A Novel Diversity-Based Replacement Strategy for Evolutionary Algorithms

Carlos Segura, *Member, IEEE*, Carlos A. Coello Coello, *Fellow, IEEE*,
Eduardo Segredo, *Member, IEEE*, and Arturo Hernández Aguirre

Abstract—Premature convergence is one of the best-known drawbacks that affects the performance of evolutionary algorithms. An alternative for dealing with this problem is to explicitly try to maintain proper diversity. In this paper, a new replacement strategy that preserves useful diversity is presented. The novelty of our method is that it combines the idea of transforming a single-objective problem into a multiobjective one, by considering diversity as an explicit objective, with the idea of adapting the balance induced between exploration and exploitation to the various optimization stages. Specifically, in the initial phases, larger amounts of diversity are accepted. The diversity measure considered in this paper is based on calculating distances to the closest surviving individual. Analyses with a multimodal function better justify the design decisions and provide greater insight into the working operation of the proposal. Computational results with a packing problem that was proposed in a popular contest illustrate the usefulness of the proposal. The new method significantly improves on the best results known to date for this problem and compares favorably against a large number of state-of-the-art schemes.

Index Terms—Diversity preservation, exploitation, exploration, replacement strategy, survivor selection.

I. INTRODUCTION

EVOLUTIONARY algorithms (EAs) are one of the most promising approaches for several kinds of optimization, such as continuous [1] and multimodal [2] optimization. Since their inception, EAs have been widely used in many different areas, and nowadays are probably one of the most well-known metaheuristics [3]. In spite of their success, adapting EAs to new problems is not an easy task, because it usually involves many difficult design decisions [4]. As an example of the difficulties that arise in the design of EAs, it is known that one of the keys to success is to induce a proper balance between exploration and exploitation [5]. However, the implications of maintaining diversity in such a balance and the way in which exploration and exploitation are promoted in EAs are not always fully understood [6] and depend on the specific variant of EA applied. For instance, while in some schemes the mutation operator is in charge of promoting exploration [7], in other cases this task is assigned to the crossover operator [8].

Since the inception of EAs, premature convergence has been recognized as one of their recurrent drawbacks [6]. Premature convergence arises when all the population members are located in a reduced part of the search space—different from the optimal region—and the components selected do not allow escaping from such a region. Many schemes have been devised in an effort to avoid this drawback [9]. These studies have revealed that for most problems, maintaining a diverse population is a prerequisite for avoiding premature convergence [6]. However, if the population is too diverse, exploitation might be prevented, resulting in too slow a convergence and in poor-quality solutions. For this reason, Mahfoud [10] used the concept of useful diversity to refer to those amounts of diversity that result in high-quality solutions.

Regarding the design of EAs, it is important to note that most initial EAs were generational approaches [11], i.e., in each generation the offspring unconditionally replaced the previous population regardless of fitness. In these initial schemes, also known as "reproduction with emphasis" schemes [12], the parent selection was in charge of biasing the search toward the most promising regions. As a result, many of the initial attempts to balance the exploration and exploitation of the search space were based on modifying the parent selection scheme [13]. Other alternatives included modifying the variation strategy [7] and/or the population model [14]. However, in most current EAs the reproduction with emphasis is replaced or at least combined with the "survival of the fittest" principle [12]. Specifically, these algorithms, instead of replacing the old population with the child population, use an additional selection stage—the replacement strategy or survivor selection—to decide which individuals survive to the next generation [15]. Our hypothesis is that by introducing a diversity preservation mechanism into the survivor selection stage,

a more proper balance between exploration and exploitation can be induced, with the long-term result being higher quality solutions. We base this on the fact that while the variation and parent selection stages make decisions that affect the current generation—creation of the offspring—the survivor selection mechanism makes decisions that might have a more drastic effect on the whole optimization process. Specifically, this mechanism is in charge of deciding which solutions survive to the next generation, so even if some improper individuals are generated by the parent selection and variation stages, this can be fixed—at least partially—by a properly designed replacement stage.

In the literature, several survivor selection schemes that take diversity into account have also been devised [16], [17]. The novelty of our scheme is the dynamic balance we achieve between exploration and exploitation by incorporating the stopping criterion as an input to the replacement strategy. Specifically, in the first stages of the optimization procedure we induce a larger exploration by diversifying the survivors, while in the last stages, we promote exploitation. Similar ideas have been adopted to maintain diversity by influencing other components [18]. However, to the best of the authors' knowledge, no survivor selection stage has incorporated the use of the stopping criterion to bias its decisions.

We selected a packing problem (2DPP) modeled on the competitions held at the 2008 Genetic and Evolutionary Computation Conference (GECCO) to illustrate the benefits of our newly designed approach.[1] This problem was tackled by several research groups, which is why we decided to use it as a benchmark problem. Moreover, it is interesting to note that the schemes that have provided the best results for this problem [19]–[21], and which will be extended in this paper [19], are memetic approaches. In memetic algorithms, premature convergence can be even more harmful than in other variants of EAs [22], [23]. Consequently, applying diversity management in the survivor selection stage seems even more important.

The rest of the paper is organized as follows. A discussion of the relevant background is given in Section II. Section III presents an analysis of some related approaches that justify the decisions made when designing the new scheme. In Section IV, the new diversity-based replacement scheme is described. The benchmark problem, as well as the basics of the memetic scheme that is expanded upon in this paper, are outlined in Section V. Section VI is devoted to presenting our experimental validation. Finally, our conclusions and some lines of future work are given in Section VII.

II. DIVERSITY PRESERVATION IN EVOLUTIONARY ALGORITHMS

Premature convergence is a well-known drawback of EAs, so a large number of techniques to deal with this problem have been devised [9]. Most of these techniques are based on

---

[1]The original website (http://www.sigevo.org/gecco-2008/competitions.html) is not being maintained. We have created a new website where the evaluator and instances can be downloaded (http://2dpp.cimat.mx).

directly or indirectly managing the diversity of the population [6]. These methodologies range from general techniques to problem-dependent heuristics. In this section, we review some of the most popular general techniques, which will be used to show the potential of the new method developed in this paper. For a broader review of diversity management techniques, the reader is referred to [6].

A. Classic Diversity Management Schemes

Since most initial EAs were generational schemes, several attempts to properly balance exploration and exploitation that do not affect the survivor selection mechanism have been developed. These methods are reviewed in this section. In generational EAs, the main source of selection pressure usually comes from the parent selection. For this reason, specially during the 90s, a large effort was dedicated to devising new parent selection schemes where the selection pressure might be controlled [15]. In addition, some schemes capable of dynamically adapting the balance between exploration and exploitation were also devised. However, some studies found that parent selectors were not able to maintain proper diversity *per se* [13] even if relatively large populations were used. Thus, these schemes have to be combined with other strategies to successfully preserve diversity.

The population model has also been studied with the aim of improving diversity preservation in EAs. EAs with structured populations—instead of panmictic schemes—have gained considerable popularity in recent years [24]. In these schemes, some recombination restrictions are imposed by taking into account the positions of the individuals in the population. Some of these schemes, such as the island-based model [25], were initially devised as a way of reducing the interactions between individuals with the aim of facilitating their parallelization. However, they have important effects on diversity [14], as a result of which they have also been used as a way of promoting exploration. Since these schemes do not explicitly try to maintain diversity, in the long term, diversity is usually highly reduced. Moreover, since several components that influence the loss of diversity have to be selected, controlling this reduction in diversity is not an easy task [26]. In order to alleviate this problem, some island-based models that try to explicitly maintain diversity have been devised [27]. However, they usually introduce many parameters and require several adjustments.

Schemes based on mating restrictions are similar to those described above in the sense that some interactions between individuals are avoided. However, this is not done based on the positions of the individuals in the population. Instead, the distance between individuals in genotypic or phenotypic space is normally used. In some cases, promoting mating between dissimilar individuals seems to be the most promising [12], while in other cases, the opposite is true [28]. A scheme that adapts the mating selection was recently proposed [29]. However, it resulted in a method with many dependencies on the problems to be solved. Note also that it is recognized that while some of these alternatives might delay convergence, this effect is not completely avoided. As a result,

they might introduce additional mechanisms. For instance, the cross generational elitist selection, heterogeneous recombination, cataclysmic mutation scheme (CHC) uses a highly disruptive crossover operator and detects convergence of the population to launch a highly disruptive mutation.

Another alternative resides in adapting the variation stage. For instance, some parameter control techniques have been devised that try to adapt the balance between exploration and exploitation by using different parameter values in different stages of the optimization process [30]. In other cases, a pool of operators with different capabilities are simultaneously considered [31]. In these schemes, diversity is not usually taken into account directly. Instead, it is indirectly managed by selecting different operators or parameter values, which can cause performance drawbacks in some cases [32]. Using diversity in the control loop [33], [34] is a very promising approach which requires further development. In order to adapt the variation stage, the complete history of the evolution might be used [35]. However, for very long executions this is usually not possible so other alternatives must be considered.

Finally, restarting schemes are also quite popular. In these schemes, instead of avoiding a fast convergence, this event is detected, and then all or part of the population is restarted. Several different ways of establishing the restarting points have been devised [36]. These schemes are easy to implement and have yielded significant improvements in some cases [37]. In addition, since they are based on recovering diversity, they can be easily combined with schemes that attempt to maintain diversity. A particular case of this type of schemes is the sawtooth genetic algorithm (saw-tooth GA) [37]. This scheme uses a variable population size and periodic partial reinitialization of the population in the shape of a saw-tooth function. In order to configure such a function two parameters are required: 1) the period (P) and 2) the amplitude (D).

