2014 IEEE Congress on Evolutionary Computation (CEC)
July 6-11, 2014, Beijing, China

Control of Numeric and Symbolic Parameters with a Hybrid Scheme based on Fuzzy Logic and Hyper-heuristics

Eduardo Segredo*, Carlos Segura†, Coromoto León*

*Parallel Algorithms and Languages Group, Departamento de Estadística, Investigación Operativa y Computación, Facultad de Matemáticas, Universidad de La Laguna (ULL).
Avda. Astrofísico Fco. Sánchez s/n. 38071 La Laguna, S/C de Tenerife, Spain
†UMI LAFMIA 3175 CNRS at CINVESTAV-IPN, Departamento de Computación,
Av. IPN No. 2508, Col. San Pedro Zacatenco, México D.F. 07300, MÉXICO
Email: esegredo@ull.es, csegura@ull.es, cleon@ull.es

***Abstract*—One of the main disadvantages of Evolutionary Algorithms (EAs) is that they converge towards local optima for some problems. In recent years, diversity-based multi-objective EAs have emerged as a promising technique to prevent from local optima stagnation when optimising single-objective problems. An additional drawback of EAs is the large dependency between the quality of the results provided and the setting of their parameters. By the use of parameter control methods, parameter values can be adapted during the run of an EA. The aim of control approaches is not only to improve the robustness of the controlled algorithm, but also to boost its efficiency. In this paper we apply a novel hybrid parameter control scheme based on *Fuzzy Logic* and *Hyper-heuristics* to simultaneously adapt several numeric and symbolic parameters of a diversity-based multi-objective EA. An extensive experimental evaluation is carried out, which includes a comparison between the hybrid control proposal and a wide range of configurations of the diversity-based multi-objective EA with fixed parameters. Results demonstrate that our control proposal is able to find similar or even better solutions than those obtained by the best configuration of the diversity-based scheme with fixed parameters in a significant number of benchmark problems, demonstrating the advantages of parameter control over parameter tuning for these test cases.**

I. Introduction

Many optimisation problems that arise in real world applications require the use of approximation algorithms. Among them, meta-heuristics are high-level strategies that guide a set of heuristics in the search of an optimum. Evolutionary Algorithms (EAs) [1] are population-based algorithms which belong to the group of meta-heuristics. EAs have shown great promise for calculating solutions to difficult problems. However, in some cases, EAs exhibit a tendency to converge towards local optima. Several methods have been designed to deal with local optima stagnation. One of the methods that has gained some popularity in recent years is based on applying multi-objective schemes to single-objective problems [2]. Several ways of applying multi-objective concepts have been devised, with diversity-based Multi-objective Evolutionary Algorithms (MOEAs) as one of the most promising approaches. A metric of the diversity introduced by each individual is used as an auxiliary objective in these schemes, and therefore they can better deal with premature convergence.

Most popular EA variants have several components and/or parameters, such as the variation operators, or their rates, which must be specified. Generally, the performance of an EA, and consequently the quality of its results are highly dependent on these components and parameters. Hence, the parameter values of an EA must be properly selected. However, finding appropriate parameter settings remains one of the persistent challenges for Evolutionary Computation (EC) [1]. Parameter setting methods are divided into two categories: parameter *tuning* and parameter *control*. In parameter *tuning* the aim is to identify the best set of values for the parameters of a given EA, and then to execute the EA with these values, which remain fixed during the whole 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 being executed.

In this paper we propose a novel control scheme based on the usage of *Fuzzy Logic* and *Hyper-heuristics*. The main advantage of hyper-heuristics is that they are able to control symbolic and numeric parameters. The size of the set of low-level configurations is generally fixed and finite, however, meaning that in the case of controlling numeric parameters, the number of possible values that can be assigned to them is therefore also finite. In contrast, the main benefit of using the FLCs is that the possible values that can be assigned to a certain parameter are not selected from a finite set. Its main drawback lies in the fact that it cannot be directly applied to control symbolic parameters. In order to avoid their drawbacks, and to profit from the strong points of the two approaches, they are combined into a hybrid control scheme to simultaneously adapt symbolic and numeric parameters. This hybrid approach is applied to adapt some parameters of a diversity-based MOEA designed for single-objective optimisation. The contributions of this research work are:

- A novel hybrid parameter control scheme based on fuzzy logic and hyper-heuristics applicable to both numeric and symbolic parameters.
- First time that such a hybrid control scheme is used to simultaneously adapt numeric and symbolic parameters of a diversity-based MOEA.

- A wide comparison between control methods and schemes with fixed parameters that highlights the benefits of parameter control vs. parameter tuning.

The rest of the paper is organised as follows. In Section II, some background in parameter control in EAs is given. Then, Section III describes the diversity-based MOEA applied in this work and provides some background in similar schemes. Our novel hybrid control approach is explained in Section IV, while a detailed analysis of the experimental results is exposed in Section V. Finally, the conclusions and future lines of work are given in Section VI.

