Soft Computing manuscript No.
(will be inserted by the editor)

A fuzzy logic controller applied to a diversity-based multi-objective evolutionary algorithm for single-objective optimisation

Eduardo Segredo · Carlos Segura · Coromoto León · Emma Hart

Received: date / Accepted: date

Abstract In recent years, Multi-Objective Evolutionary Algorithms (MOEAs) that consider diversity as an objective have been used to tackle single-objective optimisation problems. The ability to deal with premature convergence has been greatly improved with these schemes. However, they usually increase the number of free parameters that need to be tuned. To improve results and avoid the tedious hand-tuning of algorithms, the use of automated *parameter control* approaches that are able to adapt parameter values during the course of an evolutionary run are becoming more common in the field of Evolutionary Computation (EC). This research focuses on the application of parameter control approaches to diversity-based MOEAs. Two *external* parameter control methods are investigated; a novel method based on Fuzzy Logic and a recently proposed Hyper-heuristic. These are compared to an *internal* control method that uses self-adaptation. An extensive comparison of the three methods is carried out using a set of single-objective benchmark problems of diverse complexity. Analyses include comparisons to a wide-range of schemes with fixed parameters and to a single-objective approach. The results show that the fuzzy logic and hyper-heuristic methods are able to find similar or better solutions than the fixed parameter methods for a significant number of problems, with considerable savings in computational resources and time, whereas the self-adaptive strategy provides little benefit. Finally, we also demonstrate that the controlled diversity-based MOEA outperforms the single-objective scheme in most cases, thus showing the benefits of solving single-objective problems through diversity-based multi-objective schemes.

Keywords Parameter control · Fuzzy logic controllers · Hyper-heuristics · Self-adaptation · Diversity preservation · Benchmark problems

E. Segredo · C. León
Dpto. Ingeniería Informática y de Sistemas
Universidad de La Laguna. Avda. Astrofísico Fco. Sánchez.
Edif. Matemáticas, 38071, Santa Cruz de Tenerife, Spain.
Tel.: +34 922 319 191
Fax: +34 922 318 170
E-mail: esegredo@ull.es

C. Segura
Centro de Investigación en Matemáticas
Área Computación
Callejón Jalisco s/n, Mineral de Valenciana, Guanajuato, Guanajuato 36240, Mexico
E-mail: csegura@ull.es

C. León
E-mail: cleon@ull.es

E. Hart
Institute for Informatics and Digital Innovation.
Edinburgh Napier University, Edinburgh, Scotland, UK.
E-mail: e.hart@napier.ac.uk

1 Introduction

Many real world problems require the application of optimisation strategies. Several exact approaches have been designed to deal with optimisation problems. However, exact methods are generally not affordable for many real world applications, meaning that a wide variety of approximation algorithms have been developed in an effort to obtain good quality solutions in a limited amount of time. Meta-heuristics are a family of approximation techniques that have become popular for solving optimisation problems (Glover and Kochenberger, 2003). They are high-level strategies that guide a set of heuristics in search of an optimum. Among them, Evolutionary Algorithms (EAs) (Eiben and Smith, 2003) are one of the most popular strategies. These draw their

inspiration from biological evolution and are population-based algorithms.

EAs have shown great promise for calculating solutions to large and difficult optimisation problems. However, in some problems, EAs exhibit a tendency to converge towards local optima, with the likelihood of this occurrence depending on the shape of the fitness landscape (Caamaño et al, 2010). Several methods have been designed with the aim of dealing with local optima stagnation. The reader is referred to Črepinšek et al (2013) for an extensive survey of diversity preservation mechanisms. One of the strategies that has acquired some popularity in recent years relies on using multi-objective approaches to solve single-objective optimisation problems (Segura et al, 2013a). Several guidelines for solving single-objective optimisation problems using multi-objective methods have been proposed in the last decades, with diversity-based MOEAs being one of the most promising schemes (Abbass and Deb, 2003). In this type of schemes a set of objectives is calculated for each individual. The first one is the original objective associated with the problem being solved. The remaining objectives—most proposals consider only one additional objective—are measures of the diversity. Some definitions of the auxiliary objectives require specifying additional parameters (Segura et al, 2013b). These parameters have to be tuned in order to improve the solutions obtained. However, Segura et al (2013b) have pointed out that suitable values for these parameters could depend on the problem to be solved and/or on the stage of the optimisation procedure.

In order to define the configuration for an EA, several components and/or parameters, such as the survivor selection mechanism and the variation and parent selection operators, must be specified. In general, the performance of an EA and consequently, the quality of the solutions obtained, are highly dependent on these components and parameters. As a result, it is essential that the parameters of an EA be properly set. This task, however, remains one of the persistent grand challenges in Evolutionary Computation (EC) (Eiben and Smit, 2011).

Parameter setting strategies are commonly divided into two categories: parameter *tuning* and parameter *control*. In parameter *tuning* the objective is to identify the best set of values for the parameters of a given EA, with the EA then being executed using these values, which remain fixed for the complete run. In contrast, the aim of parameter *control* is to design control strategies that select the most suitable values for the parameters at each stage of the search process while the algorithm is executed. In single-objective optimisation, it has been empirically and theoretically demonstrated that different values for the parameters might be optimal at different stages of the optimisation process (Srinivas and Patnaik, 1994; Bäck, 1992). In these cases, it seems more appropriate to apply parameter control strategies that enable the parameter values to adapt or change during the course of an EA run. Therefore, it is natural to apply parameter *control* methods to MOEAs, and particularly, to diversity-based MOEAs.

In this paper we consider novel *parameter control* strategies that can be combined with diversity-based MOEAs, and apply them to a set of well-known single-objective benchmark problems. The parameter control strategies are in charge of controlling the additional parameter added in the auxiliary objective definition. We consider *external* and *internal* methods of parameter control. In relation to the former, we develop a novel method of parameter control based on *Fuzzy Logic*, and compare it to a *Hyper-heuristic* control method proposed by the authors in Segura et al (2010). The external control algorithms are also compared to a number of variations on a method of internal control in which the parameter to be adapted is incorporated into the chromosome used to specify the problem, resulting in *self-adaptation* through evolution.

The aim of this research is not to design a complete state-of-the-art EA for continuous optimisation or to compare our adaptive diversity-based MOEA to other highly efficient EAs or meta-heuristics specifically designed for continuous optimisation, which usually incorporate mechanisms to improve the solutions found. The objective is, on the one hand, to compare our proposed control method based on FLCs against a well-established control scheme based on hyper-heuristics and against self-adaptation, and on the other hand, to show the benefits of parameter control versus parameter tuning. As a result, the contributions of this paper are:

- A novel external parameter control method based on a fuzzy logic controller.
- Novel self-adaptive schemes to control the parameters of a diversity-based MOEA.
- First application of parameter control techniques based on fuzzy logic controllers and self-adaptation to a diversity-based MOEA in which the parameters of the auxiliary objective function are adapted.
- An extensive comparison of external versus internal methods of parameter control for diversity-based MOEAs.
- A comparison of parameter control methods to fixed parameters that highlights the benefit of parameter control as opposed to parameter tuning.
- A comparison between the adaptive diversity-based MOEA and a single-objective EA that shows the ad-

vantages of using diversity-based multi-objective approaches to solve single-objective problems.

The paper is organised as follows. In Section 2, an overview of the state of the art in parameter control in EAs is given. Section 3 provides some background on fuzzy logic controllers and hyper-heuristics, which we propose as parameter control methods. Section 4 exposes the diversity-based MOEA applied herein and covers some background on related schemes. The proposed parameter control methods are explained in Section 5 followed by a detailed analysis of the experimental results in Section 6. Finally, the conclusions and future lines of work are given in Section 7.

2 State of the art of parameter control in evolutionary algorithms

Finding the most suitable configuration of an EA is one of the most challenging tasks in the field of Evolutionary Computation (EC) (Eiben and Smith, 2003). In order to completely define an instance of an EA, two types of information are required (Bartz-Beielstein et al, 2010; Maturana et al, 2009):

- *Symbolic*—also referred to as *qualitative*, *categoric* or *structure* parameters—such as crossover, mutation and selection operators.
- *Numeric*—also referred to as *quantitative* or *behavioural* parameters—such as the population size, the crossover and mutation rates.

For both kinds of parameters, the different elements of the domain are known as parameter values, and a parameter is instantiated by assigning it a value. The main difference between both types of parameters lies in the size of their respective domains. Symbolic parameters, such as the crossover operator, have a finite domain in which order is not established and a distance metric is not defined. In contrast, numeric parameters, such as the mutation rate, have an infinite domain in which a distance metric and an order can be defined for the values. Thus, optimisation and search methods can readily be used to look for the appropriate values of the numeric parameters of an EA. However, in the case of symbolic parameters, as noted above, distance metrics cannot be applied between two values, and therefore optimisation schemes are not able to profit from the definition of these types of metrics for setting such parameters. In this paper we focus on control methods for *numeric* parameters.

The goal of parameter control is to design a control strategy that selects the most suitable parameter values to use at every stage of the search process. The ideas of parameter control were first incorporated in early research into EAs (Davis, 1989; Rechenberg, 1973). Recent research, however, has seen a marked increase in proposals for methods to achieve parameter control in EAs (Lobo et al, 2007). In fact, control methods have been successfully applied to a wide range of EAs, such as *Evolution Strategies* (ES) (Kramer, 2010), *Genetic Algorithms* (GAs) (Fialho, 2010), and *Differential Evolution* (DE) (Qin et al, 2009), among others. In order to classify parameter control approaches, several taxonomies have been proposed. One of the most popular classifications (Eiben et al, 2007) groups the mechanisms according to various criteria. According to the manner in which parameter values are changed, control strategies can be classified as:

- **Deterministic parameter control**. Parameter values are altered by a deterministic rule without using any feedback from the search process.
- **Adaptive parameter control**. Parameter values are updated by a mechanism which uses some feedback from the search process. This mechanism is externally supplied.
- **Self-Adaptive parameter control**. Parameters are encoded into the chromosome and their values are modified by the EA variation operators.

A change can affect a gene, an individual, the whole population, another component, or even the evaluation function. Thus, another classification can be carried out that takes into consideration the scope or level affected by the change.

Finally, we should note that a wide variety of approaches can be found in the literature, though most research on parameter control is focused on the parameters of a 'standard' EA, i.e. the variation operators, like the mutation or the crossover operators, the population size, or combinations of all three (Eiben et al, 2007; Bäck et al, 2000). In this paper we describe the application of control techniques to parameters that adapt the behaviour of the auxiliary objective function in a diversity-based MOEA.

3 Techniques for parameter control: background

In this section we provide background information on two techniques that can be used to implement adaptive parameter control, before describing our novel implementation of both in a later section.

3.1 Fuzzy logic controllers

In recent years, our knowledge of the performance of EAs has significantly increased thanks to the large number of empirical analyses conducted over a wide range of applications in different areas. It would be desirable to profit from this human knowledge by encapsulating it within an algorithm to automate the task of improving the behaviour and the performance of EAs. However, this sort of knowledge is usually incomplete, imprecise, and/or it is not well organised. Consequently, the application of *fuzzy logic-based methods* would seem to offer a promising approach for dealing with this kind of knowledge.

One application of fuzzy logic is the design of *Fuzzy Logic Controllers* (FLCs). FLCs can be used to define control approaches in which the incorporation of human knowledge is performed intuitively. As was stated by Herrera and Lozano (2003), an FLC consists of the *knowledge base*, the *fuzzy inference engine*, and the *fuzzification* and *defuzzification interfaces*. The knowledge base is composed of two different parts, a *data base*, which includes the definitions of the membership functions of the linguistic terms for each input and output variable, and a *rule base* constituted by the collection of fuzzy control rules representing human knowledge.

The main benefit of using FLCs to adapt the parameters of an EA is that the possible values that can be assigned to a certain parameter are infinite, in contrast to other techniques that can only use a finite number of values. However, the main drawback is that FLCs cannot be directly applied to control the symbolic parameters of an EA.

A considerable body of research on FLCs and EAs already exists (Fazzolari et al, 2013; Herrera and Lozano, 2003). For example, different EAs have been used to optimise the design of FLCs for different applications (Fazzolari et al, 2013; Rui et al, 2010; Lau et al, 2009; Herrera, 2008). In this paper, however, we study the reverse of this type of application and focus on the design of FLCs that adapt the parameters of an EA, thus providing an adaptive control technique that utilises feedback from the search process. Several methods have been proposed for controlling the parameters of an EA by using an FLC (Herrera and Lozano, 2003). The main idea is to use an FLC to calculate new parameter values by taking into consideration some combination of performance metrics and current parameter values as the input to the controllers. Some of the best known variants of EAs that use FLCs in order to adapt their parameters include GAs (Varnamkhasti and Lee, 2012; Yao et al, 2012; Liu and Liu, 2011; Im and Lee, 2008; Herrera and Lozano, 2001). Some of these schemes are described in what follows.

In Varnamkhasti and Lee (2012), an FLC is used to control several parameters of a GA, which is employed to solve multi-dimensional knapsack problems. Particularly, the aim is to avoid premature convergence by controlling the population diversity through the adjustment of the crossover and mutation operators, as well as their rates of application. In Yao et al (2012), an FLC is proposed to control the crossover and mutation rates of a GA in order to mitigate the aforementioned convergence issue. Another scheme based on the usage of FLCs was introduced by Im and Lee (2008), where the parameters of the mutation, crossover, and parent selection operators of a GA are adapted. Finally, in Herrera and Lozano (2001) a separate GA is used as an automatic learning mechanism to generate the rule bases belonging to an FLC, which is responsible for adapting the genetic operators of another GA. Thus, both GAs exert an influence on the other, adapting the genetic operators by coevolution.

Some novel variants of EAs, such as DE, have also been combined with FLCs to control their internal parameters. For instance, Liu and Lampinen (2005) proposed two FLCs to adapt the mutation scale factor and the crossover rate of a DE approach.