B. Replacement Schemes Based on Diversity

Some diversity-preservation techniques that rely on modifying the replacement phase have also been devised. The basic principle of these schemes is that by diversifying the survivors, more exploration can be induced. The reason for this is twofold. First, if the diversity of a given population is large, it means that several regions of the search space are maintained. Second, most crossover operators tend to be more explorative when distant individuals are involved [38]. For instance, in geometric crossovers [39]—such as the one used in this paper—offspring are in the segment between their parents under a given metric, meaning that the distances between parents and their offspring tend to be larger when distant parents are involved. This means that when the population maintains distant parents and this kind of crossover is used, the offspring can explore other regions and, when coupled with proper mutation and replacement, a larger diversity might be induced. Note that with the use of nongeometric crossovers, such as the one defined in [40], a larger diversity might be induced by the crossover operator. However, in this case, controlling the diversity of the population seems more complex. For instance, in such a crossover a bit-flip mutation is applied

to the genes where both parents contain the same allele. This means that the degree of exploration increases when similar individuals are used. Thus, while maintaining distant individuals is appropriate for exploring different regions, the effect on the diversity induced by the variation scheme is the opposite. Moreover, defining nondestructive nongeometric crossover for the 2DPP is not easy, so we decided to apply a geometric crossover. Note that local search and mutation are also used in our method. When these components are applied, alleles not present in an individual might be recovered.

One of the first studies on using the replacement phase to control the diversity resulted in Cavicchio's preselection scheme [41]. Subsequently, preselection was extended to create crowding [42], which has been quite popular in recent years [16], [43]. The basic principle of crowding is to force new individuals entering the population to replace similar individuals. Some of the most popular crowding methods are as follows.

1) In Mahfoud's deterministic crowding [43] (DETCR) each pair of parents and their corresponding offspring are paired by minimizing the sum of the distances between them. Then, for each pair, the offspring survives if it is at least as good as the parent.

2) Probabilistic crowding [44] (POBCR) is similar to DETCR, but it uses a nondeterministic rule to establish the winner. Specifically, the probability of survival of each competitor is proportional to its fitness value. A scaled variant of POBCR (SPOBCR) was also devised [45].

3) In the recently proposed adaptive generalized crowding [16] (AGCR), the selection pressure of the replacement rule depends on the value of a parameter. Two different ways of adapting such a parameter were proposed: a) a self-adaptive and b) an adaptive scheme. In the adaptive scheme, which is the one applied in this paper, the adaptation is done based on the entropy of the population.

4) In restricted tournament selection [46] (RTS), after a new individual (C) is created, CF individuals from the current population are randomly selected. Then, C and its most similar individual—from those in the selected set—compete for a place in the population using a traditional binary tournament. The creation of individuals is done as in steady-state schemes.

Several other replacement strategies that promote diversity have been proposed. One of the most popular is probably the clearing [47] strategy (CLR). Clearing can be regarded as an extension of fitness sharing. However, while in fitness sharing the fitness of each individual is normalized depending on the number of individuals in its region—defined via the parameter σ—, in the clearing procedure the resources of a niche are attributed to the best W elements in each niche. Moreover, the winners of each niche are automatically preserved by copying them to the next population. Note that if too many niches are detected, this might lead to a large immobilization of the population. As a result, Petrowski [47] proposed preserving only the winners with a fitness greater than the mean, which is the alternative used in our paper.

In some methods, maximizing diversity is considered as an objective that is combined with the original objective to calculate the fitness of each individual. However, since the two measures are not entirely compatible, such a combination is complex and problem-dependent. In order to alleviate this problem, other ways of combining them have been devised. For instance, in COMB [17]—abreviation for combination—the individuals are sorted by the original cost and by their contribution to diversity. Then, the rankings of the individuals are combined to generate the fitness value using two parameters (N_{Close} and N_{Elite}). In each step of the replacement phase, the individual with the lowest fitness is erased and the ranks are recalculated. Since this scheme uses ranks instead of the original function value, it results in a more robust scheme. The main drawback is that two new parameters must be set, and due to the way in which individuals are scored, some individuals that contribute practically nothing to diversity might be preserved. Another alternative is the method devised in [48] (Contribution of Diversity/Replace Worst (CD/RW)). In the CD/RW method, a new individual enters the population by replacing another one that is worse both in quality and diversity contribution. If such an individual is not found, a replace-worst strategy (RW) is applied, i.e., the worst individual in the population is erased if it is worse than the newly generated individual. The main drawback is that RW does not take diversity into consideration, so if the method relies too much on this scheme, premature convergence might appear very quickly.

Finally, another quite popular alternative is to explicitly consider diversity as an objective and apply a multi-objective optimization scheme [49], [50]. These kinds of schemes are usually referred to as diversity-based multiobjective EAs (MOEAs) [51]. In these approaches, a measure of population diversity is not required. Instead, the auxiliary objective must be a measure of the diversity introduced by the individual in the population. Several different ways of calculating the auxiliary objective have been proposed [51]. One of the most popular is probably the distance to the closest neighbor (DCN) metric [49]. In DCN, the auxiliary objective of a given individual is calculated as its distance to the closest member in the population. In addition, it was shown that calculating DCN by taking into account only the members that have already been selected to survive is preferred, at least for continuous optimization problems [52]. Some variants of these schemes have been applied to the benchmark problem that is addressed in this paper [19], [53]. Finally, we would like to note that the concept of multiobjectivization [54] is highly related to diversity-based MOEAs. However, in the case of multiobjectivization the calculation of each individual's objectives does not depend on the content of the population, which represents a substantial difference in the design and analysis of this kind of approach. Readers are referred to [51] and [55] for a more extensive survey on both kinds of schemes.

III. MATHEMATICAL ANALYSIS OF RELATED APPROACHES

In this paper, an extension of the replacement strategy presented in [52] (MULTI) is devised. Initially, we attempted to

---

Algorithm 1 MULTI Survivor Selection Scheme

1: CurrentMembers = Population ∪ Offspring
2: Best = Individual with best $f(x)$ in CurrentMembers
3: NewPop = { Best }
4: CurrentMembers = CurrentMembers - { Best }
5: **while** (|NewPop| < N) **do**
6:     Calculate DCN of CurrentMembers, considering as reference NewPop
7:     ND = Non-dominated individuals of CurrentMembers (without repetitions)
8:     Selected = Randomly select an individual from ND
9:     NewPop = NewPop ∪ Selected
10:     CurrentMembers = CurrentMembers - {Selected}
11: **end while**
12: Population = NewPop

---

use this method directly to deal with the packing problem we consider here. However, the results were unsatisfactory. Our initial analyses showed that even with the incorporation of such a replacement strategy, the diversity was reduced too quickly and a proper balance between exploration and exploitation was not achieved. The difference with previous successful applications of related schemes is that, in this case, a memetic approach is used and that much longer executions are required to attain high-quality solutions. In addition, other mechanisms for promoting diversity were not incorporated. These differences have serious implications for the optimization procedure. For instance, while in [52] the variation scheme does not usually generate clones, when a local search is applied, clones are more likely to appear. Moreover, given that much longer executions are run, the balance required between exploration and exploitation might differ. In this section, we present an analysis of this replacement strategy, which justifies the design decisions made in this paper.

The replacement strategy MULTI (Algorithm 1) operates as follows. First, the population of the previous generation and the offspring are joined in a temporary set. Then, the best individual, i.e., the one with the highest original objective function value—for a maximization problem—is selected to form part of the new population. Then, until the new population is filled with N individuals, the following steps are executed. First, the DCN objective is calculated. The calculation considers the currently selected individuals as the reference, i.e., for each pending individual, the distance to the nearest individual previously selected is taken into account. Then, considering the individuals that have not been selected, the nondominated front is calculated. This front is computed as a set with no repetitions, i.e., if a nondominated member appears several times in the population, it is only included once in the front. While this does not prevent the appearance of clones, it does reduce their growth rates. Finally, a nondominated individual is randomly selected to survive. Note that in each step, the probability of selecting any nondominated individual is the same because all of them are equally valid when both objectives are considered simultaneously. However, once that an individual is selected, the DCN values are recalculated, meaning that the selected region is penalized in subsequent selections. While this action promotes a larger diversity than other standard replacement strategies, the ratio between exploration and exploitation does
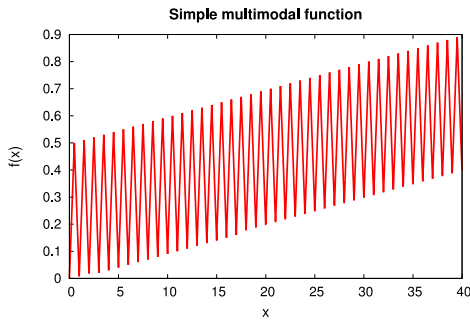
Fig. 1.   Simple multimodal function of one variable used to analyze diversity.

Fig. 2.   Evolution of the mean of the number of basins of attraction covered in the population with the MULTI replacement scheme.

not depend on the stopping criterion, so there is no proper adaptation to the requirement of the different stages of the optimization.