II. Background in Parameter Control in Evolutionary Algorithms

Finding the most suitable configuration of an EA is one of the most challenging tasks in the field of EC. In order to completely define an instance of an EA, two types of information are required [3]:

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

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 their respective domains. Symbolic parameters, such as the mutation operator, have a finite domain in which neither order is established nor distance metric is 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 methods can directly be used to look for the appropriate values of the numeric parameters of an EA. However, in the case of symbolic parameters, distance metrics cannot be applied between two values, meaning optimisation schemes are not able to profit from the definition of these types of metrics for setting such parameters.

The goal of parameter control is to design a control strategy that selects the most suitable parameter values for every stage of the optimisation process. The ideas of parameter control were first incorporated in early work on EAs [4], [5]. Nevertheless, the number of recent research works considering parameter control in EAs has increased noticeably [6]. Additionally, parameter control methods have been successfully applied to a wide range of EAs such as *Differential Evolution* (DE) [7] and *Evolution Strategies* (ES) [8], among others. Given the large number of approaches, several taxonomies have been proposed. One of the most popular [9] considers the following types of schemes:

- *Deterministic parameter control*. Parameters are modified by a deterministic rule without using any feedback from the optimisation process.
- *Adaptive parameter control*. Parameters are updated by an external mechanism that uses some feedback from the optimisation process.
- *Self-Adaptive parameter control*. Parameters are encoded into the chromosome and their values are altered by the variation operators of the EA.

The majority of the work on parameter control is focused on the most common parameters of an EA, i.e. the variation operators—mutation and crossover, including their rates—the population size or combinations of all them [9]. In this paper we describe the application of an adaptive hybrid control approach based on *Fuzzy Logic* and *Hyper-heuristics* to adapt the parameters of a diversity-based MOEA designed for single-objective optimisation. A Fuzzy Logic Controller (FLC) is used to adapt numeric parameters, while a hyper-heuristic is used to control symbolic ones. We would like to note that hyper-heuristics have been previously combined with other approaches to obtain hybrid control schemes [10]. However, as far as we know, this is the first time that FLCs and hyper-heuristics are combined into a hybrid scheme.

III. Diversity-based Multi-objective Evolutionary Algorithm

In this section we describe the evolutionary engine that is controlled by the hybrid control scheme depicted in Section IV. The engine is a diversity-based MOEA. In diversity-based multi-objective schemes, a set of objectives is calculated for each individual in the population. The first one is the original objective of the problem being solved, i.e. the value of the considered benchmark function in this case. Remaining objectives—most of the proposals consider only one auxiliary objective as in our case—are measures of the amount of diversity introduced by an individual itself.

Some of the most frequently used auxiliary objectives were defined in [11], [12]. In our proposal we use a variant of the *Distance to the Closest Neighbour* (DCN) metric as the auxiliary objective, which has to be maximised. It is named DCN-THR and it was selected based on previous work by the authors described in [13] in which it was shown that the incorporation of a threshold ratio that penalises low quality individuals provided significant benefits.

In general, any MOEA could be used in combination with this auxiliary objective. There are a large number of MOEAs described in the literature which have shown good performance [14]. The obtained overall results might be highly improved by analysing in a careful way the performance of different MOEAs combined with the auxiliary objective applied herein. However, such a study is out of the scope of this research. Instead, we have decided to use the well-known NSGA-II [15] as the core of the diversity-based multi-objective scheme due to its popularity.

Regarding the variation stage, the diversity-based MOEA relies on the application of a crossover operator and a mutation operator afterwards, with rates p_c and p_m, respectively. Three different crossover operators are tested herein: the *Simulated Binary Crossover* (SBX) [16], the *Arithmetical Crossover* (AX) [17], and the *Parent-Centric Blend Crossover* (PBX) [18] with parameter $\alpha = 0.5$. In the case of the PBX operator, instead of returning a unique offspring as the original proposal does, two offsprings are produced. Both are obtained by considering one parent as the male parent and the other one as the female parent, and vice-versa. With respect to the mutation operator, two different versions of the *Uniform Mutation* (UM) [1] and the *Polynomial Mutation* (PM) [19] are tested. The first variant only mutates one gene of the chromosome with rate p_m—UM-ONE and PM-ONE—while the second one mutates every gene in the chromosome with rate p_m—UM-ALL and PM-ALL. In order to complete the definition of the diversity-based MOEA, it is worth noting that the parent selection mechanism is the well-known *Binary Tournament* [1], whereas the individuals are encoded as arrays of D real values.

At this point, we should note that the crossover and mutation operators are symbolic parameters, while p_m is a continuous numeric parameter. The most suitable values for these parameters could depend on the problem and/or instance being solved or even on the current stage of the optimisation process, and therefore modifying them during the execution might be beneficial. Consequently, the application of parameter control techniques to automatically adapt these parameters may significantly improve both the behaviour and the robustness of the whole diversity-based MOEA.

IV. Parameter Control Scheme based on Fuzzy Logic and Hyper-heuristics

In this section we describe our novel hybrid control proposal based on fuzzy logic and hyper-heuristics. In addition, the components of both the hyper-heuristic and the FLC are detailed. The hybrid scheme is used to simultaneously adapt the crossover and mutation operators, and the mutation rate p_m of the diversity-based MOEA described in Section III.

