. CEC 2008. (IEEE World Congress on Computational Intelligence). IEEE Congress on, 2008.

[45] M. Farina, K. Deb, P. Amato, Dynamic multiobjective optimization problems: test cases, approximations, and applications, IEEE Transactions on Evolutionary Computation 8 (5) (2004) 425–442. `doi:10.1109/TEVC.2004.831456`.

[46] C. Cruz, J. González, D. A. Pelta, Optimization in dynamic environments: a survey on problems, methods and measures, Soft Computing 15 (2011) 1427–1448. `doi:10.1007/s00500-010-0681-0`.

[47] S. Das, S. Maity, B.-Y. Qu, P. Suganthan, Real-parameter evolutionary multimodal optimization - a survey of the state-of-the-art, Swarm and Evolutionary Computation 1 (2) (2011) 71 – 88. `doi:10.1016/j.swevo.2011.05.005`.

[48] H.-G. Beyer, B. Sendhoff, Robust optimization - a comprehensive survey, Computer Methods in Applied Mechanics and Engineering 196 (33-34) (2007) 3190 – 3218. `doi:10.1016/j.cma.2007.03.003`.

[49] L. Santana-Quintero, A. Montaño, C. Coello, A review of techniques for handling expensive functions in evolutionary multi-objective optimization, in: Y. Tenne, C.-K. Goh (Eds.), Computational Intelligence in Expensive Optimization Problems, Vol. 2 of Adaptation Learning and Optimization, Springer Berlin Heidelberg, 2010, pp. 29–59. `doi:10.1007/978-3-642-10701-6\_2`.

[50] M. Lozano, D. Molina, F. Herrera, Soft Computing: Special Issue on scalability of evolutionary algorithms and other metaheuristics for large-scale continuous optimization problems., Vol. 15, Springer-Verlag, 2011.

[51] J. Figueira, S. Greco, M. Ehrgott, Multiple criteria decision analysis: State of the art surveys, Springer international series, 2005.

[52] A. Lotov, K. Miettinen, Visualizing the Pareto frontier, in: J. Branke, K. Deb, K. Miettinen, R. Slowinski (Eds.), Multiobjective Optimization, Vol. 5252 of Lecture Notes in Computer Science, Springer Berlin / Heidelberg, 2008, pp. 213–243.

[53] A. Inselberg, The plane with parallel coordinates, The Visual Computer 1 (1985) 69–91.

[54] X. Blasco, J. Herrero, J. Sanchis, M. Martínez, A new graphical visualization of n-dimensional Pareto front for decision-making in multiobjective optimization, Information Sciences 178 (20) (2008) 3908 – 3924.

[55] G. Reynoso-Meza, X. Blasco, J. Sanchis, J. M. Herrero, Comparison of design concepts in multi-criteria decision-making using level diagrams, Information Sciences 221 (2013) 124 – 141. `doi:10.1016/j.ins.2012.09.049`.

[56] M. Fazzolar, R. Alcalá, Y. Nojima, H. Ishibuchi, F. Herrera, A review of the application of multi-objective evolutionary fuzzy systems: Current status and further directions., IEEE Transactions on Fuzzy Systems 45–65`doi:10.1109/TFUZZ.2012.2201338`.

[57] C.-K. Goh, E.-J. Teoh, K. C. Tan, Hybrid multiobjective evolutionary design for artificial neural networks, IEEE Transactions on Neural Networks.

[58] M. Delgado, M. Cuellar, M. Pegalajar, Multiobjective hybrid optimization and training of recurrent neural networks, IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics.

[59] Y. Jin, B. Sendhoff, Pareto-based multiobjective machine learning: An overview and case studies, IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews.

[60] T. Suttorp, C. Igel, Multi-objective optimization of support vector machines, in: Y. Jin (Ed.), Multi-Objective Machine Learning, Vol. 16 of Studies in Computational Intelligence, Springer Berlin / Heidelberg, 2006, pp. 199–220, 10.1007/3-540-33019-4_9.