Even though some related methods have been successfully applied to complex problems [19], [49], to the best of our knowledge, the diversity induced by these kinds of methods has not been previously analyzed. Thus, in order to have a better understanding of the internal operation of this replacement scheme, we present such an analysis here. A common metric applied for analyzing the properties of diversity-preservation schemes is the takeover time [13], which is defined as the number of generations required for convergence into a single solution when no variation is accomplished. In our case, due to the way in which replacement is done, complete convergence is not expected to appear. For instance, consider an extreme case where in a population of N individuals, $N - 1$ are clones of the best individual (B) and there is one individual (D) that differs. Also, assume that all the offspring are clones of B. In such a case, in each step, the nondominated set is formed by D and one copy of B, on the condition that D is not selected in any step—if D is selected at some step, complete convergence would not appear. Thus, in each of the $N - 1$ selections carried out by the replacement strategy, B is selected with probability $(1/2)$. This means that an individual B is not selected in the whole process with probability $(1/2)^{N-1}$, which can be considered negligible for typical values of N, so complete convergence does not appear.

Since calculating the takeover time makes no sense in this case, a variant of this is studied for the minimization of the single continuous mathematical function given in (1) (see Fig. 1). This function is not complex, thus making it easier to study the internal operation of MULTI. This function consists of 41 basins of attraction, so it seems interesting to calculate the number of basins of attraction where there is at least one individual. Given the complex interactions that appear in the model, this paper was carried out experimentally

$$f(x) = \min(\lceil x \rceil - x, x - \lfloor x \rfloor) + \lfloor x \rfloor \times 1e^{-2}, \quad x \in [0, 40]. \quad (1)$$

Fig. 2 shows the evolution of the number of basins of attraction covered by the population. It was calculated using the mean of 50 executions, where the parents were selected using binary tournaments, the initial population was randomly generated, the perturbation strength was a random number in $[-0.2, 0.2]$, no crossover was considered and different population sizes (N) were tested. Specifically, N was set to
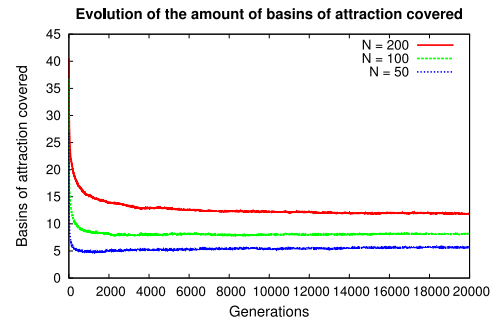
50, 100, and 200. The stopping criterion was set to 20 000 generations. We can see that in any of the cases, the number of basins of attraction covered becomes stable without complete convergence, i.e., several basins of attraction are explored even in the final stages. Moreover, as N is increased, the number of basins of attraction maintained also grows. Thus, the population size is an effective way of increasing the exploration capabilities. Note also, however, that in every case there is a rapid decrease in the number of basins of attraction maintained in the population in the first phases of the optimization before remaining practically intact until the end of the optimization. In fact, in any of the cases tested, more than half of the basins of attraction are lost early on in the optimization process. As a result, the scheme is not really maintaining a large level of exploration in the initial phases, and in some executions the basin of attraction of the global optimum is not covered. In practical cases, with the presence of crossover and larger search spaces, the loss of diversity might be slower. In any event, the following disadvantages are clear.
  1) The reduction in diversity in the initial phases is too large.
  2) The user cannot control the reduction in detail (it can only be partially controlled with N), so while this process might be suitable for certain stopping criteria, in other cases the balance induced between exploration and exploitation might not be convenient.

IV. OUR PROPOSAL

Our proposal builds on the multiobjective replacement strategy presented before (Algorithm 1), and modifies it with the aim of avoiding the drawbacks already discussed. The main advantage of the new proposal is that the balance between exploration and exploitation is automatically adjusted based on the given stopping criterion. Thus, the stopping criterion, as well as the elapsed time or the evaluations already executed, are used as inputs to the replacement strategy, which is one of the novelties of the new design. In this way, for shorter stopping criteria the method induces a faster reduction in diversity than for longer stopping criteria.

One of the reasons for the drawbacks previously discussed is that the importance assigned to the original objective and diversity objective remains intact throughout the run. In such an approach, more exploitation is done as the
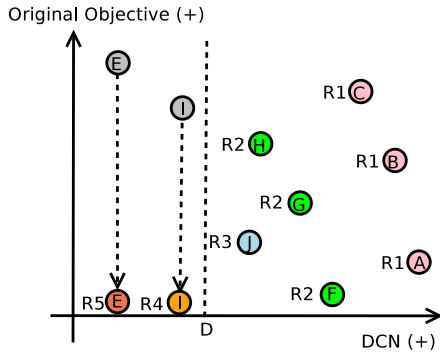
Fig. 3.   Effect of the penalty approach with distance D.

Algorithm 2 MULTI_DYNAMIC Survivor Selection Scheme

1: CurrentMembers = Population ∪ Offspring
2: Best = Individual with best $f(x)$ in CurrentMembers
3: NewPop = { Best }
4: CurrentMembers = CurrentMembers - { Best }
5: **while** (|NewPop| < N) **do**
6:     Calculate DCN of CurrentMembers, considering as reference NewPop
7:     $D = D_I - D_I * \frac{T_{Elapsed}}{T_{End}}$
8:     Penalize(CurrentMembers, D)
9:     ND = Non-dominated individuals of CurrentMembers (without repetitions)
10:     Selected = Randomly select an individual from ND
11:     NewPop = NewPop ∪ Selected
12:     CurrentMembers = CurrentMembers - {Selected}
13: **end while**
14: Population = NewPop

execution advances. However, this is only due to the selection pressure induced by the parent selection, which tends to gather more individuals in the most promising zones, and to the fact that crossover is more exploitative when acting on nearby individuals. As a result, this balance can barely be controlled. However, since the origins of EAs several authors have claimed that adapting this balance to the different stages of the optimization is quite important [38]. Using weighted fitness functions might be an alternative for balancing the priority of the objectives. However, since the scale and form of the original objective depend on the problem, generalizing this approach is truly complicated. Instead, we have defined a dynamic penalty approach that is independent of the scale of the original objective.

One of the basic principles behind the development of the replacement strategy devised in this paper (MULTI_DYNAMIC) is that individuals that contribute too little to diversity—the contribution is measured with the DCN value—should not be accepted regardless of their original objective value. In our approach, individuals that contribute too little are penalized by setting their original objective value to a very low quality value. For a non-negative maximization function (as is the case of 2DPP), the value 0 might be used. This is illustrated in Fig. 3. In this figure, the value D represents the minimum DCN required to avoid being penalized. As the figure shows, any individual whose DCN value is lower than this threshold value is penalized. As a result, the nondomination rank (shown at the left of each individual) of the penalized individuals might be increased. Obviously, with this approach, the penalized individuals will not belong to the nondominated front, unless every pending individual has been penalized.

While the above approach is quite logical, one of the key choices is how to evaluate whether an individual contributes enough or not, i.e., how to set the value D. The value of D should depend on the optimization stage. Specifically, this value should be reduced as the stopping criterion is approached. In our scheme, an initial D_I value must be set. Then, a linear reduction of D is done. The reduction is calculated in such a way that by the end of the execution, the resulting value is 0. In this paper, the stopping criterion is set by time. Thus, if T_{End} is the number of seconds allocated to the run and $T_{Elapsed}$ the elapsed time, D can be calculated as

$D = D_I - D_I*(T_{Elapsed}/T_{End})$. Some preliminary tests that took into account different ways of updating D were also developed. Specifically, we defined a pairwise function that allows us to specify the time when D starts to decrease, as well as the time when it reaches the zero value. In addition, we tested reductions that were faster as well as slower than the linear one. While these modifications provided some benefits in a few specific cases, those benefits were not very significant and depended on the particular instances, so in this paper we use the linear reduction. Some preliminary tests were also carried out that used the number of generations to set the stopping criterion. In these cases, instead of $T_{Elapsed}$ and T_{End}, the number of generations already evolved, as well as the maximum number of generations, are used to compute D. The conclusions that could be drawn when using generations instead of time as the stopping criterion were similar to those obtained in this paper. However, the algorithms that maintained higher levels of diversity incurred more expensive local searches, meaning that much more time was required to evolve the same number of generations. Consequently, the results presented in this paper rely on time as the stopping criterion. In the case of using generations as the stopping criterion, the relative order of some of the schemes tested changes, but the benefits of the new proposal remain intact. Algorithm 2 shows the pseudocode of MULTI_DYNAMIC. We would like to remark that the only stage that uses multiobjective concepts is the replacement strategy. Thus, the remaining parts of the EA are kept as a single-objective EA.

Regarding the parameterization of our proposal, we would like to note that while the setting of D_I obviously depends on the problem, the main advantage of MULTI_DYNAMIC is that DCN is defined in the space of the variables and not in the space of the objectives. The relationships among the candidate solutions in the space of the variables are usually easier to analyze than those that arise in the space of the objectives because the former space is known by the user. For instance, it is usually quite easy to calculate the maximum distance between any two solutions or the search space size. However, in the space of the objectives this is usually not possible. Obviously, there are some optimization problems where meaningful distances are not easily obtained. For instance, in evolutionary robotics

simple distances among synaptic weights are not adequate and more complex definitions are used [51]. In any case, in many problems, simple and meaningful distances can be devised. Moreover, as shown in the experimental validation attached as supplementary material, at least for the 2DPP, the scheme is quite robust in the sense that a large range of D_I values provides high-quality solutions.

Finally, it is also interesting to note that, similarly to [52], we also carried out some experiments by penalizing the individuals that obtained the worst values in the original objective. The key idea was to induce a large diversity but only between promising individuals. This accelerated the achievement of good-quality solutions, but in the long term there were no benefits. In fact, some cases with improper parameterization resulted in solutions of lower quality. Since this would require the use of additional parameters and the benefits were not very significant, we did not incorporate it in our final designs. In our opinion, incorporating these schemes in only some phases of the optimization might yield better results. However, such an analysis is beyond the scope of this paper.