Figure 1 shows the multi-level architecture of the hybrid control scheme. In the first level, a selection hyper-heuristic [20] is used to control symbolic parameters. For doing that, it selects the most promising low-level configuration—among a set of candidate ones—depending on their past performance. A low-level configuration in this case refers to an instance of the diversity-based MOEA depicted in Section III with a particular setting for the crossover and mutation operators. All other parameters of the algorithm remain constant, with except to the mutation rate p_m, which is adapted by the FLC located at the second level. Once a low-level configuration is selected, only that configuration is executed until the local stopping criterion established by the hyper-heuristic is satisfied. When this happens, another low-level configuration—which could be the same of the last execution—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 is repeated until a global stopping criterion is reached.

In the second level, an FLC is used to adapt numeric parameters. Particularly, the mutation rate p_m of the diversity-based MOEA is adapted herein. The FLC performs its decisions by considering historical information about the values of the parameters inferred in past executions. It is important to recall that, at this level, the low-level configuration is executed until the local stopping criterion established by the hyper-heuristic is achieved. Nevertheless, the FLC also infers changes over the parameter p_m periodically, so another local stopping criterion is established by the FLC. In order to clarify this fact, for instance, consider a global stopping criterion equal to $2.5 \cdot 10^6$ function evaluations, and $2.5 \cdot 10^4$ function evaluations for the local stopping criterion established by the hyper-heuristic. This means that the hyper-heuristic is able to carry out $1 \cdot 10^2$ decisions during the whole optimisation process, changing the values for the variation operators, and that every selected low-level configuration is executed for $2.5 \cdot 10^4$ function evaluations. If the local stopping criterion established by the FLC is equal to $5 \cdot 10^2$ function evaluations, then the FLC infers 50 changes over the parameter p_m during every execution of a low-level configuration.

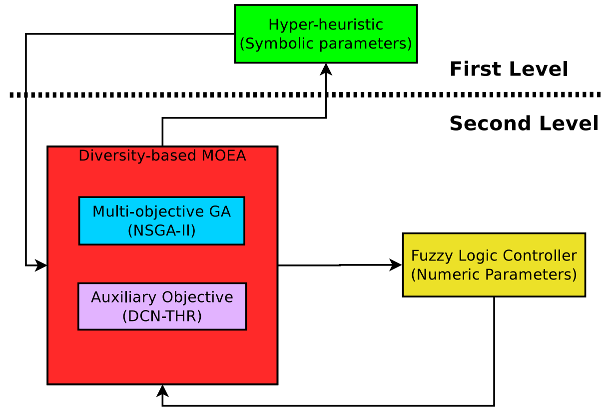

Figure 1. Multi-level architecture of the hybrid control scheme

Finally, it is worth pointing out that not only the parameters of the diversity-based MOEA considered in this work can be controlled by our hybrid proposal, but also other numeric and symbolic parameters belonging to other meta-heuristics.

A. Hyper-heuristic

In this work, a variant of the selection hyper-heuristic proposed in [21] is applied so as to control some symbolic parameters of a diversity-based MOEA. This hyper-heuristic has been successfully applied in previous works [13] and is based on using a scoring and a selection strategy for choosing the most appropriate low-level configuration. 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 current population. Thus, larger values are assigned to more promising schemes considering their historical performance. In order to calculate this estimate, the previous improvements in 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. Considering a configuration $conf$ that has been executed j times, the score $s(conf)$ is calculated as a weighted average of its last k improvements—Eq. (1).

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

In Eq. (1), $\gamma[conf][j-i]$ represents the improvement achieved by configuration $conf$ in execution number $j-i$. The adaptation level of the hyper-heuristic, i.e. the amount of historical knowledge considered to perform its decisions, can be varied depending on the value of k. Finally, the weighted average assigns a greater importance to the latest executions.

The score $s(conf)$ is used to calculate the probability that a particular low-level configuration is selected. However, the stochastic behaviour of the low-level configurations involved may lead to variations in the results they yield. Hence, the probability calculation also enables a fraction of selections based on a random scheme. 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 based on a uniform distribution is performed in $\beta \cdot n_h$ percentage of the cases. Therefore, the probability of selecting each configuration $conf$ is defined by Eq. (2).

$$prob(conf) = \beta + (1 - \beta \cdot n_h) \cdot \left[\frac{s(conf)}{\sum_{i=1}^{n_h} s(i)} \right] \tag{2}$$

B. Fuzzy Logic Controller

This section describes the FLC used to control the mutation rate p_m of the diversity-based MOEA. It incorporates 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 p_m. The pseudocode of this FLC is shown in Algorithm 1.

It is important to remark that the initialisation and learning stages—lines 1–4—are only executed the first time that each low-level configuration is selected by the hyper-heuristic. Moreover, if the low-level configuration had been previously selected, its state after finishing its last execution—including the state of the FLC—is restored before the new run starts. Thus, when the local stopping criterion established by the hyper-heuristic is reached, the state of the current low-level configuration has to be stored for future executions. For the fuzzy inference process—lines 9–11—we note that Mamdani's fuzzy inference method is used. Furthermore, the fuzzy logic operator AND [1] and the implication method apply the minimum T-norm, the aggregation method applies the maximum S-norm, and the centroid algorithm is used as the defuzzification method. All of these components were selected because they are usually implemented together with Mamdani-type FLCs.