The feature common to most of the research described in the literature is that the FLCs are designed to adapt the parameters of the mutation or crossover operators, the population size, or combinations of all three (Herrera and Lozano, 2003). In addition, FLCs are usually tailor-made for a specific EA and/or parameters, and they only make use of a unique rule base. The main novelties of the FLC proposed herein are therefore the following:

- The approach is general in that it can be used to adapt different numeric parameters of different EAs.
- The system proposed contains multiple rule bases. A rule base is enabled at a certain time depending on historical information extracted from the optimisation process. This historical data is used to guide the adjustment of the parameter being considered.
- It is the first time that an FLC is used to control the parameters of the auxiliary objective in a diversity-based MOEA.

3.2 Hyper-heuristics

Hyper-heuristics can be defined as search methods or learning mechanisms for *selecting* or *generating* heuristics to solve computational search problems (Burke et al, 2010). Hyper-heuristics based on *heuristic selection* try

to identify and select the most promising heuristics or meta-heuristics—from a set of candidates—to solve a particular instance of a problem. Alternatively, hyper-heuristics based on *heuristic generation* aim to generate heuristics automatically in order to solve a particular instance of a problem. In addition, hyper-heuristics can be further classified as *online* or *offline*, in which the former select or generate heuristics while solving an instance of a problem, and the latter typically learn a mapping between characteristics of a problem during a training phase that can then be applied to new instances. In this paper we consider only online and selection hyper-heuristics. Thus, a hyper-heuristic can be viewed as a method that iteratively chooses between a set of candidate low-level heuristics or meta-heuristics in order to solve an optimisation problem (Burke et al, 2003). The hyper-heuristic learns to carry out choices while solving the optimisation problem at hand.

Hyper-heuristics operate at a higher level of abstraction than traditional heuristics because they have no knowledge of the problem domain. The motivation behind the approach is that, ideally, once a hyper-heuristic is designed, several optimisation problems and/or instances of a problem might be addressed by simply replacing the set of low-level heuristics or meta-heuristics. As a result, the aim of using a hyper-heuristic is to raise the level of generality at which the majority of current heuristic approaches operate (Burke et al, 2003).

Hyper-heuristics are highly correlated to parameter control approaches (Smit and Eiben, 2009). For instance, the candidate low-level approaches might represent different configurations of the same meta-heuristic with variations in the parameters being controlled. The hyper-heuristic would then select the configuration with the most appropriate set of parameters at each point in the search. In fact, hyper-heuristics can be further classified as adaptive parameter control techniques if they receive some kind of feedback from the search process.

Hyper-heuristics are independent of the methods adapted, and therefore they can be designed to control a wide range of approaches. In those cases where the best configuration, in terms of performance, of the same meta-heuristic varies depending on the current stage of the optimisation process, the hyper-heuristics could be used to select the most suitable configuration for each stage. Thus, it seems reasonable to expect the results obtained by the hyper-heuristic to be better than those obtained by any of the candidate low-level configurations executed independently. Furthermore, the use of a hyper-heuristic would permit low-level configurations to have variations in both their numeric and symbolic parameters, thus providing a straightforward mechanism for symbolic parameter control. However, the main drawback of the hyper-heuristic approach is the need to specify the set of candidate low-level configurations. Moreover, since the size of the set of candidate low-level approaches is generally fixed and finite, in the case of controlling numeric parameters, the number of possible values that can be assigned to the numeric parameters is therefore also finite [1]. Despite this, hyper-heuristics have successfully been applied as adaptive parameter control techniques, both to benchmark problems (Ren et al, 2012) and to real world applications (Segura et al, 2013c).

4 Diversity-based multi-objective evolutionary algorithms

Multi-objective methods have been proposed with the aim of optimising several objective functions simultaneously. Moreover, using multi-objective methods to ensure proper diversity when solving single-objective problems is a promising approach. As a result, the use of diversity metrics to define additional objectives might provide a proper balance between exploration and exploitation. For this reason, several studies have analysed the use of MOEAs to promote diversity maintenance in single-objective optimisation. As previously mentioned, these schemes are based on defining a new set of objectives that provide measures of the diversity. Several options have been proposed to define the auxiliary objectives (Greiner et al, 2007; Bui et al, 2005; Toffolo and Benini, 2003). In fact, Segura et al (2013a) gives a taxonomy to classify the different proposals.

In this work, we operate with genotypic measures that consider the values of the genes in order to define the auxiliary objectives. One of the most popular auxiliary objectives was proposed by Toffolo and Benini (2003). Specifically, it is calculated as the mean Euclidean distance in the genotypic space to the remaining individuals in the population and it is called the Average Distance to all Individuals (ADI). Based on these ideas, two new auxiliary objectives were defined by Bui et al (2005). These are the Distance to the Closest Neighbour (DCN) and the Distance to the Best Individual (DBI) functions. Note that all of the above auxiliary objectives have to be maximised.

An extension of the DCN scheme, called DCN-THR, was proposed in Segura et al (2013b). It attempts to limit the survival of very low quality individuals by using a *threshold ratio* $th \in [0, 1]$, which has to be specified by the user. A threshold v is used to penalise individuals

[1] Although a recent publication attempts to address this with a hyper-heuristic that is able to adapt the parameters of the low-level heuristics (Ren et al, 2012).

that have low quality with respect to the original objective function by assigning an auxiliary objective value equal to 0 to said individuals. If $bestObjectiveValue$ is the original objective value of the best individual in the population, and $shift$ is a value that ensures that $bestObjectiveValue - shift \geq 0$ throughout the entire optimisation procedure, then the threshold value v—for a minimisation problem—can be defined as:

$$v = \frac{(bestObjectiveValue - shift)}{th} + shift \quad (1)$$

After v is calculated, all individuals whose original objective value is higher than v—for a minimisation problem—have the value of their auxiliary objective assigned to 0. For the remaining individuals, their auxiliary objective is calculated as the DCN, i.e. the Euclidean distance in the genotypic space to the closest individual. Consequently, individuals that are not able to achieve the specified threshold are penalised. In the special case where $th = 0$, Equation 1 does not hold, and therefore individuals are never penalised. Thus, DCN-THR with $th = 0$ behaves like the DCN approach.

The DCN-THR approach was selected based on previous research by the authors described in Segura et al (2013b), in which it was shown that the incorporation of the threshold ratio th provided significant benefits with respect to those schemes that did not make use of it. However, the main drawback of the approach is that the most appropriate value for this parameter depended on the problem being solved. In addition, it was further suggested by Segura et al (2013b) that the best value of th is also dependent on the stage of the optimisation process, and consequently th should be varied over the course of a run. Therefore, the application of parameter control techniques to automatically adapt this parameter ought to significantly improve both the behaviour and the robustness of the diversity-based MOEA. Let us consider this idea in detail.

In general, any MOEA can be used in combination with the DCN-THR auxiliary objective. There exist a large number of MOEAs described in the literature which have shown good performance (Zhou et al, 2011). The overall results obtained might be greatly improved upon by carefully analysing the performance of different MOEAs in combination with the parameter control strategies described herein. However, such a study is beyond the scope of this research. Instead, and considering the popularity of the Non-dominated Sorting Genetic Algorithm II (NSGA-II) (Deb et al, 2002), we have decided to apply parameter control methods to this algorithm. For a deeper insight into the behaviour of the NSGA-II when combined with the DCN-THR scheme, the reader is referred to Segura et al (2013b).

Algorithm 1 FLC pseudocode

1: **Initialisation**: Generate sample values for the parameter th distributed uniformly in the range $[0, 1]$ considering a certain value Δ as the difference between two consecutive samples
2: **for** (each generated sample value of the parameter th) **do**
3: **Learning**: Execute $numGen$ generations of the diversity-based MOEA with said value for the parameter th in order to gather knowledge
4: **end for**
5: **while** (diversity-based MOEA stopping criterion is not satisfied) **do**
6: **Calculation of input variables**. Set the values for the input variables IMP, VAR, TH-IN, BEST-TH-IN
7: **Selection of the rule base**. Select the most suitable rule base considering the last k decisions carried out by the FLC and the scoring function shown in Equation 3.
8: **Fuzzification**. Transform the crisp values of the input variables to fuzzy sets using the fuzzification interface
9: **Mamdani's fuzzy inference**. Apply the fuzzy operator AND (min), the implication method (min) and the aggregation method (max) using the selected rule base to obtain the fuzzy set of the output variable TH-OUT.
10: **Defuzzification**: Transform the fuzzy set of the output variable TH-OUT to a crisp value Δ_{th} using the defuzzification interface (centroid method)
11: **Parameter update**: $th = th + \Delta_{th}$. The value of th is enclosed in the range $[0, 1]$
12: **Execution**: Execute $numGen$ generations of the diversity-based MOEA with the new value of th
13: **end while**

In order to complete the definition of the diversity-based MOEA some traditional components are used. Particularly, the parent selection mechanism is the well-known *Binary Tournament* (Eiben and Smith, 2003), while the variation operators are the *Uniform Mutation* (UM) (Eiben and Smith, 2003) and the *Simulated Binary Crossover* (SBX) (Deb and Agrawal, 1995). Mutation and crossover operators are applied with probabilities p_m and p_c, respectively.

5 Parameter control methods

In this section we describe in detail the three parameter control approaches that are evaluated in later sections. The first two approaches—fuzzy logic controllers and hyper-heuristics—provide an external control mechanism for altering parameter th during the course of a run. The final approach uses self-adaptation by incorporating the parameter into the chromosome that defines the problem solution.

5.1 Fuzzy logic controllers

This section describes a novel FLC introduced by the authors to control the parameter th in the DCN-THR approach. Its main novelty lies in the incorporation of a set of different rule bases that are enabled depending on historical information extracted from the optimisation process. This historical data is used to guide the adjustment of the parameter th. The pseudocode of the FLC is shown in Algorithm 1. It operates as follows:

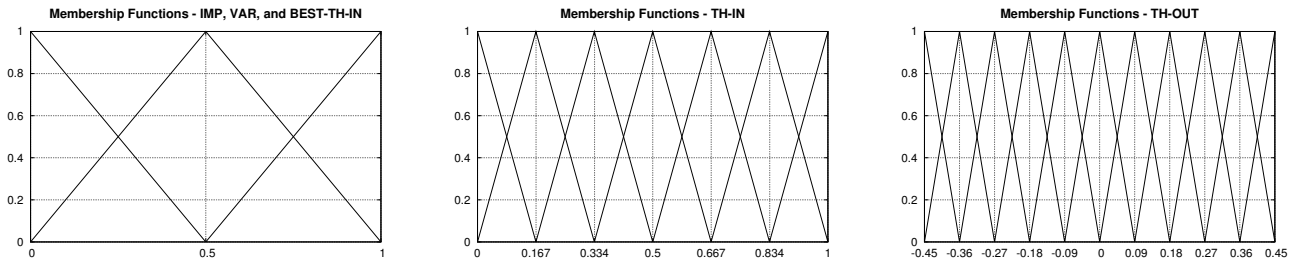


Fig. 1 Membership functions of the input and output variables

Firstly, the initialisation and learning stages—lines 1–4—are carried out. During the initialisation stage, different sample values are generated for the parameter th, distributing them uniformly within the range $[0, 1]$. In order to generate them, a value Δ is considered as the difference between two consecutive samples. Although Δ might be considered as a parameter of the FLC, it is assigned a fixed value regardless of the optimisation problem. Then, in the learning stage, the diversity-based MOEA explained in Section 4 is executed for $numGen$ generations for each of the generated samples in order to gather sufficient information. Once these two stages are complete, the FLC infers the change to be applied to the parameter th—lines 6–11—taking into account the values of the input variables and the rule base selected. Then, the diversity-based MOEA is executed for $numGen$ generations—line 12—with the new value of th. Finally, this process is repeated until the global stopping criterion of the diversity-based MOEA is satisfied.

Note that Mamdani's fuzzy inference method is used for the fuzzy inference process—lines 9–11. In addition, the fuzzy logic operator AND [2] uses the minimum T-norm, the implication method uses the minimum T-norm, the aggregation method applies the maximum S-norm and the centroid algorithm is applied as the defuzzification method. All of these components were selected because they are usually implemented together with Mamdani FLCs. It is important to note that zero-order Takagi-Sugeno-Kang (TSK) FLCs—where the linguistic terms of the output variables are described using a zero order (constant) function instead of membership functions—were also implemented. These FLCs use the weighted average as the defuzzification method. Furthermore, they do not require the use of an aggregation method. The remaining components of the fuzzy inference process were the same as those applied in the Mamdani FLCs exposed herein. The differences between Mamdani and TSK FLCs, however, were not statistically significant. Consequently, only Mamdani FLCs are considered in this paper.

The input variables of the FLC are the following:

- IMP. Calculated as the improvement in the original objective value of the best individual achieved by the diversity-based MOEA—line 12 of Algorithm 1—during the latest $numGen$ generations. This input variable is normalised in order to delimit it to the range $[0, 1]$.
- VAR. A measure of the diversity of the population. The higher its value the more diverse the population. The calculation of this input variable with no normalisation is shown in Equation 2. The values of the decision variable i of individuals j and k are given by $x_j[i]$ and $x_k[i]$. The total number of decision variables is represented by D and N is the population size. The value of VAR* is normalised to enclose the variable VAR in the range $[0, 1]$:

$$\text{VAR}^* = \sum_{i=0}^{D-1} \left[\sum_{j=0}^{N-1} \left[x_j[i] - \frac{1}{N} \cdot \left(\sum_{k=0}^{N-1} x_k[i] \right) \right]^2 \right] \quad (2)$$

- TH-IN. Defined as the current value of parameter th within the range $[0, 1]$.
- BEST-TH-IN. Defined as that value of parameter th that has attained the maximum improvement in the original objective value considering the last k values of the parameter th inferred by the FLC. Its value is also in the range $[0, 1]$.