A. Reduction of Diversity

In the previous sections, we identified some of the drawbacks of earlier schemes by considering the simple function given in (1). As in previous cases, for our new model, analyzing the takeover time makes no sense. However, it is interesting to study the dependency between the parameters of the new approach and the trend in the number of basins of attraction covered by the population. We considered the same variation operator and parent selection as in previous cases. Thus, the only modification was the replacement strategy. In order to study the effects of the parameterization, three different population sizes (50, 100, and 200) were used. Moreover, for each case, two different values of D_I were taken into account: $(40/N)$ and $(40/2 \times N)$. These values were selected considering the range of admissible values for the variable x. In the first case, in the initial stages of the optimization there is no penalty only if the individuals are perfectly distributed. This thus promotes more exploration. In the second case, the balance toward exploration is reduced.

Fig. 4 shows the evolution of the number of basins of attraction (mean of 50 executions) covered for each case. We can clearly see that as the final stages are approached, the number of occupied regions diminishes. However, the reduction can now be controlled and depends on the number of generations allocated to the execution. Two important effects are illustrated in this figure. First, the setting of N is particularly important to the amount of exploration induced at the end of the optimization, i.e., for a fixed value of N the number of basins of attraction covered at the end of the optimization is not heavily dependent on D_I. The reason is that, regardless of D_I, at the end of the optimization, D is set to 0. The value of D_I is useful for setting the initial balance between exploration and exploitation, and the way in which this balance changes. Specifically, larger D_I values imply that in the initial stages, more exploration is induced and diversity is reduced
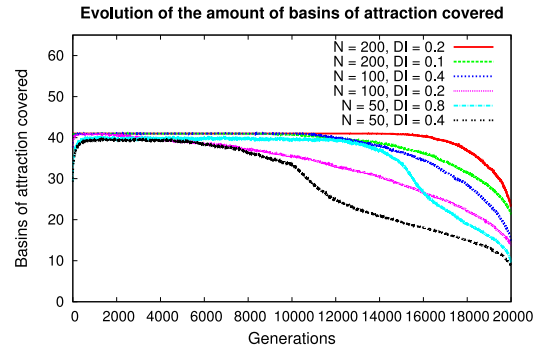

Fig. 4. Evolution of the mean of the number of basins of attraction covered in the population with the MULTI_DYNAMIC replacement scheme.

later but more quickly. Finally, it is also interesting to note the behavior of MULTI_DYNAMIC when $N = 50$ is used. In this case, in the initial phases the number of basins of attraction covered increases. The reason is that for $N = 50$, the random initialization might not cover every basin of attraction and that with large values of D, in some of the selections carried out by MULTI_DYNAMIC, all of the pending individuals are penalized. In these cases, the only objective taken into account is DCN, so the scheme is very explorative and occupies the under-explored regions. The advantage of maintaining a larger exploration in the initial phases is that each basin of attraction can be better explored, so the individuals are placed near the local optima. This means that in the long term, additional information is used to discard individuals, meaning that better decisions can be made. In fact, 10 000 executions of the scheme with the MULTI_DYNAMIC replacement were run, taking into account the aforementioned population sizes, and in all of them the basin of attraction of the best solution was covered. In contrast, when the original MULTI scheme was applied, this basin of attraction was lost multiple times.

V. Two-Dimensional Packing Problem

A. Formal Definition

The problem used as a benchmark in this paper is the packing problem that was proposed in the GECCO 2008 competition session. Problem instances are described by the following data.
1) The sizes of a rectangular grid: X and Y.
2) The maximum number which can be assigned to a grid position: M. The value assigned to each grid location is an integer in the range $[0, M]$.
3) The score or value associated with the appearance of each pair (a, b), where $a, b \in [0, M] : v(a, b)$.

A candidate solution is obtained by assigning a number to each position on the grid. Thus, the search space consists of $(M + 1)^{X \cdot Y}$ candidate solutions. The objective of the problem is to pack a grid so that the sum of the point scores for every pair of adjacent numbers is maximized. Two positions are considered to be adjacent if they are neighbors in the same row, column, or diagonal of the grid. Once a particular pair is collected, it cannot be collected a second time in the

same grid. Mathematically, the objective is to find the grid G which maximizes the objective function f

$$f = \sum_{a=0}^{M} \sum_{b=0}^{M} v_2(a, b) \qquad (2)$$

where

$$v_2(a, b) = \begin{cases} 0 & \text{if } (a, b) \text{ are not adjacent in } G \\ v(a, b) & \text{if } (a, b) \text{ are adjacent in } G. \end{cases} \qquad (3)$$

In order to better illustrate the benefits of the schemes devised in this paper, an instance generator was developed.[2] The generator allows for the creation of instances of different complexities. Specifically, the grid size (X and Y), the maximum value (M) and a ratio (R) must be specified. Each entry $v(a, b)$ with $a, b \in [0, M]$ is set to a random number between 0 and 999 999 with probability R. Otherwise, it is set to 0. Thus, the complexity of the instances can be increased by using larger values of X, Y, M, or R.

B. Memetic Approach

A memetic scheme (Algorithm 3) that follows the Lamarckian approach [56] is used in this paper. Since in the Lamarckian model modifications made by the individual learning procedure (local search in our case) are placed back into the individuals, the burden associated with this kind of scheme is usually shorter than with Baldwinian schemes, which is why this model was selected. Specifically, note that in our scheme the local search is stopped when a local optimum is reached. In the cases where the modifications are placed back into the individuals, the new individuals created by variation are usually closer to a local optimum that when this information is not placed back, which is why using the Lamarckian model is faster in our case. The scheme considers the local search and variation operators used in [19], applies a single-objective parent selection based on binary tournaments, and takes into account the MULTI_DYNAMIC survivor selection. In our proposal, the local search is applied to all the offspring. Note that in some cases, schemes for balancing between genetic and local searches were used [57] and it is known that several other design decisions might affect performance [58]. For instance, local search might be applied only to a subset of individuals selected according to some features [59], e.g., the best ones. Applying local search only to the best individuals is, in some ways, contrary to the principles of our scheme. Note that our new proposal dynamically changes the balance between exploration and exploitation. Thus, in each stage, all the members of the population are considered promising. For this reason, it is important to intensify the search in all the regions maintained in the population. This is exacerbated by the fact that in this problem, crossover can be quite disruptive, but with the application of local search, drastic improvements can be obtained. Specifically, while many good pairs are maintained in the candidate solutions, some improper pairs are also generated in the line selected by the crossover (see the crossover definition for details). As a result, applying a local search to all the new

[2]It can be downloaded from http://2dpp.cimat.mx

Algorithm 3 Diversity-Based Lamarckian Memetic Algorithm

1: **Initialization**: Generate an initial population P_0 with N individuals. Assign $t = 0$.
2: **Local Search**: Perform local search for every individual in the population.
3: $D_I = R_I \times G$
4: **while** (not stopping criterion) **do**
5:     **Evaluation**: Evaluate all individuals in the population.
6:     **Mating selection**: Perform binary tournament selection on P_t in order to fill the mating pool.
7:     **Variation**: Apply genetic operators to the mating pool to create a child population CP.
8:     **Local Search**: Perform a local search for every individual in the offspring.
9:     **Survivor selection**: Combine P_t and CP, and apply the MULTI_DYNAMIC replacement scheme to create P_{t+1}.
10:     $t = t + 1$
11: **end while**

---

members seems quite important; otherwise, the new individuals might quickly be selected for removal. In order to test these hypotheses, we used ten different instances and tested our models by applying local search only to a subset of the offspring. Specifically, two different alternatives were tested. In the first one, the best 10% individuals were selected, while in the second case, the same number of individuals were selected but randomly. When comparing the results with the case where all the offspring undergo a local search, we noted that in the case of selecting the best individuals, the results obtained were inferior in every case and the differences were statistically significant. Selecting individuals randomly was more encouraging. Specifically, in four of the instances the results were similar to those obtained by applying local search to all the offspring, and in one case it obtained better results. However, in the remaining instances, applying local search to all the offspring was preferred, so this is the alternative used in this paper.

In our proposal, Hamming distances are used to calculate the DCN value, meaning the maximum distance that can appear between any two solutions is equal to the number of genes ($G = X \times Y$). Note also that the number of alleles in each gene is $M + 1$. Thus, if the population size is lower than $M + 2$, the distances between any two individuals in the population might be as large as G. As a result, it might make sense to set the value of D_I to G. However, we expect that this would induce an exceedingly large explorative behavior for some instances. In any case, relating the value of D_I to G seems appropriate. Particularly, in this paper the value of D_I is set to a percentage ($R_I \times 100$) of G, and the implication of using different R_I values is analyzed.

A brief description of the variation operators and local search is attached. For a more comprehensive description, readers are referred to [19].

The local search procedure, which was initially proposed in [20], is a single-objective stochastic hill-climbing approach. In this scheme, the order in which neighbors are analyzed is determined randomly. The local search moves to the first newly generated neighbor that improves the current solution, and it stops when none of the neighbors improves the

current solution. The definition of neighborhood applied considers a new neighbor for each pair of adjacent grid positions (i, j) and (k, l). Each neighbor is constituted by assigning the best possible values to the positions (i, j) and (k, l) while leaving intact the assignments in all other grid locations. In order to assign the best values to both locations a pruning-based mechanism that speeds up the generation of neighbors is used.