The input variables of the FLC—line 7—are as follows:

- IMP. Calculated as the improvement of the original objective value of the best individual achieved by the diversity-based MOEA—line 14 of Algorithm 1—over the last $numEvals$ evaluations. This input variable is delimited to the range $[0, 1]$. Note that the FLC local stopping criterion is given by $numEvals$.

[1] Only the fuzzy logic operator AND is considered.

Algorithm 1 Pseudocode for the fuzzy logic controller

1: **Initialisation**: Generate sample values for the parameter p_m distributed uniformly in its corresponding range considering a certain value Δ as the difference between two consecutive samples
2: **for** (each generated sample value of the parameter p_m) **do**
3: **Learning**: Execute the diversity-based MOEA with this value of p_m during $numEvals$ evaluations in order to gather knowledge
4: **end for**
5: **while** (Hyper-heuristic local stopping criterion is not satisfied) **do**
6: **Transformation of the parameter** p_m. If the range of the parameter p_m is different from the range $[0, 1]$, the current value of this parameter is scaled to the range $[0, 1]$ and named p'_m
7: **Calculation of input variables**. Set the values for the input variables IMP, VAR, PM-IN
8: **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 Eq. (4)
9: **Fuzzification**. Transform the crisp values of the input variables to fuzzy sets using the fuzzification interface
10: **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 PM-OUT
11: **Defuzzification**: Transform the fuzzy set of the output variable PM-OUT to a crisp value Δ_{p_m} using the defuzzification interface (centroid method)
12: **Parameter update**: $p'_m = p'_m + \Delta_{p_m}$. The value of p'_m is enclosed in the range $[0, 1]$
13: **Transformation of the parameter** p'_m. If the range of the parameter p_m is different from the range $[0, 1]$, the current value of p'_m is scaled to the range of the parameter p_m
14: **Execution**: Execute the diversity-based MOEA with the new value of p_m during $numEvals$ evaluations.
15: **end while**

- 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 Eq. (3). 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 VAR in the range $[0, 1]$.

$$\text{VAR}^* = \sum_{i=0}^{D-1} \left[\sum_{j=0}^{N-1} \left[x_j[i] - \frac{1}{N} \cdot \left(\sum_{k=0}^{N-1} x_k[i] \right) \right]^2 \right] \tag{3}$$

- PM-IN. Defined as the current value of the parameter p_m within the range $[0, 1]$.

Only one output variable is defined for the FLC, referred to as PM-OUT, which represents the increment or decrement to be applied to the parameter p_m in order to alter its value. The membership functions for both the input and output variables are shown in Figure 2. Due to the computational simplicity and efficiency advantage they offer, triangular-shaped membership functions were selected for the input and output variables. The linguistic terms represented by the membership functions—from left to right in Figure 2—are as follows:

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

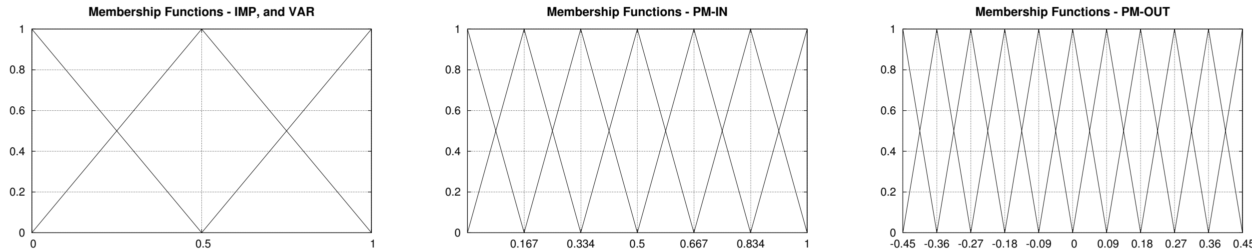

Figure 2. Membership functions for the input and output variables of the fuzzy logic controller

Different rule bases are defined for the FLC. The reason for using different 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 low values of the parameter p_m, the fuzzy rules should promote the usage of such low values. Every rule base is composed of different IF-THEN fuzzy rules. Table I shows one of the rule bases defined for the FLC. The remaining rule bases are not shown due to space constraints. Only the fuzzy logic operator AND is used in the antecedents of these fuzzy rules. In general, every fuzzy rule considers three input variables and one output variable. In those cases where a '-' is shown, the corresponding fuzzy rule has no dependency on the corresponding variable.

In order to select the most suitable set of rules, a scoring function that relies on a weighted mean is used. It considers historical data on both the improvement in the original objective and on the degrees of membership of the parameter p_m to each term defined for the input variable PM-IN. The value of k is defined as the amount of historical knowledge considered by the FLC, i.e. information on the latest k decisions inferred by the FLC is taken into account. On the other hand, d is the total number of decisions that the FLC has carried out, and $numTerms$ is the number of terms defined for the input variable PM-IN. The score assigned to each linguistic term $i \in [0, numTerms - 1]$ is given by Eq. (4). The improvement achieved during execution $d-j$ of the diversity-based MOEA—line 14 of Algorithm 1—is given by $\gamma[d-j]$. In addition, the degree of membership of parameter p_m to the linguistic term i during execution $d-j$ is represented by $\delta[i][d-j]$. Thus, the linguistic term i will be assigned a higher score if the values of parameter p_m have larger degrees of membership to said linguistic term, and if, at the same time, the values of parameter p_m are able to achieve higher improvements in the original objective. Finally, note that the scoring function assigns more importance to the latest decisions.