Two different versions of the FLC are applied. The first one is named FUZZY-A and makes usage of the input variables IMP, VAR, and TH-IN. The second one utilises the input variables IMP, TH-IN, and BEST-TH-IN, and it is called FUZZY-B. For both variants of the FLC, only one output variable is defined, called TH-OUT, which represents the increment or decrement to be applied to the parameter th in order to change its value. The membership functions for both the input and output variables are shown in Figure 1. Due to the computational simplicity and efficiency advantages they offer,

[2] Only the fuzzy logic operator AND is used in the antecedents of the fuzzy rules.

triangular-shaped membership functions were selected for the input and output variables. The linguistic terms represented by the memberships functions—from left to right in Figure 1—are as follows:

- Input variables IMP, VAR, and BEST-TH-IN: LOW (L), MED (M), and HIGH (H).
- Input variable TH-IN: LOW (L), LOW-MED-B (LMB), LOW-MED-A (LMA), MED (M), MED-HIGH-A (MHA), MED-HIGH-B (MHB), and HIGH (H).
- Output variable TH-OUT: NEG-GIANT (NG), NEG-HUGE (NU), NEG-HIGH (NH), NEG-MED (NM), NEG-LOW (NL), ZERO (Z), POS-LOW (PL), POS-MED (PM), POS-HIGH (PH), POS-HUGE (PU), and POS-GIANT (PG).

For each FLC several rule bases are defined. The reason for the use of several rule bases is that different fuzzy rules will be applicable depending on the behaviour exhibited during the previous execution. For instance, if the best results were historically obtained by high values of the parameter th, the selected rule base should promote the use of said high values. Every rule base is composed of different IF-THEN fuzzy rules. The left-hand side of Table 1 shows one of the rule bases defined for the FUZZY-A approach, while the right-hand side shows another one for the FUZZY-B scheme. Only the logic operator AND is used in the antecedents of said fuzzy rules. In general, every fuzzy rule considers three input variables and one output variable. In cases where a '-' is shown, the corresponding fuzzy rule has no dependency on the corresponding variable. The remaining rule bases are not shown due to space constraints, but are similar to those shown herein [3].

In order to select the most suitable set of rules, we propose a novel scoring function. It uses a weighted average that considers historical data on both the improvement in the original objective value and on the degrees of membership of parameter th to each term defined for the input variable TH-IN.

Equation 3 assigns a score to each linguistic term $i \in [0, numTerms - 1]$. After every execution of the optimisation scheme—line 12 of Algorithm 1—the improvement achieved is entered into those vectors $\gamma[i]$ for which the degree of membership of parameter th to the linguistic term i is different from zero. In the same way, said degree of membership is entered into vector $\delta[i]$. Hence, the term d denotes the number of items in the vectors $\delta[i]$ and $\gamma[i]$. Additionally, the value of k is defined as the amount of historical knowledge considered by the FLC, i.e. for each linguistic term, information on the last k improvements achieved is considered. Specifically, for each linguistic term the equation represents a weighted average of its improvements, where greater importance is given to the last executions in which values of the controlled parameter have a high degree of membership to the corresponding linguistic term. Thus, the linguistic term i will be assigned a higher score if the values of parameter th have larger degrees of membership to said linguistic term, and if, at the same time, the values of parameter th are able to achieve better improvements in the original objective.

Note that if $numTerms$ linguistic terms are defined for the input variable TH-IN, $numTerms$ rule bases have to be implemented in order for the FLC to work with the proposed scoring function. Figure 1 shows that seven linguistic terms are defined for the input variable TH-IN, meaning seven different rule bases are implemented. We tested different numbers of fuzzy rule bases and found that the higher the number of rule bases, the smoother the variations in the parameter th inferred by the FLC, and thus the steadier the FLC. However, when considering more than seven fuzzy rule bases, the performance started to degrade somewhat, as it also did with a lower number of fuzzy rule bases. Thus, we opted for seven rule bases, as this yielded the best performance for the FLC. This fact also justifies the usage of seven linguistic terms for the input variable TH-IN, instead of using three linguistic terms as in the case of the remaining input variables. For the remaining input variables, three linguistic terms are used so as to maintain the rule bases as simple as possible. Finally, we should note that the different fuzzy rule sets were obtained using expert knowledge.

$$score[i] = \frac{\sum_{j=1}^{min(k,d)} \gamma[i][d-j] \cdot \delta[i][d-j] \cdot (min(k,d) - j + 1)}{\sum_{j=1}^{min(k,d)} \delta[i][d-j] \cdot (min(k,d) - j + 1)} \quad (3)$$

Once the scores are calculated, the linguistic term with the maximum score is selected. This means that those values of parameter th with a large enough degree of membership to said linguistic term should yield better performance than other values. Therefore, if the linguistic term i is selected, rule base i is enabled. This selected rule base is responsible for adapting the value of parameter th so that it approaches the values represented by term i. For instance, assume that the current value of the parameter th is 0.99 and the most suitable rule base—considering the scoring function—is the one represented by the linguistic term LOW of the input variable TH-IN. This means that historically low values of the parameter th have been able to provide good improvements to the original objective value. Thus, the

[3] The complete specifications for all of the rule bases designed for both versions of the FLC are available as online supplementary material.

Table 1 Rule bases of the FUZZY-A (left-hand side) and FUZZY-B (right-hand side) schemes

Rules	Inputs			Output
ID	TH-IN	IMP	VAR	TH-OUT
1	L	L	L	NL
2	L	L	M	NL
3	L	L	H	PL
4	L	M	-	Z
5	L	H	-	Z
6	LMB	L	-	NM
7	LMB	M	-	NL
8	LMB	H	-	Z
9	LMA	L	-	NH
10	LMA	M	-	NL
11	LMA	H	-	Z
12	M	L	-	NU
13	M	M	-	NL
14	M	H	-	Z
15	MHA	L	-	NG
16	MHA	M	-	NL
17	MHA	H	-	Z
18	MHB	L	-	NG
19	MHB	M	-	NL
20	MHB	H	-	Z
21	H	L	-	NG
22	H	M	-	NL
23	H	H	-	Z

Rules	Inputs			Output
ID	TH-IN	IMP	BEST-TH-IN	TH-OUT
1	L	L	-	PG
2	L	M	-	PL
3	L	H	-	Z
4	LMB	L	-	PG
5	LMB	M	-	PL
6	LMB	H	-	Z
7	LMA	L	-	PG
8	LMA	M	-	PL
9	LMA	H	-	Z
10	M	L	-	PU
11	M	M	-	PL
12	M	H	-	Z
13	MHA	L	-	PH
14	MHA	M	-	PL
15	MHA	H	-	Z
16	MHB	L	-	PM
17	MHB	M	-	PL
18	MHB	H	-	Z
19	H	L	L	NL
20	H	L	M	NL
21	H	L	H	PL
22	H	M	-	Z
23	H	H	-	Z

rule base to be applied in this case is precisely the one shown in the left-hand side of Table 1, considering the approach FUZZY-A. Taking into account a fuzzy set for the variable IMP, which has a large degree of membership to the term LOW, since TH-IN—with value 0.99—is represented by a fuzzy set with a large degree of membership to the term HIGH, the output fuzzy set—the one corresponding to the output variable TH-OUT—will have a large degree of membership to the linguistic term NEG-GIANT (NG). Consequently, the value of the parameter th will be considerably decreased so that it will tend towards lower values.

5.2 Hyper-heuristics

An extension of the hyper-heuristic approach to parameter control first described by Vink and Izzo (2007) is implemented in order to control the parameter th in the DCN-THR approach. This hyper-heuristic has been successfully applied in previous papers (Segura et al, 2013b; Segura, 2012) and is based on using a scoring strategy and a selection strategy for selecting the most appropriate low-level configuration of the approach to be executed. A candidate low-level configuration in this case refers to an instance of the diversity-based MOEA described in Section 4 with a particular setting for the variable th of the auxiliary objective DCN-THR (all other parameters of the algorithm remaining constant). Once a strategy is selected, it is executed until a local stopping criterion is achieved. Afterwards another low-level configuration is selected and executed. the final population of the last low-level configuration used becomes the initial population of the new low-level configuration. This process continues until a global stopping criterion is satisfied. The low-level configuration that must be executed is selected as follows.

First, the *scoring strategy* assigns a score to each low-level configuration. This score estimates the improvement that each low-level configuration can achieve starting from the currently obtained set of solutions. In order to calculate this estimate, the previous improvements on the original objective value achieved by each configuration are used. The improvement (γ) is defined as the difference, in terms of the original objective value, between the best achieved individual and the best initial individual. Assuming a configuration $conf$ which has been executed j times, the score is calculated as a weighted average of its latest k improvements (Equation 4).

$$s(conf) = \frac{\sum_{i=1}^{min(k,j)} (min(k,j) + 1 - i) \cdot \gamma[conf][j-i]}{\sum_{i=1}^{min(k,j)} i} \quad (4)$$

In Equation 4, $\gamma[conf][j-i]$ represents the improvement achieved by configuration $conf$ in execution number $j - i$. Depending on the value of k, the adaptation level of the hyper-heuristic, i.e. the total amount of historical knowledge that the hyper-heuristic considers in order to perform its decisions, can be varied. The weighted average assigns a greater importance to the latest executions.

The score $s(conf)$ is used to calculate a probability of selecting a particular low-level configuration. However, the stochastic behaviour of the low-level meta-

heuristics involved may lead to variations in the results they obtain. As a result, the probability calculation also enables a fraction of selections based on a random scheme. This is implemented as follows. Specifically, the hyper-heuristic can be tuned by means of a parameter β, which represents the minimum selection probability that should be assigned to a low-level configuration. If n_h is the number of low-level configurations involved, a random selection following a uniform distribution is performed in $\beta \cdot n_h$ percentage of the cases. Therefore, the probability of selecting each configuration $conf$ is defined as shown in Equation 5.

$$prob(conf) = \beta + (1 - \beta \cdot n_h) \cdot \left[\frac{s(conf)}{\sum_{i=1}^{n_h} s(i)} \right] \quad (5)$$

Two different schemes based on this hyper-heuristic are applied in this paper.

- The first one is a probabilistic version—HH-PROB—in which the selection probability is proportional to the score $s(conf)$ (Equation 5).
- The second one is an elitist version—HH-ELI—which always selects the low-level configuration with the maximum score $s(conf)$, in addition to the minimum random selections performed for each configuration.

5.3 Self-adaptation

In order to enable the parameter th to undergo self-adaptation, it is encoded within the chromosome, where it is subjected to mutation and crossover operators that change its value during the optimisation process. This relies on the premise that better values of the parameter th will produce better individuals, which in turn will have more opportunities to survive, and consequently propagate these improved parameter values. This is *"the idea of the evolution of evolution"* (Eiben et al, 2007). In the case of self-adaptive approaches to parameter control, the selection and variation operators of the EA are responsible for changes in the parameter values, i.e. the updating mechanism that adapts the parameters is implicit. This differentiates the method from the previously described FLCs and hyper-heuristics in which the adaptation mechanism is external to the EA used.

Figure 2 shows the chromosome of an individual that considers the self-adaptation of the parameter th. For an individual representing a solution to a problem with D decision variables, the values $x[i], i \in [0, D-1]$ represent these decision variables. Three novel versions of the self-adaptive approach are proposed herein:

Fig. 2 Chromosome of the self-adaptive approaches

- SELF-A. The value of the parameter th of the best individual in the population—the one with the lowest original objective value—is applied at each generation to calculate the auxiliary objective value of all individuals.
- SELF-B. The mean value of the parameter th considering all individuals in the population is used in each generation to calculate the auxiliary objective value of every individual.
- SELF-C. The corresponding encoded value of the parameter th is applied to each individual in each generation in order to calculate its own auxiliary objective value.

In regard to the taxonomy discussed in Section 2, which considers the scope affected by the change, SELF-A and SELF-B act at the population level, while SELF-C acts at an individual level. Thus, the encoding of the parameter th into the chromosome can be interpreted in different ways, supplying different algorithm variants in which the scope of the parameter varies.

6 Experimental evaluation

In this section, the experiments conducted with the diversity-based MOEA and the parameter control approaches presented in Sections 4 and 5 are described.

Experimental method Both the diversity-based MOEA and the parameter control approaches were implemented using METCO (León et al, 2009) (*Meta-heuristic-based Extensible Tool for Cooperative Optimisation*). Tests were run on a Debian GNU/Linux computer with four AMD ® Opteron ™ (model 6164 HE) at 1.7 GHz and 64 GB RAM. The compiler was GCC 4.6.3, while the FLCs were implemented using the *fuzzylite* 3.1 library (Rada-Vilela, 2013). As all experiments used stochastic algorithms, each execution was repeated 32 times. Comparisons were performed by applying the following statistical analysis. First, a *Shapiro-Wilk test* was conducted in order to check whether the values of the results followed a normal (Gaussian) distribution or not. If so, the *Levene test* checked for the homogeneity of the variances. If samples had equal variance, an ANOVA *test* was done. Otherwise, a *Welch test* was performed. For non-Gaussian distributions, the non-parametric *Kruskal-Wallis* test was used to compare

Table 2 Parameterisation of the diversity-based MOEA

Parameter	Value	Parameter	Value
Stopping criterion (F1–F19)	$2.5 \cdot 10^6$ evals.	Stopping criterion (Rotated)	$3 \cdot 10^6$ evals.
Population size N (F1–F19)	5 individuals	Population Size N (Rotated)	50 individuals
Mutation rate p_m	$1/D$	Crossover rate p_c	1

Table 3 Parameterisation of the FUZZY-A and FUZZY-B fuzzy logic controllers

Parameter	Value	Parameter	Value
Number of generations ($numGen$)	$5 \cdot 10^2$	Difference among samples (Δ)	0.1
Number of linguistic terms ($numTerms$)	7	Historical knowledge (k)	5

Table 4 Parameterisation of the HH-ELI and HH-PROB hyperheuristics

Parameter	Value	Parameter	Value
Local stopping criterion (generations)	$5 \cdot 10^2$	Minimum selection rate (β)	0.1
Number of low-level configs. (n_h)	6	Historical knowledge (k)	5

the medians of the algorithms. A significance level of 5% was considered.