The variation stage is based on the application of crossover and mutation [60]. Crossover is applied with probability p_c and two different parameters ($\min\_p_m$ and $\max\_p_m$) are used to control the mutation. The crossover operator is the 2-D substring crossover (SSX) [61]. It is an extension of one-point crossover for 2-D chromosomes. The mutation operator applied is the uniform mutation with domain information (UMD). Prior to the mutation, a random number (ap_m) between $\min\_p_m$ and $\max\_p_m$ is generated. Then, each gene is mutated with a probability ap_m. In order to make new assignments to the gene, a random value is selected from among those that produce a nonzero increase in the objective value. If such a value does not exist, a random value between 0 and M is used.

VI. Experimental Evaluation

In this section, the experiments conducted with our new scheme are described. The optimization schemes were implemented using metaheuristic-based extensible tool for cooperative optimization (METCO) [60]. The analyses were performed with ten different instances of the 2DPP. One of the instances (GECCO) was the one used during the GECCO 2008 contest. The remaining ones were created with the newly designed instance generator. In order to test cases of different complexities, instances with different features were used. Specifically, three small, three medium, and three large instances were created. In the small instances, X and Y were set to 10. In the medium instances, they were set to 15. Finally, in the large instances, they were set to 20, which is the same size as the one used in the GECCO instance. Each group consists of three instances. The M value was set to 400 in every case and in each group three different R values were considered: 0.05, 0.10, and 0.15. The tags used for denoting the instances in this paper are "X_Y_M_R_S", where the symbol S denotes the seed used for the random number generator and the remaining symbols have already been introduced.

Since stochastic algorithms were considered in this paper, each execution was repeated 30 times and comparisons were carried out by applying a set of statistical tests. A similar guideline as the one applied in [62] was used. Specifically, the following tests were applied, assuming a significance level of 5%. First, a Shapiro–Wilk test was performed to check whether or not the values of the results followed a Gaussian distribution. If so, the Levene test was used to check for the homogeneity of the variances. If samples had equal variance, an ANOVA test was done; if not, a Welch test was performed. For non-Gaussian distributions, the nonparametric Kruskal–Wallis test was used to test whether samples are drawn from the same distribution. In this paper, the sentence "algorithm A is better than algorithm B" means that

the differences between them are statistically significant, and that the mean and median obtained by A are higher than the mean and median achieved by B. In order to show the benefits of the new model and to better understand its internal operation, two different sets of experiments were performed.

A. Comparison With Other Schemes

In order to show the validity of the new scheme, it is very important to do an extensive comparison with other schemes developed in the specialized literature. We selected a large set of schemes, including several recent as well as more mature methods. Specifically, the following schemes were used: CD/RW, CLR, DETCR, SPOBCR, AGCR, RTS, COMB, RW, CHC, and saw-tooth GA. In addition, a generational scheme with elitism (GEN_ELIT) was used. In this scheme, $N - 1$ offspring are created and all of them, as well as the best individual of the previous generation survive. The variation scheme was the same in every case except in CHC, and it was based on the crossover and mutation operators already described. As in previous studies [19], the parameters p_c, $\min\_p_m$, and $\max\_p_m$ were set to 1, 0.1 and 0.15, respectively, while the population size was set to 50. In CHC, the use of half-uniform crossover and no mutation (except for reinitialization) were taken into account. For the reinitialization, the probability of mutation was set to 0.35. In DETCR and SPOBCR, the members of the population were paired randomly, so each member takes part in the creation of one child. This is a requisite because in these schemes, the parents and their corresponding children are paired to compete for a location in the new population. In the case of CHC, as in its original version, they were also paired randomly. In the remaining schemes, parents were selected through binary tournaments. In the case of CLR, RTS, COMB, AGCR, and saw-tooth GA additional parameters must be set. These parameters were set experimentally and taking into account the recommendations given by their corresponding authors. In CLR, σ was set to $0.20 \times X \times Y$, and results for three different values of W (1, 2, and 5) are shown. The tag CLR_W is used to denote each configuration. In RTS, results for five different values of CF (2, 5, 10, 25, and 50) are presented. The tag RTS_CF is used to denote these schemes. In COMB, N_{Close} and N_{Elite} were set to 3 and 8 as recommended by the authors (other parameters were unsuccessfully tested). In AGCR, results for two different scale factors ϕ (0.25 and 0.75) are shown. The tag AGCR_ϕ is used for these models. Finally, in saw-tooth GA, D, and P were set to 45 and 50, respectively. This was the best configuration found among the 12 different parameterizations that were checked. These configurations combined the values $D = \{45, 30, 15\}$ with $P = \{30, 50, 100, 200\}$.

The previous schemes, as well as the new MULTI_DYNAMIC model, were executed using a stopping criterion of 24 h. In this first experiment, the parameter R_I of MULTI_DYNAMIC was set to 0.5, i.e., initially the individuals are penalized if more than half of its genes are similar to those of an already accepted individual. Additionally, MULTI_DYNAMIC with R_I set to 0 was also executed. Note that with this parameterization, this scheme behaves like the MULTI scheme because penalties are not

TABLE I
STATISTICAL COMPARISON OF EVERY CONSIDERED SCHEME IN 24 h

	SMALL			MEDIUM			LARGE			GECCO			TOTAL		
	↑	↓	↔	↑	↓	↔	↑	↓	↔	↑	↓	↔	↑	↓	Score
MULTI_DYNAMIC	57	0	0	57	0	0	57	0	0	19	0	0	190	0	190
MULTI	47	4	6	48	5	4	51	6	0	17	2	0	163	17	146
CHC	39	12	6	47	7	3	54	3	0	18	1	0	158	23	135
COMB	46	4	7	39	12	6	39	9	9	12	4	3	136	29	107
CLR_1	33	14	10	41	12	4	40	9	8	15	3	1	129	38	91
CLR_2	27	18	12	35	17	5	39	9	9	12	4	3	113	48	65
GEN_ELIT	15	32	10	35	17	5	37	10	10	12	3	4	99	62	37
SPOBCR	40	9	8	52	3	2	8	48	1	3	16	0	103	76	27
CLR_5	21	22	14	27	24	6	30	20	7	11	7	1	89	73	16
CD/RW	3	43	11	14	25	18	29	20	8	11	4	4	57	92	-35
SawTooth-GA	16	27	14	15	25	17	15	37	5	5	9	5	51	98	-47
RTS_10	10	35	12	11	27	19	21	27	9	5	9	5	47	98	-51
RTS_25	16	29	12	14	26	17	15	38	4	5	9	5	50	102	-52
AGCR_0.25	50	3	4	3	37	17	3	54	0	1	18	0	57	112	-55
RTS_50	27	19	11	8	33	16	12	45	0	4	15	0	51	112	-61
RTS_5	3	43	11	8	33	16	22	23	12	5	9	5	38	108	-70
RTS_2	3	44	10	8	33	16	20	24	13	5	9	5	36	110	-74
RW	3	45	9	8	31	18	19	28	10	5	9	5	35	113	-78
DETCR	23	19	15	5	51	1	6	50	1	2	17	0	36	137	-101
AGCR_0.75	0	57	0	0	57	0	0	57	0	0	19	0	0	190	-190

used at all. As we mentioned before, we also tested this scheme by incorporating the penalties proposed in [52]. However, the application of these components involved the use of additional parameters that have to be set by the user, and no significant benefits were obtained. As a result, the integration of these components with the newly designed elements is left for future work. In order to obtain an overall ranking of the different approaches tested, pairwise statistical comparisons between the 20 configurations were carried out. Since the benchmark set comprises ten instances, 190 statistical tests were done for each scheme. Table I shows, grouped by category, the results of these statistical tests. For each category, columns with the symbol ↑ show the number of cases where the model listed in each row is statistically better. The number of cases where it is worse is shown in the column with the symbol ↓. Finally, the number of cases where the differences are not statistically significant are shown in the column with the symbol ↔. In addition, a score is assigned to each model. This score is equal to the number of cases where the model was superior minus the number of cases where the model was inferior. The models are sorted by taking this score into account. The statistical tests clearly show the superiority of the MULTI_DYNAMIC model. In fact, since its score is 190, it means that it was superior to all the remaining models in every instance. Among the remaining models, the behavior depends on the category of the instances. For example, for small instances it is quite important to preserve diversity. For this reason, the approaches that place a larger emphasis on exploitation, such as RTS with low CF values or RW, are the worst-behaved schemes. However, in the larger instances, it is also quite important to promote exploitation, so the most explorative schemes, like DETCR or RTS with high CF values, are not proper approaches. Note that the MULTI and CHC schemes also yielded quite promising behaviors. However, both of them failed in several instances and did not obtain results as good as MULTI_DYNAMIC in any of the instances.

In order to better understand the reasons for the vast superiority of MULTI_DYNAMIC, it is important to analyze how diversity is managed in the different schemes. Entropy [16] is a popular diversity metric that can be used for this purpose. Note that, in our opinion, using entropy alone to control diversity is not adequate. For instance, in a problem where the representation is done with binary strings, a population where half of the members are individuals containing 0 in every gene and half of the members are individuals containing 1 in every gene would have maximum entropy. In any case, a low entropy value implies low diversity and since in our scheme we ensure a minimal distance among individuals, analyzing how the entropy evolves provides valuable information. Fig. 5 shows the evolution of the mean of the entropy for all the models considered in the GECCO instance. In the case of CHC and saw-tooth GA, only one execution was used to generate this trend. Specifically, the executions were sorted taking into account the fitness at the end of the runs and the one that obtained the fifteenth position was used. The reason is that, since the exact times where the restarts are triggered vary for each run, showing the mean of several executions is less meaningful. Note that the only model where there is a slow but continuous decrease in the entropy is MULTI_DYNAMIC. Due to the way in which the crossover operates, this means that the balance between exploration and exploitation changes gradually in our case. In the remaining models, given that the stopping criterion is not taken into account, achieving such a gradual reduction in diversity is too complex.