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

Note that if $numTerms$ linguistic terms are defined for the variable PM-IN, $numTerms$ rule bases have to be implemented such that the FLC works with the proposed scoring function. Figure 2 shows that seven linguistic terms are defined for the input variable PM-IN, so 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 of the parameter p_m 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 rule bases. For the remaining input variables, three linguistic terms are used so as to maintain the rule bases as simple as possible.

Table I. DEFINITION OF ONE OF THE RULE BASES FOR THE FLC

Rules	Inputs			Output
ID	PM-IN	IMP	VAR	PM-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	PL
20	H	L	M	PL
21	H	L	H	NL
22	H	M	-	Z
23	H	H	-	Z

Once scores are calculated, the term with the maximum score is selected. This means that those values of parameter p_m with a large enough degree of membership to this linguistic term should provide better performance than other values. Therefore, if the linguistic term i is selected as the most proper one, rule base i is enabled. This rule base is responsible for adapting the value of parameter p_m so that it approaches the values represented by term i. For instance, assume that the current value of parameter p_m is 0.01 and the most suitable rule base—considering the scoring function—is the one represented by the linguistic term HIGH of the input variable PM-IN. This means that historically high values of parameter p_m have yielded good improvements in the original objective value. Thus, the rule base to be applied in this case is precisely the one shown in Table I. If a fuzzy set for the variable IMP, which has a large degree of membership to the

Table II. PARAMETERISATION OF THE DIVERSITY-BASED MOEA

Parameter	Value	Parameter	Value
Stopping criterion	$2.5 \cdot 10^6$ evals.	Crossover rate (p_c)	1
Population size (N)	5 individuals	Mutation rates (p_m) – UM-ONE, PM-ONE	$0.2, 0.36, 0.52, 0.68, 0.84, 1$
Crossover operators	SBX, AX, PBX	Mutation rates (p_m) – UM-ALL, PM-ALL	$0.0002, 0.00056, 0.00092, 0.00128, 0.00164, 0.002$
Mutation operators	UM-ONE, PM-ONE, UM-ALL, PM-ALL	Auxiliary objective	DCN-THR

Table III. PARAMETERISATION OF THE HYPER-HEURISTIC

Parameter	Value	Parameter	Value
Local stopping criterion	$2.5 \cdot 10^4$ evals.	Minimum selection rate (β)	0.1
Number of low-level configs. (n_h)	12 configs.	Historical knowledge (k)	5

Table IV. PARAMETERISATION OF THE FUZZY LOGIC CONTROLLER

Parameter	Value	Parameter	Value
Local stopping criterion ($numEvals$)	$5 \cdot 10^2$ evals.	Difference among samples (Δ)	0.1
Number of linguistic terms ($numTerms$)	7	Historical knowledge (k)	5
Range of the parameter p_m – UM-ONE, PM-ONE	$[0.2, 1]$	Range of the parameter p_m – UM-ALL, PM-ALL	$[0.0002, 0.002]$

term LOW, since PM-IN—with value 0.01—is represented by a fuzzy set with a large degree of membership to the term LOW, then the output fuzzy set—the one corresponding to PM-OUT—will have a large degree of membership to the linguistic term POS-GIANT (PG). Thus, the value of the parameter p_m will be increased so that it will tend towards higher values.

V. EXPERIMENTAL EVALUATION

In this section, the experiments conducted with the diversity-based MOEA and the hybrid parameter control scheme presented in Sections III and IV are described.

a) Experimental Method: Both the diversity-based MOEA and the hybrid control approaches were implemented using METCO (*Meta-heuristic-based Extensible Tool for Cooperative Optimisation*) [22]. The compiler was the GCC 4.6.3, while the FLC was implemented using the *fuzzylite* 3.1 library [23]. As all experiments used stochastic algorithms, each execution was repeated 32 times. Comparisons were performed applying the following statistical analysis. First, a *Shapiro-Wilk test* was performed 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 the 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. A significance level of 5% was considered.

b) Problem Set: Experiments were carried out using a set of 19 single-objective benchmark problems—F1–F19—proposed in [24]. The set defines a number of scalable continuous optimisation problems, which combine different properties regarding the modality, the separability, and the ease of optimisation dimension by dimension, i.e. whether the objective value can be optimised by independently adjusting each decision variable or not. In the current work, D—the number of decision variables—was fixed to 500.