Problem set Experiments were carried out using a set of 19 single-objective benchmark problems—F1–F19—proposed by Lozano et al (2011). The set defines a number of scalable continuous optimisation problems that combine different properties involving the modality, the separability, and the ease of optimisation dimension by dimension. In the current work, D—the number of variables of these problems—was fixed to 500. Additionally, another experiment was carried out using a set of 9 rotated problems—F4–F6, F9–F11, and F14–F16—proposed by Tang et al (2009). In this case, D was fixed to 1,000 decision variables. In what follows, we will refer to these problems as R4–R6, R9–R11, and R14–R16 to differentiate them from the above benchmarks.

Parameters The experiments conducted used a common parameterisation for the diversity-based MOEA and the different parameter control schemes. Table 2 shows the parameterisation of the diversity-based MOEA described in Section 4. In the case of the F1–F19 benchmarks, the population size was fixed to 5 individuals since, in previous research carried out by the authors (Segura et al, 2013b), it was shown that the application of diversity-based MOEAs with said population size to those problems provided the best results in terms of the error achieved at the end of the executions. Other research has also demonstrated the suitability of EAs with small population sizes to solving certain high-dimensional benchmarks (Olguin-Carbajal et al, 2013). In the case of the rotated problems, the population size was fixed to 50 individuals in order to better deal with premature convergence issues. Finally, the stopping criterion was fixed in keeping with the suggestions given in Lozano et al (2011) for the F1–F19 benchmarks, and in Tang et al (2009) for the rotated problems.

The parameterisations of the different parameter control approaches are shown in Tables 3 and 4, for the FLCs and the hyper-heuristics, respectively. The parameter values of the control methods—minimum selection rate, historical knowledge, number of low-level configurations, number of linguistic terms, etc.—were the same regardless of the problem in question. This means that the control approaches proposed herein are robust, since for a wide range of test cases, promising results can be obtained without changing these parameter values. Thus, the parameters of the control methods do not place additional burdens on the configuration of the diversity-based MOEA. Note also that theHH-ELI and HH-PROB hyper-heuristics were applied using six low-level configurations ($n_h = 6$). Generally, and based on previous work by the authors, a high number of low-level configurations involve a decrease in the quality of the solutions obtained because the hyper-heuristic is not able to carry out the right decisions when a large set of candidate configurations is defined. That is why we selected six low-level configurations instead of assigning a larger value to the parameter n_h. The only difference among the low-level configurations is the value assigned to the parameter th. The values are distributed uniformly in the range $[0, 1]$. Thus, the low-level configurations are defined with values 0, 0.2, 0.4, 0.6, 0.8, and 1 for the parameter th.

Describing the parameterisation of the self-adaptive parameter control approaches is not necessary. Since no additional parameters are defined for them, their parameterisation is the same as in the case of the diversity-based MOEA (Table 2).

Finally, a single-objective EA (*Single-EA*) was also applied to study whether the controlled diversity-based

Table 5 Mean original objective value of best approaches after 5e5 evaluations

Problem	Best Self-Adaptive	HH-Eli	HH-Prob	Fuzzy-A	Fuzzy-B	Single-EA
F1	-4.478003e+02	-4.478371e+02	-4.474119e+02	**-4.483882e+02***	**-4.481555e+02**	-4.428261e+02
F2	-3.976507e+02	**-4.002689e+02***	**-3.995730e+02**	-3.977567e+02	-3.984597e+02	-3.896465e+02
F3	7.860950e+03	**6.307528e+03***	6.892168e+03	7.114925e+03	7.206458e+03	6.795756e+03
F4	-2.788402e+02	**-3.106858e+02***	-3.054929e+02	-3.050468e+02	-3.054261e+02	-2.794897e+02
F5	-1.798369e+02	-1.798284e+02	-1.798023e+02	**-1.798794e+02**	**-1.798873e+02***	-1.792260e+02
F6	-1.399212e+02	-1.399205e+02	-1.399115e+02	-1.399267e+02	**-1.399412e+02***	-1.396095e+02
F7	3.740799e-01	2.928847e-01	3.744123e-01	**2.395465e-01***	**2.492768e-01**	2.654012e+00
F8	**3.570116e+05**	**3.664986e+05**	3.720928e+05	**3.563382e+05***	**3.647259e+05**	4.502028e+05
F9	**2.968789e+02**	**2.957150e+02**	3.014180e+02	**2.928711e+02**	**2.866811e+02***	4.386443e+02
F10	**1.812262e+00***	**2.249266e+00**	2.478268e+00	**2.140365e+00**	**1.893920e+00**	1.188721e+01
F11	**2.949134e+02***	3.035599e+02	3.073306e+02	**2.963466e+02**	**2.999489e+02**	4.407029e+02
F12	4.009808e+01	**3.308254e+01***	**3.481530e+01**	3.874683e+01	3.827866e+01	1.034016e+02
F13	1.017289e+04	**3.860397e+03**	4.409750e+03	7.421083e+03	8.632196e+03	**3.837678e+03***
F14	4.569614e+01	**3.072660e+01***	**3.281441e+01**	**3.343408e+01**	3.537184e+01	5.682281e+01
F15	4.122504e+00	**2.468268e+00***	**2.646807e+00**	3.470253e+00	3.276949e+00	4.448329e+00
F16	1.114501e+02	**9.613186e+01***	1.033108e+02	1.109923e+02	1.092793e+02	2.064661e+02
F17	2.689799e+03	**1.287036e+03***	1.564690e+03	2.383842e+03	2.068250e+03	**1.454827e+03**
F18	5.346952e+01	**5.112424e+01**	5.419024e+01	**4.860644e+01***	5.111202e+01	7.809513e+01
F19	**2.062142e+00**	**1.619345e+00***	**1.982881e+00**	2.149803e+00	2.119459e+00	4.531929e+00

MOEA offers any benefits with respect to said single-objective optimiser when solving single-objective problems. It was selected since it is very similar to the diversity-based MOEA considered herein and because it provided the best results for the majority of the F1–F19 benchmark problems in previous research carried out by the authors (Segura et al, 2013b). Said single-objective approach applies an elitist-based generational survivor selection mechanism, i.e. all parents, except the fittest one, are discarded and replaced by the new offspring for the next generation. The remaining components and parameters were the same as those defined for the diversity-based MOEA (Section 4 and Table 2).

In order to carry out a fair comparison among the control approaches, all of the methods are run considering the same amount of function evaluations, as shown in Table 2 (stopping criterion). One of the main goals of this work is to validate diversity-based MOEAs, as well as the different control approaches proposed herein to control their parameters, by applying them to a set of well-known single-objective benchmarks. It would be interesting, however, to also apply the aforementioned schemes to more complex real-world applications, a significant number of which have execution times that are highly correlated to the number of function evaluations. Since the evaluation of an individual is the most time-consuming part of the entire optimisation scheme, the remaining operations that it must perform are insignificant in terms of the time invested. As a result, in this paper the comparisons carried out only consider the number of function evaluations.

6.1 Analysis of parameter control schemes over a short evaluation timeframe

In the first experiment, the various control approaches are applied to the parameter *th* of the diversity-based MOEA in order to solve each of the benchmark functions F1–F19. The main aim of this study is to analyse the performance of the different parameter control schemes over a relatively short period of $5 \cdot 10^5$ evaluations of the objective functions. Table 5 shows for each benchmark function the mean of the original objective value achieved by the different parameter control approaches after $5 \cdot 10^5$ evaluations. In the case of the different versions of the self-adaptive approach only the data for the scheme which reached the lowest mean of the original objective value after $5 \cdot 10^5$ evaluations is shown. Furthermore, the results obtained by the single-objective EA are also shown. The schemes whose data is shown in bold with an asterisk obtained the lowest mean of the original objective value. It is important to note that these approaches exhibited statistically significant differences when compared to the control methods whose data is not shown in bold, based on the statistical procedure described earlier in this section. If the data from several approaches are shown in bold without an asterisk, then those schemes did not exhibit statistically significant differences versus those which obtained the lowest mean of the original objective value. We note the following observations:

- One or both of the FUZZY-A and FUZZY-B FLCs obtained a statistically significant lower mean with respect to the original objective than the remaining approaches in **four** problems: F1 and F5–F7.
- One or both of the HH-ELI and HH-PROB hyper-heuristics obtained a statistically significant lower mean for the original objective value than the remaining schemes in **six** problems: F2–F4, F12, F15, and F16.
- In **five** problems—F8–F10, F14, and F18—there were no statistically significant differences between the two types of external controllers.

– The self-adaptive approaches and the single-objective EA did not provide any statistically significant advantage on their own in any of the benchmarks.

Thus, following a short evaluation period, the hyper-heuristic-based schemes, and in particular the HH-ELI approach, appear better able to adapt the parameter *th* than the parameter control approaches based on FLCs and self-adaptation. This could be due to the fact that the hyper-heuristic-based methods only have to select the most suitable value for the parameter *th* from amongst a finite set of candidate values that coarsely span the parameter space. In contrast, the FLCs are able to select from an infinite range of possible values. Thus, given only a *short* learning period, the hyper-heuristics outperform the other methods as they are able to exploit existing values that may lie close to the optimal ones, rather than having to explore the space for suitable values. As a result, the methods based on hyper-heuristics are able to provide better solutions than the FLCs in the short term for a larger number of problems.

With respect to the self-adaptive approaches, although they did not outperform either of the other control approaches, for some problems they were able to achieve the same quality level as the other parameter control methods. In the case of problem F19, they outperformed the FLCs together with the hyper-heuristics, and in the case of problem F11, they outperformed the hyper-heuristics together with the FLCs.

Finally, in regard to the single-objective EA, it was outperformed by the diversity-based MOEA adapted by the different control methods in the majority of test cases, except for the F13 and F17 benchmarks, where it did not exhibit statistically significant differences from the HH-ELI hyper-heuristic.

6.2 Analysis of parameter control schemes over a long evaluation time frame

In this section, the parameter control schemes are analysed over a longer evaluation period of $2.5 \cdot 10^6$ evaluations, i.e. at the end of the executions, considering the F1–F19 benchmark functions. Hence, Table 6 shows the same information as Table 5, but for $2.5 \cdot 10^6$ evaluations. We observe the following:

– One or both of the FUZZY-A and FUZZY-B FLCs obtained a statistically significant lower mean with respect to the original objective value than the remaining approaches in **four** problems: F3, F7, F12, and F14.
– One or both of the HH-ELI and HH-PROB hyper-heuristics obtained a statistically significant lower mean for the original objective value than the remaining schemes in **two** problems: F2 and F4.
– In **twelve** problems, there were no statistically significant differences between the two types of external controllers: F1, F5, F6, F8–F11, F13, F15, F16, F18, and F19.
– The self-adaptive approaches, as well as the single-objective EA, did not provide any statistically significant advantage on their own in any of the benchmarks.

It is clear that given a long enough evaluation time, the parameter control methods become less distinguishable. In the long evaluation period, the external method does not exhibit any significant differences for 12 cases, compared to 5 cases in the short evaluation period. It is also clear that given a longer running time, the FLCs are able to provide the best results in a higher number of problems than hyper-heuristics. Regarding the self-adaptive approaches, as in the short term, again they did not provide any benefit for any benchmark in the long term, and in fact were outperformed by the FLCs and hyper-heuristic-based control schemes in every problem, except for the F8 function. Something similar happened in the case of the single-objective EA, which was outperformed by the diversity-based MOEA adapted by the different control schemes in almost every case, except for the F5 and F17 benchmark functions.

6.3 Comparison and analysis between short and long evaluation periods

It is illuminating to compare the change in performance in the methods applied to each of the problems under the two different evaluation scenarios, short-term and long-term. We define the *winning* method as the approach that outperforms each of the other schemes (statistically significant). Thus, the winner is either self-adaptation, hyper-heuristic, FLC, Single-EA or none if no single method differs statistically from all of the others. If we then examine the change in the winning method as we switch from short-term to long-term evaluation, the following observations can be made:

– For **eleven** problems—F2, F4, F7–F11, F13, and F17–F19—changing the length of the evaluation period has no impact on the winning control method.
– For **five** problems—F1, F5, F6, F15, and F16—while there is a clear winner in the short-term evaluation experiment, there is no significant difference between methods when evaluated over a longer time period.
– For **two** problems—F3 and F12—the best method switches from the hyper-heuristic in the short-term

Table 6 Mean original objective value of best approaches after 2.5e6 evaluations

Problem	Best Self-Adaptive	HH-Eli	HH-Prob	Fuzzy-A	Fuzzy-B	Single-EA
F1	-4.499892e+02	**-4.499992e+02***	-4.499988e+02	-4.499988e+02	**-4.499992e+02***	-4.499972e+02
F2	-4.341193e+02	**-4.350975e+02***	**-4.349620e+02**	-4.340944e+02	-4.341680e+02	-4.277183e+02
F3	1.427243e+03	1.807100e+03	1.718612e+03	**1.282274e+03***	**1.547327e+03**	1.909534e+03
F4	-3.299951e+02	**-3.300000e+02***	-3.299999e+02	-3.299998e+02	-3.299998e+02	-3.299943e+02
F5	-1.799766e+02	**-1.799713e+02**	**-1.799836e+02**	**-1.799717e+02**	**-1.799792e+02**	**-1.799904e+02***
F6	-1.399978e+02	**-1.399990e+02**	-1.399988e+02	**-1.399991e+02***	**-1.399991e+02***	-1.399971e+02
F7	7.065683e-03	4.597348e-03	5.994893e-03	**2.057316e-03***	2.533643e-03	4.233317e-02
F8	**1.668117e+05**	**1.676752e+05**	1.690159e+05	**1.624589e+05***	1.706433e+05	2.224828e+05
F9	3.248838e+01	**1.681051e+01**	2.372986e+01	**1.617945e+01**	**1.613916e+01***	7.367794e+01
F10	3.394049e-03	**2.852807e-04***	3.736034e-04	5.373773e-04	**3.173994e-04**	2.169285e-03
F11	3.371631e+01	**1.598078e+01***	2.490122e+01	**1.728919e+01**	**1.728853e+01**	7.274837e+01
F12	9.409282e-01	9.932408e-01	1.060437e+00	**5.706366e-01***	**5.746609e-01**	9.932672e+00
F13	1.948127e+03	**1.782808e+03**	**1.290099e+03***	**1.473589e+03**	**1.581966e+03**	1.723598e+03
F14	3.839717e-01	1.958826e-01	3.125988e-01	**1.418254e-01**	**1.371941e-01***	2.531747e+00
F15	1.514281e-02	**6.026768e-03***	7.459278e-03	**6.769493e-03**	**6.590881e-03**	2.802138e-02
F16	3.635383e+00	**2.254722e+00**	3.054237e+00	**2.237479e+00**	**2.185866e+00***	2.561759e+01
F17	8.514682e+02	**5.468047e+02**	**5.769930e+02**	7.511799e+02	5.339560e+02	**5.256555e+02***
F18	3.500672e+00	**1.813913e+00***	2.260901e+00	**2.039246e+00**	**1.859091e+00**	1.106717e+01
F19	5.165002e-03	**9.889879e-04**	1.524792e-03	**1.077074e-03**	**9.886900e-04***	6.212275e-03

evaluation to the FLCs in the long-term evaluation experiment.
- For **one** problem—F14—while there is no significant difference in method in the short term, the FLC emerges as the best method in the long term.