Previous analyses show the clear superiority of the MULTI_DYNAMIC scheme. In order to better illustrate the advantages of our proposal, Fig. 6 shows the evolution of the mean of the fitness for the different schemes in the GECCO instance. It is evident how, due the explicit control of the diversity, the convergence in MULTI_DYNAMIC is much slower than in many other schemes. In fact, even after 10 h, the fitness values reported by MULTI_DYNAMIC are not as
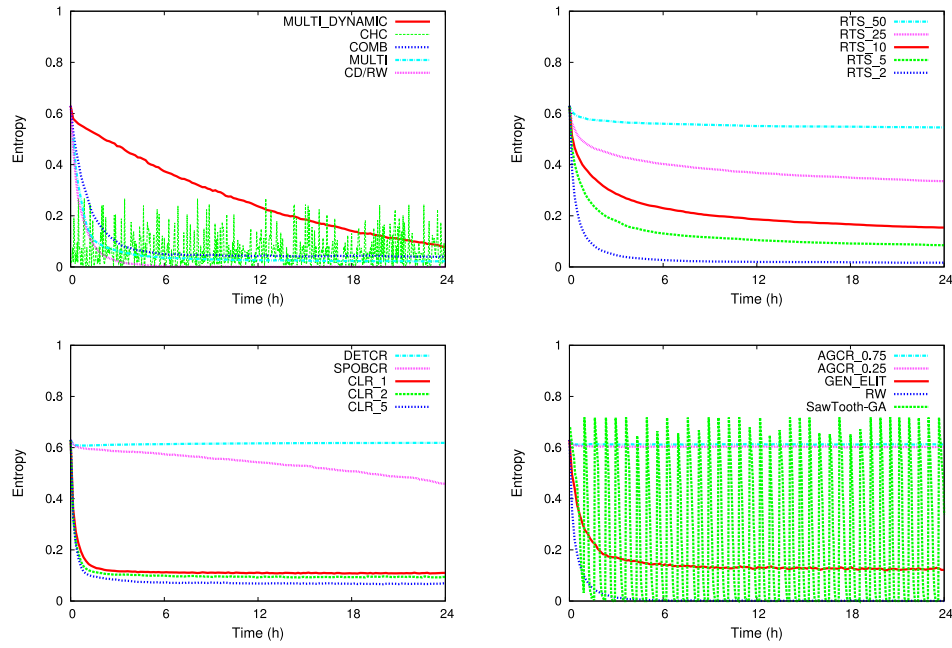
Fig. 5. Evolution of the mean of the entropy for all the considered schemes (GECCO instance).
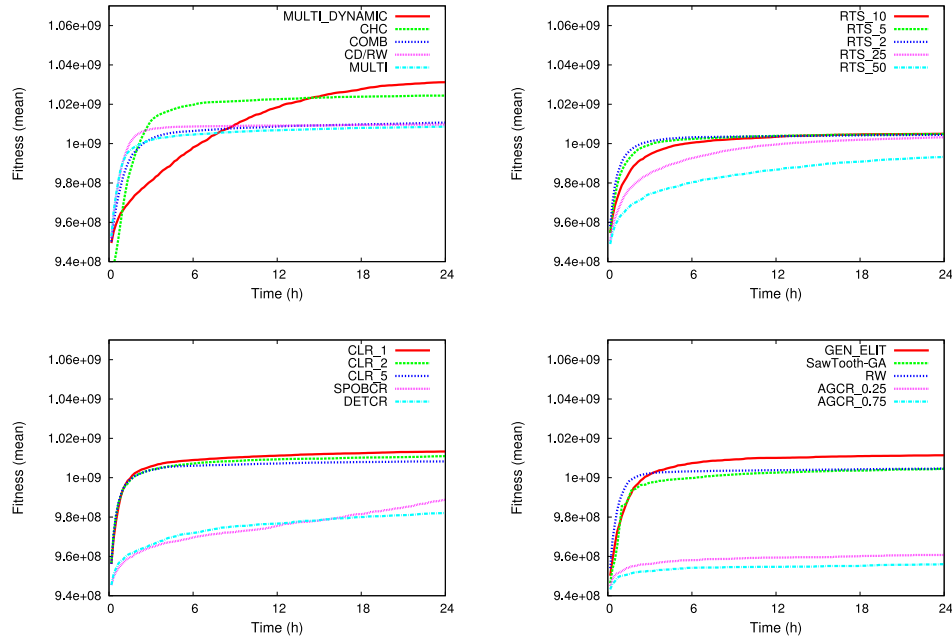

Fig. 6. Evolution of the mean of the fitness for all the considered schemes (GECCO instance).

high as those attained by CLR_1 or CHC. However, this slow convergence is quite useful, in the sense that at the end of the executions, the differences between the MULTI_DYNAMIC results and those obtained by any other schemes are quite large. The behavior of CHC is especially interesting. In this case, the diversity is lost quite fast but is then recovered in each reinitialization. This yields very good values in the middle term; however, in the long term it is clearly inferior to MULTI_DYNAMIC. Note that these results do not imply that our scheme is not useful for lower stopping criteria. The reason is that if the stopping criterion is set to a more restricted time, the balance between exploration and exploitation induced by

our scheme would be altered quicker. In any case, the application of our proposal is not encouraging when very short times are taken into account because the main advantages arise from the controlled loss of diversity, which is not an important issue in very short executions. In order to better illustrate this issue, we note that when Table I is generated with a stopping criterion of 3 h, MULTI_DYNAMIC ranks second with 149 points. For this time period, CHC was the best approach with 169 points. When the stopping criterion was set to 5 h, MULTI_DYNAMIC was the best method, obtaining 173 points. Finally, in the 12-h run, MULTI_DYNAMIC was also the best scheme and obtained the maximum attainable score.

It is also interesting to remark that, in the largest instances, the fitness of MULTI_DYNAMIC is still increasing around the termination time. This means that starting the intensification earlier in the optimization process might be beneficial. In Section VII, we discuss some alternatives to improve the performance for those cases.

Finally, we would like to remark that our proposal should not be regarded as an anytime EA [63]. In fact, in anytime optimizers, the stopping criterion is usually unknown, so our adaptive penalties cannot be applied. In addition, usually it is not just the end results that are used to measure the performance [64]; the progress over time is also taken into account. Since our proposal induces a slow convergence, it will not perform correctly in this kind of metric.

In this experimental validation, the evolution of the fitness and entropy is only shown for the GECCO instance. However, it was also analyzed for the remaining ones. In those cases, similar conclusions can be drawn. It is also important to note that other methods for solving the 2DPP based on different ways of controlling the diversity have also been provided in the literature. For instance, island-based schemes and methods that resort to restarts and populations with variable sizes have been tested [19], [53]. All of these methods are significantly inferior to the one devised in this paper.

B. Analysis of Robustness

Since the MULTI_DYNAMIC scheme incorporates a new parameter (D_I), it is very important to analyze the robustness of the approach in terms of it. A detailed analysis of the robustness of MULTI_DYNAMIC with respect to this parameter is attached as supplementary material. The main conclusions obtained from this paper are as follows.

1) In instances with different features, the same parameterization is able to yield high-quality solutions.
2) Small modifications in the parameterization of MULTI_DYNAMIC do not have a large effect on the results.

Finally, note that in order to facilitate future comparisons, for each instance the best and mean results obtained with our proposal are also incorporated in the supplementary materials.

VII. Conclusion

Premature convergence is a well-known drawback of EAs that requires further research. Since this drawback is so popular, several schemes for dealing with it have been devised. Among these schemes, those based on modifying the replacement phase of EAs seem quite promising. However, they have not been the most popular schemes. This is why several ideas that have been used to tackle premature convergence by adapting different components of EAs have not been applied to modify the replacement phase. Particularly, in this paper we have presented the first replacement scheme (MULTI_DYNAMIC) that combines the idea of transforming a single-objective into a multiobjective problem, by considering diversity as an explicit objective, with the principle of adapting the balance induced between exploration and exploitation to the various optimization stages. In order to achieve a proper balance in the different stages, in this paper the stopping criterion, as well as the elapsed time or the evaluations already executed, are used as an input to the replacement strategy. This information is used to adapt the importance assigned to each objective involved, which is done through a dynamic penalty approach. In order to validate the method devised in this paper, some analyses with simple continuous functions are included. These analyses show that our proposal provides a gradual reduction in diversity that cannot be obtained with other approaches. We also study the influence of the internal parameters of the EA on this reduction. An extensive experimental analysis with a complex benchmark problem (2DPP) that was defined for a popular competition shows that MULTI_DYNAMIC is able to outperform a large number of methods in a large set of instances with different complexities. In fact, this new method significantly improves on the best-known results to date for this problem. Moreover, an analysis of the robustness reveals several other benefits of this newly designed method. Particularly, the analyses indicate that for the 2DPP, the same parameterization is able to obtain high-quality solutions for instances with different features, and that small changes to its internal parameter do not have a large impact on the results. Thus, the new method provides high-quality results and its use is not complex.