c) Parameters: Table II shows the parameterisation of the diversity-based MOEA. In the first experiment, 72 different configurations of the diversity-based MOEA with fixed values for the parameters were executed for each benchmark function. The configurations were obtained by combining the values shown in Table II for the crossover and mutation operators, as well as for the mutation rate p_m. Depending on the applied mutation operator, the values assigned to p_m were different. In addition, note that the population size was fixed to 5 individuals. In previous research, the application of diversity-based MOEAs to this set of benchmarks reported the best results when using low population sizes [13]. Finally, since the evaluation of an individual is one of the most computationally expensive operations, the different stopping criteria were set in terms of the total amount of evaluations performed. In the second experiment, the hybrid control scheme was executed to adapt the values of the crossover and mutation operators, and the mutation rate p_m of the diversity-based MOEA. The remaining parameters of the MOEA were kept as shown in Table II. The hyper-heuristic and the FLC parameterisations are shown in Tables III and IV, respectively. It can be noted that the hyper-heuristic of the hybrid scheme had to select among $n_h = 12$ low-level configurations. This is because twelve possible combinations were defined considering three crossover operators and four mutation operators—Table II. Therefore, the differences among the low-level configurations lied in the particular values given to the crossover and mutation operators. Lastly, depending on the mutation operator defined for each low-level configuration, the range of possible values that the FLC was able to infer for p_m was different.

The main aim of the experiments was twofold. Firstly, to study the performance of our novel hybrid control scheme here proposed. Secondly, to analyse if parameter control gives some benefit with regard to tuning the crossover and mutation operators, and the mutation rate p_m.

Table V shows, for each benchmark function, the parameter values of the best configuration—the one that achieved the lowest median of the error—of the diversity-based MOEA executed with fixed parameters. Remember that 72 different configurations of the diversity-based MOEA were executed for each function. With regard to parameter tuning, the PBX crossover operator seems to be the most suitable, while the most appropriate values for the mutation operator and its rate p_m seem to be the UM-ALL operator and 0.0002, respectively, for a wide range of benchmark functions. Nevertheless, the most appropriate values for the crossover and mutation op-

Table V. PARAMETERISATION OF THE BEST FIXED CONFIGURATION OF THE DIVERSITY-BASED MOEA

Problem	Crossover	Mutation	p_m
F1	SBX	UM-ALL	0.0002
F2	PBX	UM-ALL	0.00092
F3	PBX	UM-ALL	0.00056
F4	SBX	UM-ONE	0.68
F5	SBX	UM-ONE	0.2
F6	PBX	PM-ALL	0.00056
F7	PBX	PM-ALL	0.0002
F8	AX	PM-ALL	0.00128
F9	PBX	UM-ALL	0.0002
F10	SBX	UM-ALL	0.0002
F11	PBX	UM-ALL	0.0002
F12	PBX	UM-ALL	0.0002
F13	SBX	UM-ALL	0.0002
F14	PBX	UM-ALL	0.0002
F15	PBX	PM-ALL	0.0002
F16	PBX	UM-ALL	0.0002
F17	SBX	UM-ALL	0.0002
F18	PBX	UM-ALL	0.0002
F19	SBX	UM-ALL	0.0002

Table VI. MEDIAN OF THE ERROR FOR THE HYBRID CONTROL SCHEME AND THE BEST FIXED CONFIGURATION OF THE DIVERSITY-BASED MOEA

Problem	Hybrid Scheme	Best Fixed Conf.	↑	↓	↔
F1	1.3281037e-06	**3.4106051e-13**	66	5	1
F2	**7.6975000e+00**	1.0592500e+01	72	0	0
F3*	9.4457000e+02	**2.9075500e+02**	40	18	13
F4	**3.0795973e-05**	7.8476094e-04	72	0	0
F5*	2.6413124e-07	8.3753093e-10	68	0	2
F6	8.3838312e-05	7.5147903e-05	71	0	1
F7	**1.6071950e-04**	2.1048950e-04	72	0	0
F8	4.7819750e+04	**3.2891500e+04**	55	15	2
F9	**3.5459550e+00**	5.5819750e+00	72	0	0
F10	1.5165250e-06	**5.1276500e-13**	65	5	2
F11	**3.6804850e+00**	5.6666000e+00	72	0	0
F12	1.5824200e-01	**1.1245450e-01**	71	1	0
F13*	7.8921500e+02	**2.9386450e+02**	62	2	4
F14	**5.1547950e-02**	7.4521000e-02	72	0	0
F15	1.3515900e-04	**9.6688000e-05**	71	1	0
F16	**1.0623750e+00**	1.7717600e+00	72	0	0
F17	9.0679350e+01	7.1111250e+01	67	0	5
F18	**4.2261050e-01**	6.4425800e-01	72	0	0
F19	3.0685100e-05	**3.0367050e-08**	67	5	0

erators, and for the mutation rate p_m change depending on the considered benchmark function. Therefore, it would be interesting to check whether our hybrid control approach is able to provide similar results without the need of executing a large amount of configurations of the diversity-based MOEA in order to look for its best parameter values. For benchmarks where different parameter values are the most appropriate depending on the stage of the search process, the hybrid control scheme could provide even better results than those given by the best fixed configuration of the diversity-based MOEA.