These results suggest that the coarse-grained approach of the hyper-heuristic, which defines a fixed set of possible values for the parameter th spread uniformly across the full range of possible values, provides sufficient variety to this parameter to yield high-quality results in most problems. The results also suggest that the FLCs select similar values to those already defined by the hyper-heuristics and that the problems are relatively robust to the exact value of th over a certain interval. This is evidenced by the fact that in 5 problems, the performance of the hyper-heuristic and FLCs converged given a long enough evaluation time. In addition, for 12 test cases the hyper-heuristics and FLCs did not present statistically significant differences at the end of the executions.

For 3 problems—F3, F12, and F14—the FLC emerged as the winner in the long term, while either the hyper-heuristic was the clear winner or no method dominated in the short term. This suggests that these problems are particularly sensitive to the parameter th, and that the FLC is able to find a value that yields better results, one that is not present in any of the hyper-heuristic configurations.

To summarise, the most appropriate parameter control approach depends on the optimisation problem being solved. However, for the majority of test cases, FLCs or hyper-heuristics can be applied to obtain promising results, whereas the self-adaptive approaches are not able to provide any advantage over the other control methods when adapting the parameter th. The results appear to substantiate the statement made in Eiben et al (2007): *"self-adaptive methods are efficient methods when applicable ... but are outperformed by clever adaptive methods"*. Finally, we should note that the advantages of using diversity-based MOEAs to solve single-objective problems are proven, since the diversity-based MOEA adapted by the different control approaches was able to statistically outperform the single-objective EA in most problems.

6.4 Comparison of parameter control methods to fixed parameters

The parameter control methods applied herein adapt the value of the parameter th during the course of an evolutionary run; thus, a single run of the optimisation algorithm may utilise many different values over the entire run. In this section, we assess the benefit of adapting the value of the parameter over the course of the run as opposed to simply choosing a single value that remains fixed throughout. The latter approach requires a suitable value for th to be defined; we define 21 different configurations of the diversity-based MOEA with the parameterisation shown in Table 2 for the F1–F19 functions. Every configuration differs only in the value of th. The 21 values of th tested are distributed uniformly in the range $[0, 1]$.

Figure 3 shows the mean of the original objective value achieved after $2.5 \cdot 10^6$ evaluations by the parameter control approaches and by the diversity-based MOEA executed within a range of fixed values of the parameter th—FIXED—for several of the benchmark functions [4]. The conclusions drawn from the plots shown can be generalised as follows. Observing the plots, at least one of the parameter control methods was able to obtain

[4] Due to space constraints, the graphics for every problem are not shown but are available as online supplementary material.

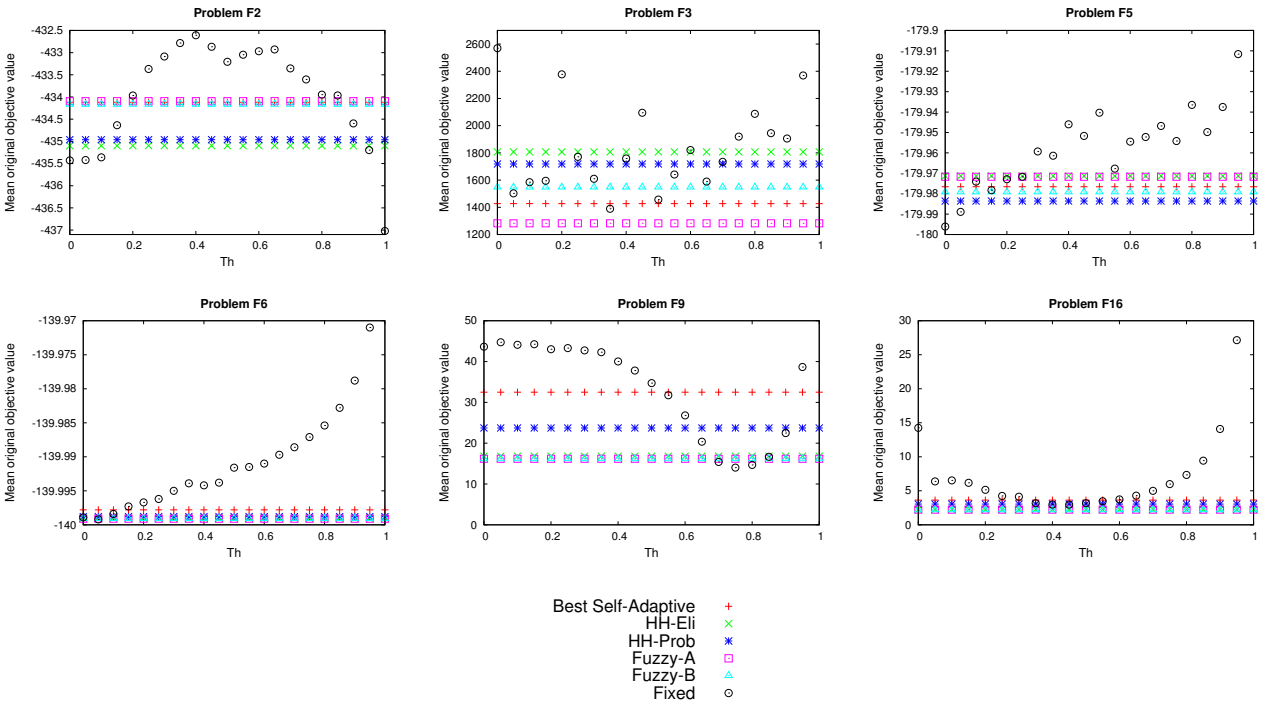


Fig. 3 Mean objective value achieved by the parameter control methods and by the diversity-based MOEA executed with fixed values of the parameter *th*

either similar or better results than any fixed configuration of the FIXED approach in 12 of the 19 problems, namely in problems F3, F6, and F16 shown in Figure 3, as well as in benchmarks F1, F4, F7, F10, F12–F15, and F19. Thus, it appears that the majority of benchmarks benefitted from an approach in which *th* can be varied during the course of the run and an optimal fixed value of *th* cannot be found.

On the other hand, in 7 of the 19 test cases—F2, F5, F8, F9, F11, F17, and F18—some configurations of the FIXED scheme yielded better results than the parameter control approaches, suggesting that there exists some fixed value for the parameter *th* that is adequate during the whole optimisation process. An alternative explanation, however, might lie in the fact that adapting *th* may improve the algorithm but that the changes in the values of the parameter *th* take place so fast that parameter control approaches are not able to detect the changes at the rate required. In this case, fixing the parameter to a suitable value produces more robust behaviour in the diversity-based MOEA. Despite this fact, the results obtained by the control techniques were close to those provided by the best configurations of the FIXED approach for this set of 7 problems.

It is crucial to note that finding a suitable fixed value for *th* required 21 separate runs of the optimisation algorithm. These results, however, are compared to a *single* run of the parameter control methods. Consequently, in addition to the fact that the parameter control methods obtain high quality solutions for most of the problems, the savings in computational resources and time required to produce a good solution are significant across all problems.

In order to quantify these savings, we conducted an additional analysis that relied on *Run-Length Distributions* (RLDs) (Hoos and Stützle, 2005). RLDs show the relationship between the success rate and the number of evaluations needed to achieve it, where the success rate of a particular approach is defined as its probability of achieving a certain quality level. In this case, we fix the quality level as the highest median of the original objective value achieved by the considered schemes at the end of the executions, i.e. at $2.5 \cdot 10^6$ evaluations.

We further calculate the percentage of evaluations—*p*—saved by a certain scheme as compared to the approach that required the largest number of evaluations. This is calculated using Equation 6, where the number of evaluations performed by the scheme considered is denoted by $numEvals$, and $maxEvals$ is the largest number of evaluations performed by any approach, for a particular benchmark function.

$$p = \left(1 - \frac{numEvals}{maxEvals}\right) \cdot 100 \tag{6}$$

Table 7 Maximum number of evaluations needed to achieve the specified quality level and percentage of evaluations saved by each approach

Problem	Best Self-Adaptive	HH-Eli	HH-Prob	Fuzzy-A	Fuzzy-B	Best Fixed
F1	2.45e+06	36.73%	30.61%	38.78%	40.82%	**42.86%**
F2	2.50e+06	6.00%	6.00%	2.50e+06	2.50e+06	**20.00%**
F3	16.00%	4.00%	2.50e+06	**34.00%**	22.00%	22.00%
F4	12.50%	**39.58%**	31.25%	33.33%	**39.58%**	2.40e+06
F5	6.67%	42.22%	2.25e+06	46.67%	**53.33%**	**53.33%**
F6	2.40e+06	27.08%	18.75%	37.50%	31.25%	**50.00%**
F7	2.50e+06	16.00%	8.00%	42.00%	36.00%	**44.00%**
F8	2.00%	2.00%	2.00%	10.00%	2.50e+06	**12.00%**
F9	2.50e+06	24.00%	12.00%	24.00%	24.00%	**28.00%**
F10	2.50e+06	32.00%	30.00%	36.00%	**42.00%**	28.00%
F11	2.50e+06	**28.00%**	12.00%	26.00%	22.00%	**28.00%**
F12	6.00%	6.00%	2.50e+06	32.00%	28.00%	**34.00%**
F13	2.50e+06	72.00%	64.00%	54.00%	52.00%	**80.00%**
F14	2.50e+06	26.00%	20.00%	28.00%	**32.00%**	14.00%
F15	2.50e+06	18.00%	12.00%	16.00%	14.00%	**20.00%**
F16	2.50e+06	16.00%	8.00%	16.00%	**18.00%**	10.00%
F17	2.50e+06	72.00%	68.00%	46.00%	48.00%	**82.00%**
F18	2.50e+06	24.00%	12.00%	16.00%	24.00%	**28.00%**
F19	2.50e+06	28.00%	20.00%	28.00%	**30.00%**	26.00%
Mean	2.27%	27.35%	18.66%	29.70%	29.31%	32.75%

The results are shown in Table 7. For each benchmark problem, the table gives the actual number of evaluations needed by the method that requires the highest number to reach a 50% success rate. In every other column, the percentage of evaluations saved by the corresponding method is shown. In order to calculate the values, in the case of the self-adaptive control scheme, the approach which obtained the lowest mean of the original objective at the end of the executions was used. The single value of th that was found to be optimal in the experiments with the fixed values was also used. Data in bold highlights the approaches that were able to save the highest number of evaluations for each problem. Finally, the last row shows the mean percentage of saved evaluations for all the test cases.

Note that for the majority of benchmarks, the self-adaptive approach invested the highest number of evaluations to achieve a 50% success rate. The FIXED scheme provided the greatest savings in evaluations in 13 benchmark functions. However, it is important to note that in order to find the appropriate value of th with which to run this experiment, 21 separate configurations had to be executed in order to select the best value for this parameter. This represents a significant hidden cost that is not apparent in this table. In contrast, the parameter control methods based on FLCs and hyper-heuristics saved the largest number of evaluations in 8 test cases and required only a *single* run of the algorithm. In 6 of these 8 test cases, the FUZZY-B scheme provided the largest savings.

We should mention that the size of the savings in the case of the control approaches was also significant for the 13 test cases in which the FIXED scheme provided the greatest savings. For instance, for the function F1, the FIXED approach resulted in 42.86% fewer evaluations, while the FUZZY-B scheme saved 40.82%. This fact is also evidenced by the mean percentages of saved evaluations. The FLCs obtained mean percentages that were quite close to the mean percentage of the FIXED scheme. This demonstrates the advantages of using the proposed control approaches versus searching for a fixed value for the parameter th.

6.5 Evaluation of the control schemes with rotated problems

In the last experiment, the different control approaches are applied to the parameter th of the diversity-based MOEA in order to solve the rotated problems R4–R6, R9–R11, and R14–R16. Since in this work the adapted scheme is a diversity-based MOEA, it would be interesting to evaluate how said approach preserves diversity through its application to this set of rotated problems. Table 8 shows the same information as Tables 5 and 6. In this case, the self-adaptive approach is not shown because it did not provide any advantage in previous experiments.