[61] K. Åström, T. Hägglund, The future of PID control, Control Engineering Practice 9 (11) (2001) 1163 – 1175. `doi:10.1016/S0967-0661(01)00062-4`.

[62] K. J. Åström, T. Hägglund, Advanced PID Control, ISA - The Instrumentation, Systems, and Automation Society, Research Triangle Park, NC 27709, 2005.

[63] R. Vilanova, V. M. Alfaro, Robust PID control: an overview (in spanish), Revista Iberoamericana de Automática e Informática Industrial 8 (3) (2011) 141 – 158.

[64] W. Tan, J. Liu, F. Fang, Y. Chen, Tuning of PID controllers for boiler-turbine units, ISA Transactions 43 (4) (2004) 571 – 583. `doi:10.1016/S0019-0578(07)60169-4`.

[65] G. Stewart, T. Samad, Cross-application perspectives: Application and market requirements, in: T. Samad, A. Annaswamy (Eds.), The Impact of Control Technology, IEEE Control Systems Society, 2011, pp. 95 – 100.

[66] A. Herreros, E. Baeyens, J. R. Perán, Design of PID-type controllers using multiobjective genetic algorithms, ISA Transactions 41 (4) (2002) 457 – 472. `doi:DOI:10.1016/S0019-0578(07)60102-5`.

[67] S. Tavakoli, I. Griffin, P. Fleming, Multi-objective optimization approach to the PI tuning problem, 2007.

[68] L. Huang, N. Wang, J.-H. Zhao, Multiobjective optimization for controller design, Acta Automatica Sinica 34 (4) (2008) 472 – 477. `doi:DOI:10.3724/SP.J.1004.2008.00472`.

[69] M.-H. Hung, L.-S. Shu, S.-J. Ho, S.-F. Hwang, S.-Y. Ho, A novel intelligent multi-objective simulated annealing algorithm for designing robust PID controllers, IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans.

[70] B. Suman, P. Kumar, A survey of simulated annealing as a tool for single and multiobjective optimization, Journal of the Operational Research Society 57 (10) (2005) 1143–1160.

[71] G. Reynoso-Meza, X. Blasco, J. Sanchis, Multi-objective design of PID controllers for the control benchmark 2008-2009 (in spanish), Revista Iberoamericana de Automática e Informática Industrial 6 (4) (2009) 93 – 103.

[72] G. Reynoso-Meza, J. Sanchis, X. Blasco, J. M. Herrero, Multiobjective evolutionary algortihms for multivariable PI controller tuning, Expert Systems with Applications 39 (2012) 7895 – 7907. `doi:10.1016/j.eswa.2012.01.111`.

[73] K. Åström, H. Panagopoulos, T. Hägglund, Design of PI controllers based on non-convex optimization, Automatica 34 (5) (1998) 585 – 601. `doi:DOI:10.1016/S0005-1098(98)00011-9`.

[74] W. L. Luyben, Simple method for tuning SISO controllers in multivariable systems, Industrial and Engineering Chemistry Process Design (25) (1986) 654 – 660.

[75] Sidhartha, Panda, Multi-objective PID controller tuning for a facts-based damping stabilizer using non-dominated sorting genetic algorithm-II, International Journal of Electrical Power and Energy Systems 33 (7) (2011) 1296 – 1308. `doi:10.1016/j.ijepes.2011.06.002`.

[76] S. Algoul, M. Alam, M. Hossain, M. Majumder, Multi-objective optimal chemotherapy control model for cancer treatment, Medical and Biological Engineering and Computing 49 (2011) 51–65, 10.1007/s11517-010-0678-y.

[77] Y. Xue, D. Li, F. Gao, Multi-objective optimization and selection for the PI control of ALSTOM gasifier problem, Control Engineering Practice 18 (1) (2010) 67 – 76. `doi:DOI:10.1016/j.conengprac.2009.09.004`.

[78] R. Dixon, A. Pike, ALSTOM benchmark challenge II on gasifier control, IEE Proceedings Control Theory and Applications.