Table VI shows, for each problem, the median of the error achieved by the hybrid parameter control scheme and by the best fixed configuration of the diversity-based MOEA—the one shown in Table V. For benchmarks where differences were statistically significant—following the statistical procedure explained above—data belonging to the approach that obtained the lowest median and mean of the error is shown in bold. On the contrary, if differences between the hybrid control scheme and the best fixed configuration were not statistically significant, data of both approaches is not shown in bold. This last fact happened for benchmark functions **F5, F6**, and **F17**. Moreover, for each benchmark, the hybrid control scheme was statistically compared to the 72 fixed configurations of the diversity-based MOEA. Thus, Table VI also shows the number of fixed configurations which were statistically outperformed [2] (↑) by the hybrid scheme, the number of fixed configurations that statistically outperformed the hybrid scheme (↓), and the number of fixed configurations which did not present statistically significant differences with the hybrid method (↔). Finally, for test cases where an '*' is shown, some fixed configurations of the diversity-based MOEA presented statistically significant differences with the hybrid scheme. However, one of both approaches obtained the lowest mean of the error, while the other obtained the lowest median of the error, so we could not determine the winning approach.

We would like to mention that for **eight** benchmark functions—F2, F4, F7, F9, F11, F14, F16, and F18—the hybrid control scheme was able to statistically outperform every fixed configuration of the diversity-based MOEA. This could be explained by the fact that depending on the stage of the optimisation process, the most appropriate values for the controlled parameters are different, and the hybrid control scheme is able to detect these changes. Hence, the hybrid scheme provides better results since it is able to adapt the parameters of the diversity-based MOEA while it is being executed, whereas the fixed configurations are executed with the same parameter values during the whole run, without considering any adaptation.

In contrast, it can be noted that for **eight** test cases—F1, F3, F8, F10, F12, F13, F15, and F19—some fixed configurations of the diversity-based MOEA obtained statistically significant better results than the hybrid parameter control scheme, indicating that, for these benchmarks, there exist some fixed values for the parameters which are suitable during the whole optimisation process. An alternative explanation however might lie in the fact that adapting these parameters may improve the behaviour of the diversity-based MOEA but that the changes in the values of its parameters take place so fast that the hybrid control scheme is not able to detect such changes at the rate required. In this case, fixing the parameters to suitable values produce more robust behaviour in the diversity-based MOEA. Despite this fact, the results obtained by the hybrid control scheme were also competitive for this set of eight problems.

Finally, we should note that the benefits of using the hybrid control approach are even higher if we consider that 72 different configurations of the diversity-based MOEA were executed to look for the best set of values for its parameters, while only a *single* run of the hybrid control scheme was performed. Hence, besides the fact that the hybrid scheme obtains high quality results on most problems, the savings in computational resources and time required to produce good solutions are significant across all benchmarks when using it.

VI. CONCLUSIONS AND FUTURE WORK

One of the most common disadvantages of EAs—and other meta-heuristics—is that they have a tendency to converge towards local optima for some optimisation problems. Several methods have been designed with the aim of dealing with local optima stagnation. However, during last years, diversity-based MOEAs have been applied as a promising approach to prevent

[2] The approach A statistically outperforms the approach B if both present statistically significant differences, and A obtains a lower median and mean of the error than B.

from local optima stagnation when optimising single-objective problems. In order to apply a diversity-based MOEA to a single-objective problem, a metric of the diversity introduced by each individual is used as an auxiliary objective, besides the usage of the original objective of the problem which is being solved.

On the other hand, another drawback of EAs is that they have a noticeable number of parameters that must be set properly in order to improve their performance. Parameter tuning approaches attempt to find an optimal set of parameter values that remain fixed during the course of the optimisation process. In contrast, parameter control schemes attempt to adapt the values of a parameter during the course of the optimisation based on the assumption that different values are better suited at different points in the search. In this paper a novel hybrid control scheme based on an FLC and a hyper-heuristic is proposed to simultaneously adapt several numeric and symbolic parameters of a diversity-based MOEA designed for single-objective optimisation. Particularly, the crossover and mutation operators, as well as the mutation rate p_m, are controlled. To the best of our knowledge, this is the first time that different numeric and symbolic parameters of a diversity-based MOEA are simultaneously adapted by the use of a control scheme like the one proposed herein.

The extensive experimental evaluation carried out over a set of 19 well-known benchmark functions revealed that the novel hybrid control scheme is able to obtain similar or even better results than those provided by a wide range of configurations of the diversity-based MOEA executed with fixed parameters. The fact that better results are returned by the hybrid parameter control approach as compared to the fixed configurations of the diversity-based MOEA also highlights the advantage to be gained by adapting the parameters over the course of the run, i.e. in parameter control rather than in parameter tuning.

Finally, we should note that other numeric and symbolic parameters belonging to other meta-heuristics could be controlled with our novel hybrid control scheme. Both main components of the hybrid control approach—the FLC and the hyper-heuristic—have been designed so that the hybrid parameter control scheme can be applied to different meta-heuristics. An interesting line of future work could therefore be the adaptation of other numeric and symbolic parameters belonging to some state-of-the-art algorithms.

ACKNOWLEDGEMENT

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. Finally, we would like to thank ALUMNI-ULL for its support.