Several lines of future work might be explored. First, in order to analyze the generality of our scheme, we would like to apply it to different combinatorial and continuous problems. Since in many cases different ways of measuring distances between solutions might be proposed, we would like to study the properties that they should have in order to provide high-quality results. Specifically, these distances could be related to the definition of the variation operators, so some studies on this topic might be in order. Second, our experiments have shown that due to the use of a multiobjective replacement scheme, convergence is not obtained even in the final optimization stage. Moreover, in some instances the fitness value is still increasing around the termination time. One alternative to offset this drawback might be to update D in a way that, prior to the termination time, it reaches the value 0. However, even in this case complete convergence might be prevented by the use of multiobjective concepts. Since other replacement phases allow for complete convergence, they could be used in the last stages instead of the multiobjective scheme. In our opinion, intelligently combining different replacement schemes could yield some additional benefits. This might be combined with the use of different strategies to alter the D value, including some adaptive and/or self-adaptive schemes. Another alternative would be to adapt the way in which a nondominated individual is selected in our scheme. For instance, it seems encouraging to be more greedy in terms of the original objective in the final phases. Finally, we would like to combine our diversity-preservation strategies with other methods used for the 2DPP. Specifically, since the crossover definition and the management of diversity are closely linked, applying some other crossover operators as the ones used in [21] seems to be very encouraging.

References

[1] R. Storn and K. Price, "Differential evolution—A simple and efficient heuristic for global optimization over continuous spaces," *J. Glob. Optim.*, vol. 11, no. 4, pp. 341–359, Dec. 1997.

[2] Y. Wang, H. Li, G. Yen, and W. Song, "MOMMOP: Multiobjective optimization for locating multiple optimal solutions of multimodal optimization problems," *IEEE Trans. Cybern.*, vol. 45, no. 4, pp. 830–843, Apr. 2015.

[3] F. Glover and G. A. Kochenberger, *Handbook of Metaheuristics* (International Series in Operations Research & Management Science). Boston, MA, USA: Kluwer Academic, 2003.

[4] Y. Borenstein and A. Moraglio, *Theory and Principled Methods for the Design of Metaheuristics*. Berlin, Germany: Springer, 2014.

[5] F. Herrera and M. Lozano, "Adaptation of genetic algorithm parameters based on fuzzy logic controllers," in *Genetic Algorithms and Soft Computing* (Studies in Fuzziness and Soft Computing). Heidelberg, Germany: Physica-Verlag, 1996, pp. 95–125.

[6] M. Črepinšek, S.-H. Liu, and M. Mernik, "Exploration and exploitation in evolutionary algorithms: A survey," *ACM Comput. Surv.*, vol. 45, no. 3, pp. 1–33, Jul. 2013.

[7] F. Herrera and M. Lozano, "Fuzzy adaptive genetic algorithms: Design, taxonomy, and future directions," *Soft Comput.*, vol. 7, no. 8, pp. 545–562, 2003.

[8] S. N. Sivanandam and S. N. Deepa, *Introduction to Genetic Algorithms*. Berlin, Germany: Springer, 2007.

[9] H. M. Pandey, A. Chaudhary, and D. Mehrotra, "A comparative review of approaches to prevent premature convergence in GA," *Appl. Soft Comput.*, vol. 24, pp. 1047–1077, Nov. 2014.

[10] S. W. Mahfoud, "Niching methods for genetic algorithms," Dept. Gen. Eng., Univ. Illinois Urbana–Champaign, Urbana, IL, USA, Tech. Rep. 95001, 1995.

[11] K. De Jong, *Evolutionary Computation: A Unified Approach* (A Bradford Book). Cambridge, MA, USA: MIT Press, 2006.

[12] L. J. Eshelman, "The CHC adaptive search algorithm: How to have safe search when engaging in nontraditional genetic recombination," in *Foundations of Genetic Algorithms*, vol. 1, G. J. Rawlins, Ed. San Mateo, CA, USA: Morgan Kaufmann, 1991, pp. 265–283.

[13] T. Blickle and L. Thiele, "A comparison of selection schemes used in evolutionary algorithms," *Evol. Comput.*, vol. 4, no. 4, pp. 361–394, 1996.

[14] E. Alba, *Parallel Metaheuristics: A New Class of Algorithms* (Wiley Series on Parallel and Distributed Computing). Hoboken, NJ, USA: Wiley, 2005.

[15] A. Eiben and J. E. Smith, *Introduction to Evolutionary Computing* (Natural Computing Series). Berlin, Germany: Springer, 2003.

[16] O. J. Mengshoel, S. F. Galán, and A. de Dios, "Adaptive generalized crowding for genetic algorithms," *Inf. Sci.*, vol. 258, pp. 140–159, Feb. 2014.

[17] T. Vidal, T. G. Crainic, M. Gendreau, and C. Prins, "A hybrid genetic algorithm with adaptive diversity management for a large class of vehicle routing problems with time-windows," *Comput. Oper. Res.*, vol. 40, no. 1, pp. 475–489, 2013.

[18] S. Das, A. Konar, and U. K. Chakraborty, "Two improved differential evolution schemes for faster global search," in *Proc. Conf. Genet. Evol. Comput. (GECCO)*, Washington, DC, USA, 2005, pp. 991–998.

[19] E. Segredo, C. Segura, and C. León, "Memetic algorithms and hyperheuristics applied to a multiobjectivised two-dimensional packing problem," *J. Glob. Optim.*, vol. 58, no. 4, pp. 769–794, 2014.

[20] C. León, G. Miranda, and C. Segura, "A memetic algorithm and a parallel hyperheuristic island-based model for a 2D packing problem," in *Proc. 11th Annu. Conf. Genet. Evol. Comput. (GECCO)*, New York, NY, USA, 2009, pp. 1371–1378.

[21] J. Kim and B.-R. Moon, "A hybrid genetic algorithm for a variant of two-dimensional packing problem," in *Proc. 11th Annu. Conf. Genet. Evol. Comput. (GECCO)*, New York, NY, USA, 2009, pp. 287–292.

[22] M. Tang, "A memetic algorithm for the location-based continuously operating reference stations placement problem in network real-time kinematic," *IEEE Trans. Cybern.*, vol. 45, no. 10, pp. 2214–2223, Oct. 2015.

[23] G. Zhang and Y. Li, "A memetic algorithm for global optimization of multimodal nonseparable problems," *IEEE Trans. Cybern.*, 2015. Doi: 10.1109/TCYB.2015.2447574.

[24] W. Gao, G. G. Yen, and S. Liu, "A cluster-based differential evolution with self-adaptive strategy for multimodal optimization," *IEEE Trans. Cybern.*, vol. 44, no. 8, pp. 1314–1327, Aug. 2014.

[25] W. N. Martin, J. Lienig, and J. P. Cohoon, "Island (migration) models: Evolutionary algorithms based on punctuated equilibria," in *Handbook of Evolutionary Computation*. Bristol, U.K.: Inst. Phys., 1997, pp. 1–16.

[26] E. Cantu-Paz, *Efficient and Accurate Parallel Genetic Algorithms*. Norwell, MA, USA: Kluwer Academic, 2000.

[27] M. Wineberg and J. Chen, "The shifting balance genetic algorithm as more than just another island model GA," in *Genetic and Evolutionary Computation* (LNCS 3103). Berlin, Germany: Springer, 2004, pp. 318–329.

[28] K. Deb and D. E. Goldberg, "An investigation of niche and species formation in genetic function optimization," in *Proc. 3rd Int. Conf. Genet. Algorithms*, San Francisco, CA, USA, 1989, pp. 42–50.

[29] S. F. Galán, O. J. Mengshoel, and R. Pinter, "A novel mating approach for genetic algorithms," *Evol. Comput.*, vol. 21, no. 2, pp. 197–229, May 2013.

[30] W.-J. Yu *et al.*, "Differential evolution with two-level parameter adaptation," *IEEE Trans. Cybern.*, vol. 44, no. 7, pp. 1080–1099, Jul. 2014.

[31] 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.

[32] C. Segura, C. A. Coello Coello, E. Segredo, and C. Leon, "An analysis of the automatic adaptation of the crossover rate in differential evolution," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Beijing, China, Jul. 2014, pp. 459–466.

[33] F. J. Lobo, C. F. Lima, and Z. Michalewicz, Eds., *Parameter Setting in Evolutionary Algorithms* (Studies in Computational Intelligence), vol. 54. Berlin, Germany: Springer, 2007.

[34] E. Segredo, C. Segura, and C. León, "Fuzzy logic-controlled diversity-based multi-objective memetic algorithm applied to a frequency assignment problem," *Eng. Appl. Artif. Intell.*, vol. 30, pp. 199–212, Apr. 2014.

[35] S. Y. Yuen and C. K. Chow, "A genetic algorithm that adaptively mutates and never revisits," *IEEE Trans. Evol. Comput.*, vol. 13, no. 2, pp. 454–472, Apr. 2009.

[36] T. Jansen, "On the analysis of dynamic restart strategies for evolutionary algorithms," in *Parallel Problem Solving From Nature—PPSN VII* (LNCS 2439), J. J. M. Guervós, P. Adamidis, H.-G. Beyer, H.-P. Schwefel, and J.-L. Fernández-Villacañas, Eds. Berlin, Germany: Springer, 2002, pp. 33–43.

[37] V. K. Koumousis and C. P. Katsaras, "A saw-tooth genetic algorithm combining the effects of variable population size and reinitialization to enhance performance," *IEEE Trans. Evol. Comput.*, vol. 10, no. 1, pp. 19–28, Feb. 2006.

[38] A. E. Eiben and C. A. Schippers, "On evolutionary exploration and exploitation," *Fundam. Inform.*, vol. 35, nos. 1–4, pp. 35–50, 1998.