Note that FLCs and hyper-heuristics did not present statistically significant differences between them in any of the problems except for the R4 benchmark, where the hyper-heuristics outperformed the FLCs. Moreover, we should note that for most of the benchmarks, the diversity-based MOEA controlled by FLCs and hyper-heuristics yielded better results than those given by the single-objective EA. Only in problems R9 and R15 did the latter not present statistically significant differences with the former. Hence, the ability of the controlled diversity-based MOEA to preserve a proper diversity in a set of solutions is demonstrated once more when compared to a single-objective approach.

Table 8 Mean original objective value of best approaches after 3e6 evaluations for the rotated problems

Problem	HH-Eli	HH-Prob	Fuzzy-A	Fuzzy-B	Single-EA
R4	**1.046444e+13***	**1.168228e+13**	1.310206e+13	1.272722e+13	1.798935e+13
R5	**3.888834e+08***	**4.090134e+08**	**3.934412e+08**	**4.033390e+08**	4.812920e+08
R6	**1.878252e+07**	1.883079e+07	**1.861230e+07**	**1.860600e+07***	1.966962e+07
R9	**4.364128e+08**	**4.400282e+08**	**4.349376e+08**	**4.348749e+08***	**4.544174e+08**
R10	**6.725422e+03**	**6.671020e+03**	**6.619248e+03***	**6.694242e+03**	6.877248e+03
R11	**1.985014e+02**	**1.978129e+02***	**1.983303e+02**	**1.985502e+02**	2.021087e+02
R14	**6.711666e+08**	**6.606917e+08**	**6.522639e+08***	**6.638463e+08**	8.095365e+08
R15	**1.314203e+04***	**1.324957e+04**	**1.326517e+04**	**1.325826e+04**	**1.332292e+04**
R16	**3.967718e+02**	**3.966346e+02**	**3.957723e+02***	**3.961348e+02**	3.993650e+02

In addition we should mention that for some of the rotated problems, the adaptive diversity-based MOEA was able to provide better results than those given by certain schemes that were specifically designed for said problems, such as a Differential Evolution approach based on a neighbourhood search that promotes intensification with specialised mechanisms (Wang et al, 2010).

7 Conclusions and future work

Meta-heuristics are a set of approximation techniques that have shown promising performance when solving optimisation problems, although they often suffer from premature convergence, resulting in only local optima being found. In an effort to address this drawback, diversity-based MOEAs can be applied to single-objective problems, but this introduces a potential weakness by increasing the number of algorithm parameters that need to be tuned. Appropriate parameter setting is now recognised as a critical part of any meta-heuristic design. Parameter *tuning* approaches attempt to find an optimal set of parameters that remain fixed for the duration of the optimisation procedure. In contrast, parameter *control* approaches attempt to adapt the values of a parameter during the course of the optimisation, based on the assumption that different values are appropriate at different points in the search.

In this paper, we investigate the application of parameter control approaches to adapt the parameters of a diversity-based MOEA when it is applied to solve single-objective optimisation problems. Specifically, we attempt to control the parameter th of the auxiliary objective function DCN-THR. We present a novel parameter control method based on fuzzy logic, and compare this to a previous method introduced by the authors that is based on hyper-heuristics. The main difference between the two methods lies in the fact that the hyper-heuristic approach demands that a fixed set of potential values for the parameter be pre-defined by a user, whereas the fuzzy logic approach is able to select any value within a range. These two methods of externally controlling the parameter are further compared to self-adaptation of the parameter through encoding in the chromosome. Additionally, the adaptive diversity-based MOEA is also compared to a single-objective EA of similar characteristics.

Extensive testing on a wide set of benchmark functions revealed that both the hyper-heuristic and fuzzy logic methods are able to obtain results that are similar to or better than those obtained using a fixed parameter. Moreover, the savings in computational resources and time provided by both types of control schemes are significant, as it is not necessary to search for suitable values of th over multiple experiments. The fact that better results are obtained in many of the problems compared to the fixed methods also highlights that there is an advantage in adapting the parameter over the course of the run, i.e. through parameter *control* rather than parameter *tuning*. On the contrary, self-adaptive approaches did not provide any advantage over the other control strategies considered in terms of quality or length of time invested. Finally, we should note that the benefits of using the adaptive diversity-based MOEA to solve single-objective problems were also proven, as this method was able to outperform the single-objective EA used as the comparison approach in the majority of test cases. In fact, the results attained by the controlled diversity-based MOEA for some of the rotated problems were better than those provided by certain algorithms specifically designed to deal with said rotated problems, even though we did not incorporate specialised components to deal with these problems.

The FLC proposed is novel in its use of parameter control, and furthermore in its use of multiple rule bases depending on feedback from the optimisation procedure. To the best of our knowledge, this is the first time an FLC has been used to control the parameters of the auxiliary objective of a diversity-based MOEA. However, the method has a more general applicability and could be used to control other numeric parameters of other meta-heuristics in the future, including the multi-objective field. If instead of using the original objective value, the value of the input variable IMP is given by some multi-objective performance metric, the FLC can

be applied to control the parameters of multi-objective algorithms. On the other hand, since the different rule bases were obtained through expert knowledge, it would be interesting to apply some kind of automatic learning mechanism in order to obtain the different fuzzy rule sets. This might improve the performance of the FLCs.

Since our diversity-based approach does not make use of an intensification procedure and since said intensification schemes usually provide significant benefits, the fact that our diversity-based approach obtained better results than some specialised schemes is surprising and a clear indication of its promising behaviour. Consequently, the interactions between intensification procedures and diversity-based schemes should be analysed in the future. Finally, it would be interesting to apply our control proposals to adapt the parameters belonging to state-of-the-art algorithms that have provided the best results for the benchmarks considered herein.

Acknowledgements This work was supported by the EC (FEDER) and the Spanish Ministry of Science and Innovation as part of the 'Plan Nacional de I+D+i', with contract number TIN2011-25448. The work of Eduardo Segredo was funded by grant FPU-AP2009-0457. The work was also funded by the HPC-EUROPA2 project (project number: 228398) with the support of the European Commission – Capacities Area – Research Infrastructures.

References

Abbass HA, Deb K (2003) Searching under multi-evolutionary pressures. In: Proceedings of the Fourth Conference on Evolutionary Multi-Criterion Optimization, Springer-Verlag, pp 391–404

Bäck T (1992) The interaction of mutation rate, selection, and self-adaptation within a genetic algorithm. In: Proceedings of the 2nd Conference on Parallel Problem Solving from Nature, North-Holland, Amsterdam

Bäck T, Eiben AE, van der Vaart NAL (2000) An empirical study on gas "without parameters". In: Proceedings of the 6th International Conference on Parallel Problem Solving from Nature, Springer-Verlag, London, UK, UK, PPSN VI, pp 315–324

Bartz-Beielstein T, Chiarandini M, Paquete L, Preuss M (eds) (2010) Experimental Methods for the Analysis of Optimization Algorithms. Springer

Bui L, Abbass H, Branke J (2005) Multiobjective optimization for dynamic environments. In: The 2005 IEEE Congress on Evolutionary Computation, vol 3, pp 2349 – 2356 Vol. 3

Burke EK, Kendall G, Newall J, Hart E, Ross P, Schulenburg S (2003) Hyper-heuristics: An emerging direction in modern search technology. In: Glover F, Kochenberger GA (eds) Handbook of Metaheuristics, International Series in Operations Research & Management Science, vol 57, Springer US, pp 457–474

Burke EK, Hyde M, Kendall G, Ochoa G, Özcan E, Woodward JR (2010) A classification of hyper-heuristic approaches. In: Gendreau M, Potvin JY (eds) Handbook of Metaheuristics, International Series in Operations Research & Management Science, vol 146, Springer US, pp 449–468

Caamaño P, Prieto A, Becerra J, Bellas F, Duro R (2010) Real-valued multimodal fitness landscape characterization for evolution. In: Wong K, Mendis B, Bouzerdoum A (eds) Neural Information Processing. Theory and Algorithms, Lecture Notes in Computer Science, vol 6443, Springer Berlin / Heidelberg, pp 567–574

Črepinšek M, Liu SH, Mernik M (2013) Exploration and exploitation in evolutionary algorithms: A survey. ACM Comput Surv 45(3):35:1–35:33

Davis L (1989) Adapting operator probabilities in genetic algorithms. In: Proceedings of the Third International Conference on Genetic Algorithms, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, pp 61–69

Deb K, Agrawal RB (1995) Simulated binary crossover for continuous search space. Complex Systems 9:115–148

Deb K, Pratap A, Agarwal S, Meyarivan T (2002) A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Transactions on Evolutionary Computation 6:182–197

Eiben AE, Smit SK (2011) Parameter tuning for configuring and analyzing evolutionary algorithms. Swarm and Evolutionary Computation 1(1):19 – 31

Eiben AE, Smith J (2003) Introduction to Evolutionary Computing. Natural Computing Series, Springer

Eiben AE, Michalewicz Z, Schoenauer M, Smith J (2007) Parameter Control in Evolutionary Algorithms. In: Lobo FG, Lima CF, Michalewicz Z (eds) Parameter Setting in Evolutionary Algorithms, Studies in Computational Intelligence, vol 54, Springer, chap 2, pp 19 – 46

Fazzolari M, Alcala R, Nojima Y, Ishibuchi H, Herrera F (2013) A review of the application of multiobjective evolutionary fuzzy systems: Current status and further directions. IEEE Transactions on Fuzzy Systems 21(1):45–65

Fialho A (2010) Adaptive operator selection for optimization. PhD thesis, Université Paris-Sud XI, Orsay, France

Glover FW, Kochenberger GA (2003) Handbook of Metaheuristics (International Series in Operations

Research & Management Science). Springer
Greiner D, Emperador J, Winter G, Galván B (2007) Improving computational mechanics optimum design using helper objectives: An application in frame bar structures. In: Obayashi S, Deb K, Poloni C, Hiroyasu T, Murata T (eds) Evolutionary Multi-Criterion Optimization, Lecture Notes in Computer Science, vol 4403, Springer Berlin / Heidelberg, pp 575–589
Herrera F (2008) Genetic fuzzy systems: taxonomy, current research trends and prospects. Evolutionary Intelligence 1(1):27–46
Herrera F, Lozano M (2001) Adaptive genetic operators based on coevolution with fuzzy behaviors. IEEE Transactions on Evolutionary Computation 5(2):149–165
Herrera F, Lozano M (2003) Fuzzy adaptive genetic algorithms: design, taxonomy, and future directions. Soft Computing 7(8):545–562
Hoos H, Stützle T (2005) Stochastic local search: foundations and applications. The Morgan Kaufmann Series in Artificial Intelligence, Morgan Kaufmann Publishers
Im SM, Lee JJ (2008) Adaptive crossover, mutation and selection using fuzzy system for genetic algorithms. Artificial Life and Robotics 13(1):129–133
Kramer O (2010) Evolutionary self-adaptation: a survey of operators and strategy parameters. Evolutionary Intelligence 3:51–65
Lau HCW, Tang CXH, Ho GTS, Chan TM (2009) A fuzzy genetic algorithm for the discovery of process parameter settings using knowledge representation. Expert Systems and Applications 36(4):7964–7974
León C, Miranda G, Segura C (2009) METCO: A Parallel Plugin-Based Framework for Multi-Objective Optimization. International Journal on Artificial Intelligence Tools 18(4):569–588
Liu D, Liu X (2011) The improved genetic algorithm based on fuzzy controller with adaptive parameter adjustment. In: Zhu M (ed) Information and Management Engineering, Communications in Computer and Information Science, vol 235, Springer Berlin Heidelberg, pp 491–497
Liu J, Lampinen J (2005) A fuzzy adaptive differential evolution algorithm. Soft Computing 9:448–462
Lobo FG, Lima CF, Michalewicz Z (eds) (2007) Parameter Setting in Evolutionary Algorithms, Studies in Computational Intelligence, vol 54. Springer
Lozano M, Molina D, Herrera F (2011) Editorial scalability of evolutionary algorithms and other metaheuristics for large-scale continuous optimization problems. Soft Computing 15(11):2085–2087
Maturana J, Lardeux F, Saubion F (2009) Controlling behavioral and structural parameters in evolutionary algorithms. In: Collet P, Monmarch N, Legrand P, Schoenauer M, Lutton E (eds) Artificial Evolution, Springer, Strasbourg, France, Lecture Notes in Computer Science, vol 5975, pp 110–121
Olguin-Carbajal M, Alba E, Arellano-Verdejo J (2013) Micro-differential evolution with local search for high dimensional problems. In: Proceedings of the 2013 IEEE Congress on Evolutionary Computation (CEC'13), pp 48–54
Qin AK, Huang VL, Suganthan P (2009) Differential evolution algorithm with strategy adaptation for global numerical optimization. IEEE Transactions on Evolutionary Computation 13(2):398–417
Rada-Vilela J (2013) Fuzzylite: a fuzzy logic control library in C++. URL `http://www.fuzzylite.com`
Rechenberg I (1973) Evolutionsstrategie: optimierung technischer systeme nach prinzipien der biologischen evolution. Frommann-Holzboog
Ren Z, Jiang H, Xuan J, Luo Z (2012) Hyper-heuristics with low level parameter adaptation. Evolutionary Computation 20(2):189–227
Rui O, Hajizadeh A, Undeland TM (2010) Parameter optimization of a fuzzy logic controller for a power electronics boost converter using genetic algorithms. In: Proceedings of the 9th WSEAS International Conference on Artificial Intelligence, Knowledge Engineering, and Data Bases, World Scientific and Engineering Academy and Society (WSEAS), Stevens Point, Wisconsin, USA, AIKED'10, pp 120–124
Segura C (2012) Parallel Optimisation Schemes. A Hybrid Scheme based on Hyperheuristics and Evolutionary Computation. PhD thesis, La Laguna, Spain
Segura C, Miranda G, León C (2010) Parallel Hyperheuristics for the Frequency Assignment Problem. Memetic Computing 3(1):33–49
Segura C, Coello Coello C, Miranda G, León C (2013a) Using multi-objective evolutionary algorithms for single-objective optimization. 4OR 11(3):201–228
Segura C, Segredo E, León C (2013b) Analysing the robustness of multiobjectivisation approaches applied to large scale optimisation problems. In: Tantar E, Tantar AA, Bouvry P, Del Moral P, Legrand P, Coello Coello CA, Schütze O (eds) EVOLVE- A Bridge between Probability, Set Oriented Numerics and Evolutionary Computation, Studies in Computational Intelligence, vol 447, Springer Berlin Heidelberg, pp 365–391
Segura C, Segredo E, León C (2013c) Scalability and robustness of parallel hyperheuristics applied to a multiobjectivised frequency assignment problem. Soft Computing 17(6):1077–1093
Smit SK, Eiben AE (2009) Comparing parameter tuning methods for evolutionary algorithms. In: Pro-

ceedings of the Eleventh Congress on Evolutionary Computation, IEEE Press, Piscataway, NJ, USA, CEC'09, pp 399–406