[79] J. S. Rao, R. Tiwari, Design optimization of double-acting hybrid magnetic thrust bearings with control integration using multi-objective evolutionary algorithms, Mechatronics 19 (6) (2009) 945 – 964. doi:10.1016/j.mechatronics.2009.06.011.

[80] S.-Z. Zhao, M. W. Iruthayarajan, S. Baskar, P. Suganthan, Multi-objective robust PID controller tuning using two lbests multi-objective particle swarm optimization, Information Sciences 181 (16) (2011) 3323 – 3335. doi:DOI:10.1016/j.ins.2011.04.003.

[81] P. Stewart, G. Jewell, R. Clark, P. Fleming, Controllability analysis of multi objective control systems, in: Computer Aided Control System Design, 2002. Proceedings. 2002 IEEE International Symposium on, 2002, pp. 74 – 79. doi:10.1109/CACSD.2002.1036932.

[82] P. Stewart, D. Gladwin, P. Fleming, Multiobjective analysis for the design and control of an electromagnetic valve actuator, Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering 221 (2007) 567 – 577. doi:10.1243/09544070JAUTO439.

[83] H. Panagopoulos, K. Åström, T. Hägglund, Design of PID controllers based on constrained optimisation, Control Theory and Applications, IEE Proceedings -.

[84] A. Isaksson, S. Graebe, Derivative filter is an integral part of PID design, IEE Proceedings on Control Theory and Applications.

[85] G. Reynoso-Meza, X. Blasco, J. Sanchis, M. Martínez, Evolutionary algorithms for PID controller tuning: Current trends and perspectives (in spanish), Revista Iberoamericana de Automática e Informática Industrial 10 (3) (2013) 251 – 268.

[86] G. Feng, A survey on analysis and design of model-based fuzzy control systems, IEEE Transactions on Fuzzy Systems.

[87] H.-S. Kim, P. N. Roschke, Fuzzy control of base-isolation system using multi-objective genetic algorithm, Computer-Aided Civil and Infrastructure Engineering 21 (6) (2006) 436 – 449. doi:10.1111/j.1467-8667.2006.00448.x.

[88] A. Chipperfield, B. Bica, P. Fleming, Fuzzy scheduling control of a gas turbine aero-engine: a multiobjective approach, IEEE Transactions on Industrial Electronics.

[89] M. Marinaki, Y. Marinakis, G. Stavroulakis, Fuzzy control optimized by a multi-objective particle swarm optimization algorithm for vibration suppression of smart structures, Structural and Multidisciplinary Optimization 43 (2011) 29–42, 10.1007/s00158-010-0552-4.

[90] M. Gacto, R. Alcalá, F. Herrera, A multi-objective evolutionary algorithm for an effective tuning of fuzzy logic controllers in heating, ventilating and air conditioning systems, Applied Intelligence 36 (2012) 330–347, 10.1007/s10489-010-0264-x.

[91] p. Fazendeiro, J. de Oliveira, W. Pedrycz, A multiobjective design of a patient and anaesthetist-friendly neuromuscular blockade controller, IEEE Transactions on Biomedical Engineering.

[92] P. Stewart, D. Stone, P. Fleming, Design of robust fuzzy-logic control systems by multi-objective evolutionary methods with hardware in the loop, Engineering Applications of Artificial Intelligence 17 (3) (2004) 275 – 284. `doi:10.1016/j.engappai.2004.03.003`.

[93] F. Meeuse, R. L. Tousain, Closed-loop controllability analysis of process designs: Application to distillation column design, Computers and Chemical Engineering 26 (4-5) (2002) 641 – 647. `doi:10.1016/S0098-1354(01)00791-8`.

[94] A. Jamali, A. Hajiloo, N. Nariman-zadeh, Reliability-based robust Pareto design of linear state feedback controllers using a multi-objective uniform-diversity genetic algorithm (MUGA), Expert Systems with Applications 37 (1) (2010) 401 – 413. `doi:10.1016/j.eswa.2009.05.048`.