[39] A. Moraglio and R. Poli, "Product geometric crossover," in *Parallel Problem Solving From Nature—PPSN IX* (LNCS 4193), T. P. Runarsson *et al.*, Eds. Berlin, Germany: Springer, 2006, pp. 1018–1027.

[40] H. Ishibuchi, N. Tsukamoto, and Y. Nojima, "Diversity improvement by non-geometric binary crossover in evolutionary multiobjective optimization," *IEEE Trans. Evol. Comput.*, vol. 14, no. 6, pp. 985–998, Dec. 2010.

[41] D. J. Cavicchio, "Adaptive search using simulated evolution," Ph.D. dissertation, Univ. Michigan, Ann Arbor, MI, USA, 1970.

[42] K. A. De Jong, "An analysis of the behavior of a class of genetic adaptive systems," Ph.D. dissertation, Dept. Comput. Commun. Sci., Univ. Michigan, Ann Arbor, MI, USA, 1975, Art. ID aAI7609381.

[43] S. W. Mahfoud, "Crowding and preselection revisited," in *Parallel Problem Solving From Nature 2*, R. Männer and B. Manderick, Eds. Amsterdam, The Netherlands: Elsevier, 1992, pp. 27–36.

[44] O. J. Mengshoel and D. E. Goldberg, "Probabilistic crowding: Deterministic crowding with probabilistic replacement," in *Proc. Genet. Evol. Comput. Conf. (GECCO)*, Orlando, FL, USA, Jul. 1999, pp. 409–416.

[45] P. J. Ballester and J. N. Carter, "An effective real-parameter genetic algorithm with parent centric normal crossover for multimodal optimisation," in *Genetic and Evolutionary Computation (GECCO)* (LNCS 3102), K. Deb, Ed. Berlin, Germany: Springer, 2004, pp. 901–913.

[46] G. R. Harik, "Finding multimodal solutions using restricted tournament selection," in *Proc. 6th Int. Conf. Genet. Algorithm*, San Francisco, CA, USA, 1995, pp. 24–31.

[47] A. Petrowski, "A clearing procedure as a niching method for genetic algorithms," in *Proc. IEEE Int. Conf. Evol. Comput.*, Nagoya, Japan, May 1996, pp. 798–803.

[48] M. Lozano, F. Herrera, and J. R. Cano, "Replacement strategies to preserve useful diversity in steady-state genetic algorithms," *Inf. Sci.*, vol. 178, no. 23, pp. 4421–4433, 2008.

[49] L. T. Bui, H. A. Abbass, and J. Branke, "Multiobjective optimization for dynamic environments," in *Proc. IEEE Congr. Evol. Comput.*, vol. 3. Edinburgh, Scotland, 2005, pp. 2349–2356.

[50] J.-B. Mouret, "Novelty-based multiobjectivization," in *New Horizons in Evolutionary Robotics* (Studies in Computational Intelligence), vol. 341, S. Doncieux, N. Bredéche, and J.-B. Mouret, Eds. Berlin, Germany: Springer, 2011, pp. 139–154.

[51] 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.

[52] C. Segura, C. A. C. Coello, E. Segredo, G. Miranda, and C. Leon, "Improving the diversity preservation of multi-objective approaches used for single-objective optimization," in *Proc. IEEE Congr. Evol. Comput.*, Cancún, Mexico, Jun. 2013, pp. 3198–3205.

[53] C. Segura, E. Segredo, and C. León, "Parallel island-based multiobjectivised memetic algorithms for a 2D packing problem," in *Proc. 13th Annu. Conf. Genet. Evol. Comput.*, Dublin, Ireland, 2011, pp. 1611–1618.

[54] J. D. Knowles, R. A. Watson, and D. W. Corne, "Reducing local optima in single-objective problems by multi-objectivization," in *Proc. 1st Int. Conf. Evol. Multi-Criter. Optim.*, Zürich, Switzerland, 2001, pp. 269–283.

[55] C. Segura, C. A. C. Coello, G. Miranda, and C. León, "Using multi-objective evolutionary algorithms for single-objective constrained and unconstrained optimization," *Ann. Oper. Res.*, pp. 1–34, Sep. 2015. Doi: 10.1007/s10479-015-2017-z.

[56] Y.-S. Ong, M.-H. Lim, N. Zhu, and K.-W. Wong, "Classification of adaptive memetic algorithms: A comparative study," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 36, no. 1, pp. 141–152, Feb. 2006.

[57] H. Ishibuchi, T. Yoshida, and T. Murata, "Balance between genetic search and local search in memetic algorithms for multiobjective permutation flowshop scheduling," *IEEE Trans. Evol. Comput.*, vol. 7, no. 2, pp. 204–223, Apr. 2003.

[58] H. Ishibuchi, Y. Hitotsuyanagi, Y. Wakamatsu, and Y. Nojima, "How to choose solutions for local search in multiobjective combinatorial memetic algorithms," in *Parallel Problem Solving From Nature, PPSN XI* (LNCS 6238), R. Schaefer, C. Cotta, J. Kołodziej, and G. Rudolph, Eds. Berlin, Germany: Springer, 2010, pp. 516–525.

[59] V. A. Shim, K. C. Tan, and H. Tang, "Adaptive memetic computing for evolutionary multiobjective optimization," *IEEE Trans. Cybern.*, vol. 45, no. 4, pp. 610–621, Apr. 2015.

[60] C. León, G. Miranda, and C. Segura, "METCO: A parallel plugin-based framework for multi-objective optimization," *Int. J. Artif. Intell. Tools*, vol. 18, no. 4, pp. 569–588, 2009.

[61] T.-P. Hong, M.-W. Tsai, and T.-K. Liu, "Two-dimensional encoding schema and genetic operators," in *Proc. Joint Conf. Inf. Sci. (JCIS)*, Kaohsiung, Taiwan, 2006.

[62] J. J. Durillo *et al.*, "A study of multiobjective metaheuristics when solving parameter scalable problems," *IEEE Trans. Evol. Comput.*, vol. 14, no. 4, pp. 618–635, Aug. 2010.

[63] A. Radulescu, M. López-Ibáñez, and T. Stützle, "Automatically improving the anytime behaviour of multiobjective evolutionary algorithms," in *Evolutionary Multi-Criterion Optimization* (LNCS 7811), R. C. Purshouse, P. J. Fleming, C. M. Fonseca, S. Greco, and J. Shaw, Eds. Berlin, Germany: Springer, 2013, pp. 825–840.

[64] T. Weise *et al.*, "Benchmarking optimization algorithms: An open source framework for the traveling salesman problem," *IEEE Comput. Intell. Mag.*, vol. 9, no. 3, pp. 40–52, Aug. 2014.

Carlos A. Coello Coello (M'98–SM'04–F'11) received the Ph.D. degree in computer science from Tulane University, New Orleans, LA, USA, in 1996.

He is currently a Professor (CINVESTAV-3F Researcher) and the Chair of the Department of Computer Science, CINVESTAV-IPN, Mexico City, Mexico. He has authored and co-authored over 450 technical papers and book chapters. He has also co-authored the book entitled *Evolutionary Algorithms for Solving Multi-Objective Problems* (Second Edition, Springer, 2007). His publications currently report over 29 000 citations in Google Scholar with an H-index of 67. His current research interests include evolutionary multiobjective optimization and constraint-handling techniques for evolutionary algorithms.

Dr. Coello was a recipient of the 2007 National Research Award from the Mexican Academy of Sciences in the area of exact sciences, and the Medal to the Scientific Merit 2009 granted by Mexico City's congress. He is currently an Associate Editor of the IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION and serves in the editorial board of 12 other international journals. He is a member of ACM, Sigma Xi, and the Mexican Academy of Science.

Eduardo Segredo (S'14–M'15) was born in Santa Cruz de Tenerife, Spain, in 1984. He received the B.S., M.S., and Ph.D. degrees from the Universidad de La Laguna, San Cristóbal de La Laguna, Spain, in 2006, 2008, and 2014, respectively, all in computer science.

From 2009 to 2010, he was with Atos Origin Consulting Canarias, Santa Cruz de Tenerife. From 2011 to 2012, he was a Visitor with Edinburgh Parallel Computing Centre, Edinburgh, U.K. He is currently with MHP Servicios de Control, Santa Cruz de Tenerife, and collaborates in teaching tasks with the Universidad de La Laguna. His current research interests include multiobjective and single-objective optimization, meta-heuristics, hyper-heuristics, parameter setting in evolutionary algorithms, and diversity-based multiobjective evolutionary algorithms.

Carlos Segura (M'13) was born in Santa Cruz de Tenerife, Spain, in 1983. He received the M.S. and Ph.D. degrees from the Universidad de La Laguna, San Cristóbal de La Laguna, Spain, in 2009 and 2012, respectively, both in computer science.

He is currently an Associate Researcher with the Department of Computer Science, Centre for Research in Mathematics, Guanajuato, Mexico. He has authored and co-authored over 50 technical papers and book chapters, including 15 journal papers. His publications currently report over 300 citations in Google Scholar with an H-index of 12. His current research interests include design of evolutionary algorithms, diversity management, and problem solving paradigms.

Dr. Segura currently serves in the editorial board of several international conferences. He is a member of ACM.

Arturo Hernández Aguirre received the Ph.D. degree in computer science from Tulane University, New Orleans, LA, USA, in 2001.

He has authored over 30 journal papers and over 100 conference papers and edited two books of the Springer Series Lecture Notes in Computer Science. His current research interests include evolutionary computation algorithms for global and constrained optimization, and estimation of distribution algorithms for global optimization in real parameter space.

Dr. Aguirre currently serves as an Associate Editor of the *Computational Optimization and Computation Journal*.