Srinivas M, Patnaik L (1994) Adaptive probabilities of crossover and mutation in genetic algorithms. IEEE Transactions on Systems, Man and Cybernetics 24(4):656 –667

Tang K, Li X, Suganthan PN, Yang Z, Weise T (2009) Benchmark functions for the CEC2010 special session and competition on large-scale global optimization. Tech. rep., Nature Inspired Computation and Applications Laboratory, USTC, China, URL `http://nical.ustc.edu.cn/cec10ss.php`

Toffolo A, Benini E (2003) Genetic diversity as an objective in multi-objective evolutionary algorithms. Evolutionary Computation 11:151–167

Varnamkhasti MJ, Lee LS (2012) A fuzzy genetic algorithm based on binary encoding for solving multidimensional knapsack problems. Journal of Applied Mathematics 2012

Vink T, Izzo D (2007) Learning the best combination of solvers in a distributed global optimization environment. In: Proceedings of Advances in Global Optimization: Methods and Applications (AGO), Mykonos, Greece, pp 13–17

Wang H, Wu Z, Rahnamayan S, Jiang D (2010) Sequential de enhanced by neighborhood search for large scale global optimization. In: Proceedings of the 2010 IEEE Congress on Evolutionary Computation (CEC'10), pp 1–7

Yao L, Jiang YL, Xiao J (2012) An improved fuzzy adaptive genetic algorithm for function optimization. Advanced Materials Research 403-408:2598–2601

Zhou A, Qu BY, Li H, Zhao SZ, Suganthan PN, Zhang Q (2011) Multiobjective evolutionary algorithms: A survey of the state of the art. Swarm and Evolutionary Computation 1(1):32 – 49

Soft Computing manuscript No.
(will be inserted by the editor)

A fuzzy logic controller applied to a diversity-based multi-objective evolutionary algorithm for single-objective optimisation

Eduardo Segredo · Carlos Segura · Coromoto León · Emma Hart

Received: date / Accepted: date

E. Segredo · C. León
Dpto. Ingeniería Informática y de Sistemas
Universidad de La Laguna. Avda. Astrofísico Fco. Sánchez.
Edif. Matemáticas, 38071, Santa Cruz de Tenerife, Spain.
Tel.: +34 922 319 191
Fax: +34 922 318 170
E-mail: esegredo@ull.es

C. Segura
Centro de Investigación en Matemáticas
Área Computación
Callejón Jalisco s/n, Mineral de Valenciana, Guanajuato, Guanajuato 36240, Mexico
E-mail: csegura@ull.es

C. León
E-mail: cleon@ull.es

E. Hart
Institute for Informatics and Digital Innovation.
Edinburgh Napier University, Edinburgh, Scotland, UK.
E-mail: e.hart@napier.ac.uk

1 Rule bases for the fuzzy logic controller FUZZY-A

Table 1 Rule base number 0 (left-hand side) and rule base number 1 (right-hand side)

Rules	Inputs			Output
ID	TH-IN	IMP	VAR	TH-OUT
1	L	L	L	NL
2	L	L	M	NL
3	L	L	H	PL
4	L	M	-	Z
5	L	H	-	Z
6	LMB	L	-	NM
7	LMB	M	-	NL
8	LMB	H	-	Z
9	LMA	L	-	NH
10	LMA	M	-	NL
11	LMA	H	-	Z
12	M	L	-	NU
13	M	M	-	NL
14	M	H	-	Z
15	MHA	L	-	NG
16	MHA	M	-	NL
17	MHA	H	-	Z
18	MHB	L	-	NG
19	MHB	M	-	NL
20	MHB	H	-	Z
21	H	L	-	NG
22	H	M	-	NL
23	H	H	-	Z

Rules	Inputs			Output
ID	TH-IN	IMP	VAR	TH-OUT
1	L	L	-	PM
2	L	M	-	PL
3	L	H	-	Z
4	LMB	L	L	NL
5	LMB	L	M	NL
6	LMB	L	H	PL
7	LMB	M	-	Z
8	LMB	H	-	Z
9	LMA	L	-	NM
10	LMA	M	-	NL
11	LMA	H	-	Z
12	M	L	-	NH
13	M	M	-	NL
14	M	H	-	Z
15	MHA	L	-	NU
16	MHA	M	-	NL
17	MHA	H	-	Z
18	MHB	L	-	NG
19	MHB	M	-	NL
20	MHB	H	-	Z
21	H	L	-	NG
22	H	M	-	NL
23	H	H	-	Z

Table 2 Rule base number 2 (left-hand side) and rule base number 3 (right-hand side)

Rules	Inputs			Output
ID	TH-IN	IMP	VAR	TH-OUT
1	L	L	-	PH
2	L	M	-	PL
3	L	H	-	Z
4	LMB	L	-	PM
5	LMB	M	-	PL
6	LMB	H	-	Z
7	LMA	L	L	NL
8	LMA	L	M	NL
9	LMA	L	H	PL
10	LMA	M	-	Z
11	LMA	H	-	Z
12	M	L	-	NM
13	M	M	-	NL
14	M	H	-	Z
15	MHA	L	-	NH
16	MHA	M	-	NL
17	MHA	H	-	Z
18	MHB	L	-	NU
19	MHB	M	-	NL
20	MHB	H	-	Z
21	H	L	-	NG
22	H	M	-	NL
23	H	H	-	Z

Rules	Inputs			Output
ID	TH-IN	IMP	VAR	TH-OUT
1	L	L	-	PU
2	L	M	-	PL
3	L	H	-	Z
4	LMB	L	-	PH
5	LMB	M	-	PL
6	LMB	H	-	Z
7	LMA	L	-	PM
8	LMA	M	-	PL
9	LMA	H	-	Z
10	M	L	L	NL
11	M	L	M	NL
12	M	L	H	PL
13	M	M	-	Z
14	M	H	-	Z
15	MHA	L	-	NM
16	MHA	M	-	NL
17	MHA	H	-	Z
18	MHB	L	-	NH
19	MHB	M	-	NL
20	MHB	H	-	Z
21	H	L	-	NU
22	H	M	-	NL
23	H	H	-	Z

Table 3 Rule base number 4 (left-hand side) and rule base number 5 (right-hand side)

Rules	Inputs			Output
ID	TH-IN	IMP	VAR	TH-OUT
1	L	L	-	PG
2	L	M	-	PL
3	L	H	-	Z
4	LMB	L	-	PU
5	LMB	M	-	PL
6	LMB	H	-	Z
7	LMA	L	-	PH
8	LMA	M	-	PL
9	LMA	H	-	Z
10	M	L	-	PM
11	M	M	-	PL
12	M	H	-	Z
13	MHA	L	L	NL
14	MHA	L	M	NL
15	MHA	L	H	PL
16	MHA	M	-	Z
17	MHA	H	-	Z
18	MHB	L	-	NM
19	MHB	M	-	NL
20	MHB	H	-	Z
21	H	L	-	NH
22	H	M	-	NL
23	H	H	-	Z

Rules	Inputs			Output
ID	TH-IN	IMP	VAR	TH-OUT
1	L	L	-	PG
2	L	M	-	PL
3	L	H	-	Z
4	LMB	L	-	PG
5	LMB	M	-	PL
6	LMB	H	-	Z
7	LMA	L	-	PU
8	LMA	M	-	PL
9	LMA	H	-	Z
10	M	L	-	PH
11	M	M	-	PL
12	M	H	-	Z
13	MHA	L	-	PM
14	MHA	M	-	PL
15	MHA	H	-	Z
16	MHB	L	L	NL
17	MHB	L	M	NL
18	MHB	L	H	PL
19	MHB	M	-	Z
20	MHB	H	-	Z
21	H	L	-	NM
22	H	M	-	NL
23	H	H	-	Z

Table 4 Rule base number 6

Rules	Inputs			Output
ID	TH-IN	IMP	VAR	TH-OUT
1	L	L	-	PG
2	L	M	-	PL
3	L	H	-	Z
4	LMB	L	-	PG
5	LMB	M	-	PL
6	LMB	H	-	Z
7	LMA	L	-	PG
8	LMA	M	-	PL
9	LMA	H	-	Z
10	M	L	-	PU
11	M	M	-	PL
12	M	H	-	Z
13	MHA	L	-	PH
14	MHA	M	-	PL
15	MHA	H	-	Z
16	MHB	L	-	PM
17	MHB	M	-	PL
18	MHB	H	-	Z
19	H	L	L	NL
20	H	L	M	NL
21	H	L	H	PL
22	H	M	-	Z
23	H	H	-	Z

2 Rule bases for the fuzzy logic controller FUZZY-B

Table 5 Rule base number 0 (left-hand side) and rule base number 1 (right-hand side)

Rules	Inputs			Output
ID	TH-IN	IMP	BEST-TH-IN	TH-OUT
1	L	L	L	NL
2	L	L	M	PL
3	L	L	H	PL
4	L	M	-	Z
5	L	H	-	Z
6	LMB	L	-	NM
7	LMB	M	-	NL
8	LMB	H	-	Z
9	LMA	L	-	NH
10	LMA	M	-	NL
11	LMA	H	-	Z
12	M	L	-	NU
13	M	M	-	NL
14	M	H	-	Z
15	MHA	L	-	NG
16	MHA	M	-	NL
17	MHA	H	-	Z
18	MHB	L	-	NG
19	MHB	M	-	NL
20	MHB	H	-	Z
21	H	L	-	NG
22	H	M	-	NL
23	H	H	-	Z

Rules	Inputs			Output
ID	TH-IN	IMP	BEST-TH-IN	TH-OUT
1	L	L	-	PM
2	L	M	-	PL
3	L	H	-	Z
4	LMB	L	L	NL
5	LMB	L	M	PL
6	LMB	L	H	PL
7	LMB	M	-	Z
8	LMB	H	-	Z
9	LMA	L	-	NM
10	LMA	M	-	NL
11	LMA	H	-	Z
12	M	L	-	NH
13	M	M	-	NL
14	M	H	-	Z
15	MHA	L	-	NU
16	MHA	M	-	NL
17	MHA	H	-	Z
18	MHB	L	-	NG
19	MHB	M	-	NL
20	MHB	H	-	Z
21	H	L	-	NG
22	H	M	-	NL
23	H	H	-	Z

Table 6 Rule base number 2 (left-hand side) and rule base number 3 (right-hand side)

Rules	Inputs			Output
ID	TH-IN	IMP	BEST-TH-IN	TH-OUT
1	L	L	-	PH
2	L	M	-	PL
3	L	H	-	Z
4	LMB	L	-	PM
5	LMB	M	-	PL
6	LMB	H	-	Z
7	LMA	L	L	NL
8	LMA	L	M	PL
9	LMA	L	H	PL
10	LMA	M	-	Z
11	LMA	H	-	Z
12	M	L	-	NM
13	M	M	-	NL
14	M	H	-	Z
15	MHA	L	-	NH
16	MHA	M	-	NL
17	MHA	H	-	Z
18	MHB	L	-	NU
19	MHB	M	-	NL
20	MHB	H	-	Z
21	H	L	-	NG
22	H	M	-	NL
23	H	H	-	Z

Rules	Inputs			Output
ID	TH-IN	IMP	BEST-TH-IN	TH-OUT
1	L	L	-	PU
2	L	M	-	PL
3	L	H	-	Z
4	LMB	L	-	PH
5	LMB	M	-	PL
6	LMB	H	-	Z
7	LMA	L	-	PM
8	LMA	M	-	PL
9	LMA	H	-	Z
10	M	L	L	NL
11	M	L	M	Z
12	M	L	H	PL
13	M	M	-	Z
14	M	H	-	Z
15	MHA	L	-	NM
16	MHA	M	-	NL
17	MHA	H	-	Z
18	MHB	L	-	NH
19	MHB	M	-	NL
20	MHB	H	-	Z
21	H	L	-	NU
22	H	M	-	NL
23	H	H	-	Z

Table 7 Rule base number 4 (left-hand side) and rule base number 5 (right-hand side)

Rules	Inputs			Output
ID	TH-IN	IMP	BEST-TH-IN	TH-OUT
1	L	L	-	PG
2	L	M	-	PL
3	L	H	-	Z
4	LMB	L	-	PU
5	LMB	M	-	PL
6	LMB	H	-	Z
7	LMA	L	-	PH
8	LMA	M	-	PL
9	LMA	H	-	Z
10	M	L	-	PM
11	M	M	-	PL
12	M	H	-	Z
13	MHA	L	L	NL
14	MHA	L	M	NL
15	MHA	L	H	PL
16	MHA	M	-	Z
17	MHA	H	-	Z
18	MHB	L	-	NM
19	MHB	M	-	NL
20	MHB	H	-	Z
21	H	L	-	NH
22	H	M	-	NL
23	H	H	-	Z