[95] A. Herreros, E. Baeyens, J. R. Perán, MRCD: a genetic algorithm for multiobjective robust control design, Engineering Applications of Artificial Intelligence (15) (2002) 285 – 301.

[96] A. Molina-Cristóbal, I. Griffin, P. Fleming, D. Owens, Linear matrix inequialities and evolutionary optimization in multiobjective control, International Journal of Systems Science 37 (8) (2006) 513 – 522.

[97] A. Bemporad, D. M. de la Peña, Multiobjective model predictive control, Automatica 45 (12) (2009) 2823 – 2830. `doi:10.1016/j.automatica.2009.09.032`.

[98] V. M.Zavala, A. Flores-Tlacuahuac, Stability of multiobjective predictive control: An utopia-tracking approach, Automatica 48 (10) (2012) 2627 – 2632.

[99] A. Ramírez-Arias, F. Rodríguez, J. Guzmán, M. Berenguel, Multiobjective hierarchical control architecture for greenhouse crop growth, Automatica 48 (3) (2012) 490 – 498. `doi:10.1016/j.automatica.2012.01.002`.

[100] J. J. V. García, V. G. Garay, E. I. Gordo, F. A. Fano, M. L. Sukia, Intelligent multi-objective nonlinear model predictive control (iMO-NMPC): Towards the on-line optimization of highly complex control problems, Expert Systems with Applications 39 (7) (2012) 6527 – 6540. `doi:10.1016/j.eswa.2011.12.052`.

[101] O. D. Chuk, B. R. Kuchen, Supervisory control of flotation columns using multi-objective optimization, Minerals Engineering 24 (14) (2011) 1545 – 1555. `doi:10.1016/j.mineng.2011.08.003`.

[102] K.-S. Low, T.-S. Wong, A multiobjective genetic algorithm for optimizing the performance of hard disk drive motion control system, IEEE Transactions on Industrial Electronics.

[103] P. Stewart, J. C. Zavala, P. Fleming, Automotive drive by wire controller design by multi-objective techniques, Control Engineering Practice 13 (2) (2005) 257 – 264. `doi:10.1016/j.conengprac.2004.03.010`.

[104] M. Alam, M. Tokhi, Designing feedforward command shapers with multi-objective genetic optimisation for vibration control of a single-link flexible manipulator, Engineering Applications of Artificial Intelligence 21 (2) (2008) 229 – 246. `doi:10.1016/j.engappai.2007.04.008`.

[105] J. H. Crews, M. G. Mattson, G. D. Buckner, Multi-objective control optimization for semi-active vehicle suspensions, Journal of Sound and Vibration 330 (23) (2011) 5502 – 5516. `doi:10.1016/j.jsv.2011.05.036`.

[106] A. Hajiloo, N. Nariman-zadeh, A. Moeini, Pareto optimal robust design of fractional-order PID controllers for systems with probabilistic uncertainties, Mechatronics 22 (6) (2012) 788 – 801. `doi:10.1016/j.mechatronics.2012.04.003`.

[107] C. A. Monje, B. M. Vinagre, V. Feliu, Y. Chen, Tuning and auto-tuning of fractional order controllers for industry applications, Control Engineering Practice 16 (7) (2008) 798 – 812. `doi:10.1016/j.conengprac.2007.08.006`.

[108] Z. Michalewicz, Quo vadis, evolutionary computation?, in: Advances in Computational Intelligence, Springer, 2012, pp. 98–121.

[109] R. Roy, S. Hinduja, R. Teti, Recent advances in engineering design optimisation: Challenges and future trends, CIRP Annals - Manufacturing Technology 57 (2) (2008) 697 – 715. `doi:10.1016/j.cirp.2008.09.007`.

[110] J. R. R. A. Martins, A. B. Lambe, Multidisciplinary design optimization: A survey of architectures, AIAA Journal(In press).

[111] K. Sörensen, Metaheuristics - the metaphor exposed, International Transactions in Operational Research.