REFERENCES

[1] A. E. Eiben and J. Smith, *Introduction to Evolutionary Computing*, ser. Natural Computing Series. Springer, 2003.

[2] C. Segura, C. A. Coello Coello, G. Miranda, and C. León, "Using multi-objective evolutionary algorithms for single-objective optimization," *4OR*, vol. 11, no. 3, pp. 201–228, 2013.

[3] S. K. Smit and A. E. Eiben, "Comparing parameter tuning methods for evolutionary algorithms," in *Proceedings of the Eleventh Congress on Evolutionary Computation*, ser. CEC'09. Piscataway, NJ, USA: IEEE Press, 2009, pp. 399–406.

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

[5] I. Rechenberg, *Evolutionsstrategie: optimierung technischer systeme nach prinzipien der biologischen evolution*. Frommann-Holzboog, 1973.

[6] F. G. Lobo, C. F. Lima, and Z. Michalewicz, Eds., *Parameter Setting in Evolutionary Algorithms*, ser. Studies in Computational Intelligence. Springer, 2007, vol. 54.

[7] A. K. Qin, V. L. Huang, and P. N. Suganthan, "Differential evolution algorithm with strategy adaptation for global numerical optimization," *IEEE Trans. Evol. Comput.*, vol. 13, no. 2, pp. 398–417, Apr. 2009.

[8] O. Kramer, "Evolutionary self-adaptation: a survey of operators and strategy parameters," *Evolutionary Intelligence*, vol. 3, pp. 51–65, 2010.

[9] A. E. Eiben, Z. Michalewicz, M. Schoenauer, and J. Smith, "Parameter Control in Evolutionary Algorithms," in *Parameter Setting in Evolutionary Algorithms*, ser. Studies in Computational Intelligence. Springer, 2007, vol. 54, ch. 2, pp. 19 – 46.

[10] Z. Ren, H. Jiang, J. Xuan, and Z. Luo, "Hyper-heuristics with low level parameter adaptation," *Evol. Comput.*, vol. 20, no. 2, pp. 189–227, Jun. 2012.

[11] A. Toffolo and E. Benini, "Genetic diversity as an objective in multi-objective evolutionary algorithms," *Evol. Comput.*, vol. 11, pp. 151–167, May 2003.

[12] L. Bui, H. Abbass, and J. Branke, "Multiobjective optimization for dynamic environments," in *The 2005 IEEE Congress on Evolutionary Computation*, vol. 3, September 2005, pp. 2349 – 2356 Vol. 3.

[13] C. Segura, E. Segredo, and C. León, "Analysing the robustness of multiobjectivisation approaches applied to large scale optimisation problems," in *EVOLVE- A Bridge between Probability, Set Oriented Numerics and Evolutionary Computation*, ser. Studies in Computational Intelligence. Springer Berlin Heidelberg, 2013, vol. 447, pp. 365–391.

[14] A. Zhou, B.-Y. Qu, H. Li, S.-Z. Zhao, P. N. Suganthan, and Q. Zhang, "Multiobjective evolutionary algorithms: A survey of the state of the art," *Swarm and Evol. Comput.*, vol. 1, no. 1, pp. 32 – 49, 2011.

[15] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, pp. 182–197, 2002.

[16] K. Deb and R. B. Agrawal, "Simulated binary crossover for continuous search space," *Complex Systems*, vol. 9, pp. 115–148, 1995.

[17] Z. Michalewicz, D. Dasgupta, R. G. L. Riche, and M. Schoenauer, "Evolutionary algorithms for constrained engineering problems," *Evol. Comput.*, vol. 4, pp. 1–32, 1996.

[18] M. Lozano, F. Herrera, N. Krasnogor, and D. Molina, "Real-coded memetic algorithms with crossover hill-climbing," *Evol. Comput.*, vol. 12, no. 3, pp. 273–302, Sep. 2004.

[19] K. Deb and M. Goyal, "A Combined Genetic Adaptive Search (GeneAS) for Engineering Design," *Computer Science and Informatics*, vol. 26, no. 4, pp. 30–45, 1996.

[20] E. K. Burke, M. Hyde, G. Kendall, G. Ochoa, E. Ozcan, and R. Qu, "Hyper-heuristics: A Survey of the State of the Art," School of Computer Science and Information Technology, University of Nottingham, Computer Science, Tech. Rep. NOTTCS-TR-SUB-0906241418-2747, 2010.

[21] T. Vink and D. Izzo, "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, June 2007, pp. 13–17.

[22] C. León, G. Miranda, and C. Segura, "METCO: A Parallel Plugin-Based Framework for Multi-Objective Optimization," *International Journal on Artificial Intelligence Tools*, vol. 18, no. 4, pp. 569–588, 2009.

[23] J. Rada-Vilela, "fuzzylite: a fuzzy logic control library in C++," 2013. [Online]. Available: http://www.fuzzylite.com

[24] M. Lozano, D. Molina, and F. Herrera, "Editorial Scalability of Evolutionary Algorithms and Other Metaheuristics for Large-scale Continuous Optimization Problems," *Soft Computing - A Fusion of Foundations, Methodologies and Applications*, pp. 1–3, 2010.