Rules	Inputs			Output
ID	TH-IN	IMP	BEST-TH-IN	TH-OUT
1	L	L	-	PG
2	L	M	-	PL
3	L	H	-	Z
4	LMB	L	-	PG
5	LMB	M	-	PL
6	LMB	H	-	Z
7	LMA	L	-	PU
8	LMA	M	-	PL
9	LMA	H	-	Z
10	M	L	-	PH
11	M	M	-	PL
12	M	H	-	Z
13	MHA	L	-	PM
14	MHA	M	-	PL
15	MHA	H	-	Z
16	MHB	L	L	NL
17	MHB	L	M	NL
18	MHB	L	H	PL
19	MHB	M	-	Z
20	MHB	H	-	Z
21	H	L	-	NM
22	H	M	-	NL
23	H	H	-	Z

Table 8 Rule base number 6

Rules	Inputs			Output
ID	TH-IN	IMP	BEST-TH-IN	TH-OUT
1	L	L	-	PG
2	L	M	-	PL
3	L	H	-	Z
4	LMB	L	-	PG
5	LMB	M	-	PL
6	LMB	H	-	Z
7	LMA	L	-	PG
8	LMA	M	-	PL
9	LMA	H	-	Z
10	M	L	-	PU
11	M	M	-	PL
12	M	H	-	Z
13	MHA	L	-	PH
14	MHA	M	-	PL
15	MHA	H	-	Z
16	MHB	L	-	PM
17	MHB	M	-	PL
18	MHB	H	-	Z
19	H	L	L	NL
20	H	L	M	NL
21	H	L	H	PL
22	H	M	-	Z
23	H	H	-	Z

Mean of the original objective value versus threshold value

Soft Computing manuscript No.
(will be inserted by the editor)

A fuzzy logic controller applied to a diversity-based multi-objective evolutionary algorithm for single-objective optimisation

Eduardo Segredo · Carlos Segura · Coromoto León · Emma Hart

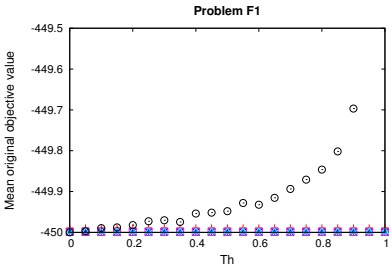
Received: date / Accepted: date

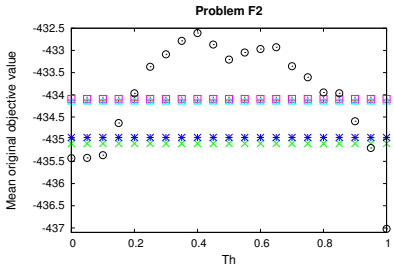
E. Segredo · C. León
Dpto. Ingeniería Informática y de Sistemas
Universidad de La Laguna. Avda. Astrofísico Fco. Sánchez.
Edif. Matemáticas, 38071, Santa Cruz de Tenerife, Spain.
Tel.: +34 922 319 191
Fax: +34 922 318 170
E-mail: esegredo@ull.es

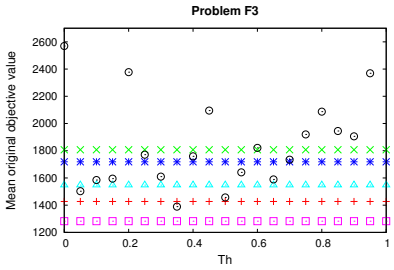
C. Segura
Centro de Investigación en Matemáticas
Área Computación
Callejón Jalisco s/n, Mineral de Valenciana, Guanajuato, Guanajuato 36240, Mexico
E-mail: csegura@ull.es

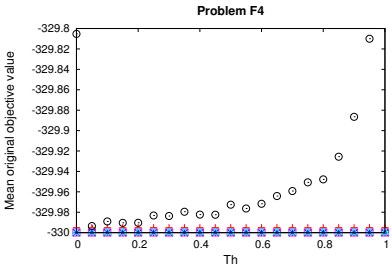
C. León
E-mail: cleon@ull.es

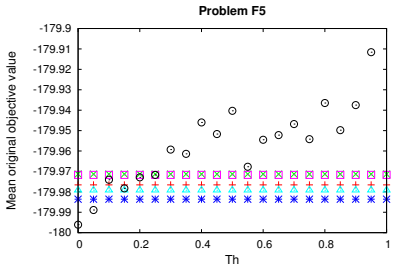
E. Hart
Institute for Informatics and Digital Innovation.
Edinburgh Napier University, Edinburgh, Scotland, UK.
E-mail: e.hart@napier.ac.uk

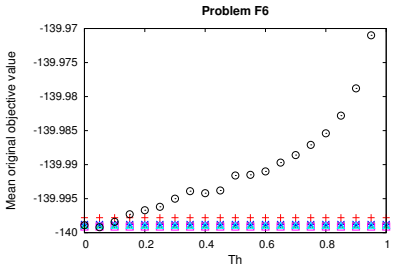

Problem F1
Mean original objective value
-449.5
-449.6
-449.7
-449.8
-449.9
-450
0
0.2
0.4
0.6
0.8
1
Th

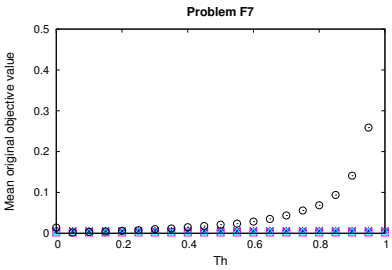

Problem F2
Mean original objective value
-432.5
-433
-433.5
-434
-434.5
-435
-435.5
-436
-436.5
-437
0
0.2
0.4
0.6
0.8
1
Th

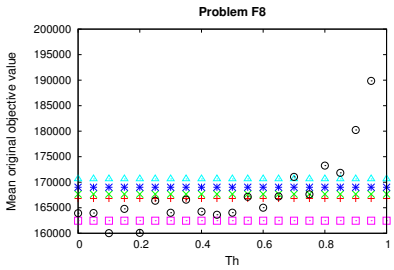

Problem F3
Mean original objective value
2600
2400
2200
2000
1800
1600
1400
1200
0
0.2
0.4
0.6
0.8
1
Th

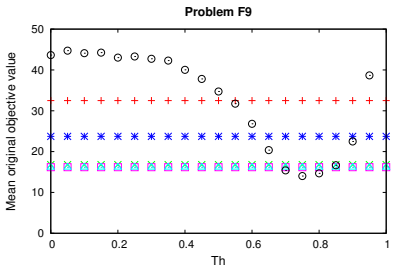

Problem F4
Mean original objective value
-329.8
-329.82
-329.84
-329.86
-329.88
-329.9
-329.92
-329.94
-329.96
-329.98
-330
0
0.2
0.4
0.6
0.8
1
Th

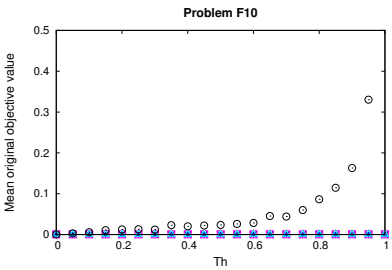

Problem F5
Mean original objective value
-179.9
-179.91
-179.92
-179.93
-179.94
-179.95
-179.96
-179.97
-179.98
-179.99
-180
0
0.2
0.4
0.6
0.8
1
Th

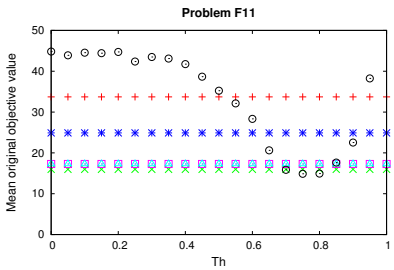

Problem F6
Mean original objective value
-139.97
-139.975
-139.98
-139.985
-139.99
-139.995
-140
0
0.2
0.4
0.6
0.8
1
Th

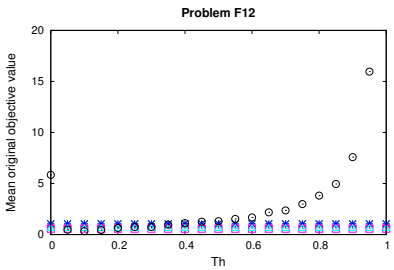

Problem F7
Mean original objective value
0.5
0.4
0.3
0.2
0.1
0
0
0.2
0.4
0.6
0.8
1
Th

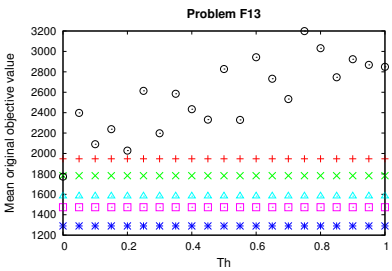

Problem F8
Mean original objective value
200000
195000
190000
185000
180000
175000
170000
165000
160000
0
0.2
0.4
0.6
0.8
1
Th

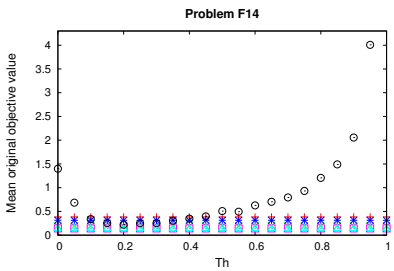

Problem F9
Mean original objective value
50
40
30
20
10
0
0
0.2
0.4
0.6
0.8
1
Th

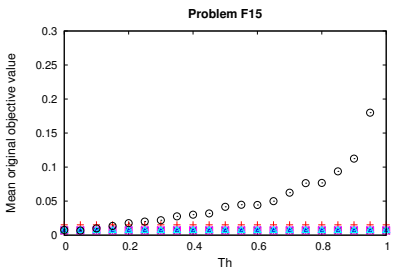

Problem F10
Mean original objective value
0.5
0.4
0.3
0.2
0.1
0
0
0.2
0.4
0.6
0.8
1
Th

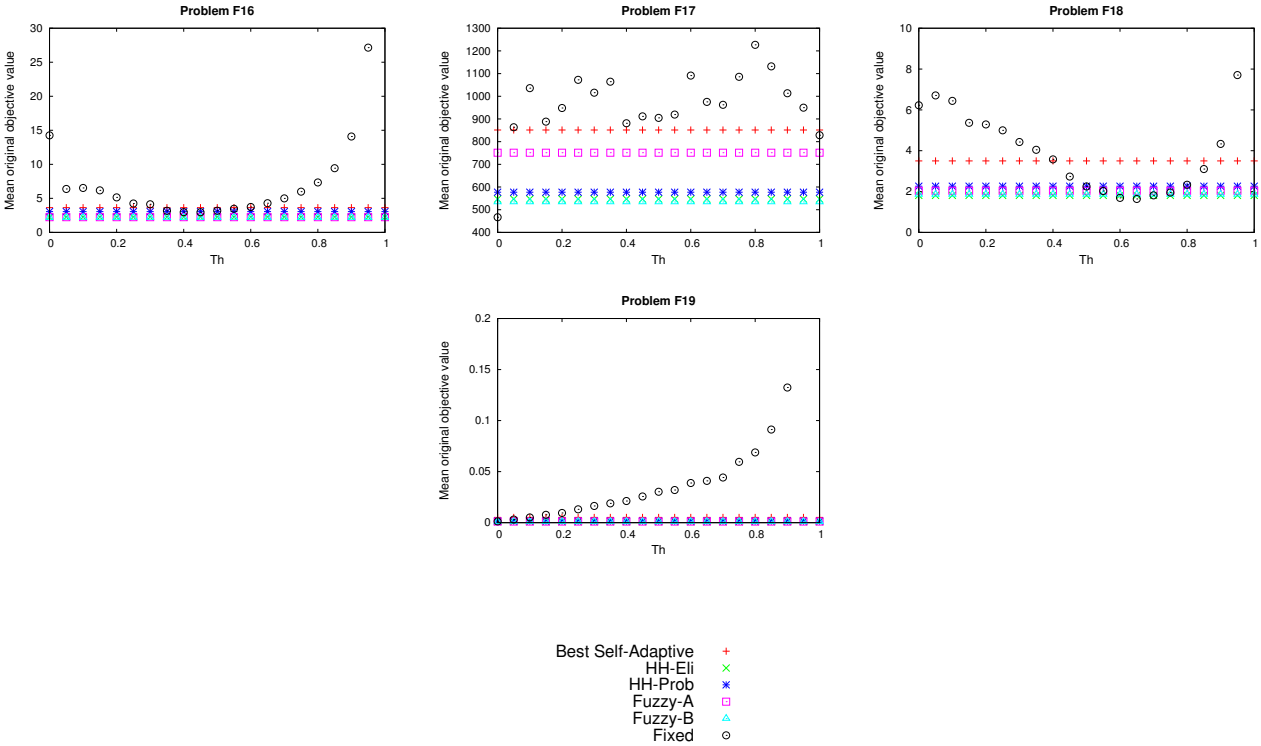

Problem F11
Mean original objective value
50
40
30
20
10
0
0
0.2
0.4
0.6
0.8
1
Th


Problem F12
Mean original objective value
20
15
10
5
0
0
0.2
0.4
0.6
0.8
1
Th


Problem F13
Mean original objective value
3200
3000
2800
2600
2400
2200
2000
1800
1600
1400
1200
0
0.2
0.4
0.6
0.8
1
Th


Problem F14
Mean original objective value
4
3.5
3
2.5
2
1.5
1
0.5
0
0
0.2
0.4
0.6
0.8
1
Th


Problem F15
Mean original objective value
0.3
0.25
0.2
0.15
0.1
0.05
0
0
0.2
0.4
0.6
0.8
1
Th


Problem F16
Mean original objective value
Th
Problem F17
Mean original objective value
Th
Problem F18
Mean original objective value
Th
Problem F19
Mean original objective value
Th
Best Self-Adaptive
HH-Eli
HH-Prob
Fuzzy-A
Fuzzy-B
Fixed