Swarm and Evolutionary Computation 54 (2020) 100671

Contents lists available at ScienceDirect

Swarm and Evolutionary Computation

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

A better balance in metaheuristic algorithms: Does it exist?

Bernardo Morales-Castañeda *, Daniel Zaldívar, Erik Cuevas, Fernando Fausto, Alma Rodríguez

Departamento de Electrónica, Universidad de Guadalajara, CUCEI, Blvd. Marcelino García Barragán 1421, 44430, Guadalajara, Mexico

ARTICLE INFO

Keywords:
Metaheuristic optimization
Balance
Exploration-exploitation
Population diversity

ABSTRACT

The constant development of new metaheuristic algorithms has led to a saturation in the field of stochastic search. There are now hundreds of different algorithms that can be used to solve any problem. To produce a good performance, every metaheuristic method needs to address a satisfactory equilibrium between exploration and exploitation of the search space. Although exploration and exploitation represent two fundamental concepts in metaheuristics, the main questions about their combination and balance have not been yet completely understood. Most of the existent analyzes conducted on metaheuristic techniques consider only the comparison of their final results which cannot evaluate the nature of a good or bad balance. This paper presents an experimental analysis that quantitatively evaluates the balance between exploration and exploitation of several of the most important and better-known metaheuristic algorithms. In the study, a dimension-wise diversity measurement is used to assess the balance of each scheme considering a representative set of 42 benchmark problems that involve multimodal, unimodal, composite and shifted functions. As a result, the analysis provides several observations that allow understanding how this balance affects the results in each type of functions, and which balance is producing better solutions.

1. Introduction

In recent years, metaheuristics search algorithms have gained popularity as tools for solving a wide array of optimization problems in many different areas of application, including engineering design, digital image processing, and computer vision, networks and communications, power, and energy management, data analysis and machine learning, robotics, medical diagnosis, and others [1].

Most metaheuristic methods model population-based search schemes, in which a population of search agents (or individuals) applies specific sets of rules to explore different candidate solutions within a feasible solution space. These optimization frameworks present several advantages, including the interaction among individuals (which promotes the exchange of knowledge among different solutions) and the diversification of the population (which is important to ensure the efficient exploration of the search space and the ability to overcome local optima) [2].

Metaheuristic search methods are so numerous and varied in terms of design and potential applications [41]; however, for such an abundant family of optimization techniques, there seems to be a question that needs to be answered: Which part of the design in a metaheuristic algorithm contributes more to its performance? One widely accepted principle among researchers considers that metaheuristic search methods can reach a better performance when an appropriate balance between exploration and exploitation of solutions is achieved [3]. While there seems to exist a general agreement on this concept, in fact, there is barely a vague conception of what the balance of exploration and exploitation really represent [4,5]. Indeed, the classification of search operators and strategies present in a metaheuristic method is often ambiguous, since they can contribute in some way to explore or exploit the search space [6].

In the absence of consistent knowledge about the mechanism that controls the balance between exploration and exploitation, several attempts have been conducted to fill these gaps. Most of these efforts have proposed interesting metrics that allow quantifying the level of exploration and exploitation in search algorithms through the monitoring of the current population diversity [4–11]. Although several indexes exist and more are being proposed, there is no definitive way to objectively measure the rate of exploration/exploitation provided in a metaheuristic scheme.

One of these metrics is the dimension-wise diversity measurement proposed in Ref. [10,12]. This index calculates the averaged sum distance between all the solutions to the median of each dimension and solutions.

* Corresponding author.
E-mail addresses: jb.moralescastaneda@gmail.com (B. Morales-Castañeda), daniel.zaldivar@cucei.udg.mx (D. Zaldívar), erik.cuevas@cucei.udg.mx (E. Cuevas), abraham.fausto@academicos.udg.mx (F. Fausto), alma.rvazquez@academicos.udg.mx (A. Rodríguez).

https://doi.org/10.1016/j.swevo.2020.100671
Received 28 May 2019; Received in revised form 21 December 2019; Accepted 26 February 2020
Available online 2 March 2020

Then, the exploration percent of each iteration is calculated by dividing the diversity value between the maximum diversity value encounter in the whole optimization process. On the other hand, the exploitation rate is considered the inverse of exploration percent. This measurement allows to find out how spread or clustered the search agents are over specific iterations during the search process. These values give information on how much time the algorithm behaves exploring or exploiting solutions. In spite of the information provided by this index, it has not been adopted in the community yet to characterize the balance between exploration and exploitation in metaheuristic methods.

In this paper, an experimental analysis is proposed to quantitatively evaluate the balance between exploration and exploitation on several of the most important and better-known metaheuristic algorithms. As a result, the analysis provides several observations that allow understanding of how this balance affects the results in each type of functions, and which balance is producing for better solutions. The methods studied in this work include Artificial Bee Colony (ABC), Bat Algorithm (BA), Covariance Matrix Adaptation Evolution Strategies (CMA-ES), Crow Search Algorithm (CSA), Differential Evolution (DE), Firefly Algorithm (FA), Grey Wolf Optimizer (GWO), Moth-Flame Optimization (MFO), Particle Swarm Optimization (PSO), Social Spiders Optimization (SSO), Teaching-Learning Based Optimization (TLBO) and Whale Optimization Algorithm (WOA), which are considered some of the most important metaheuristic search algorithms on the current literature in virtue of their performance, novelty and potential.

This paper is organized as follows: In Section 2, we open a discussion related to the concepts of Exploration and Exploitation. In Section 3, we discuss the dimension-wide diversity measurement proposed in Ref. [12] and their applications for the analysis of exploration and exploitation in metaheuristic search methods. In Section 4, we present our experimental analysis. Section 5 presents a discussion of the results. Finally, in Section 6, conclusions are drawn.

2. Exploration and exploitation

For every metaheuristic algorithm, exploration and exploitation represent the most important characteristics for attaining success when solving a particular optimization problem. Exploration refers to the ability of a search algorithm to discover a diverse assortment of solutions, spread within different regions of the search space. On the other hand, exploitation emphasizes the idea of intensifying the search process over-promising regions of the solution space with the aim of finding better solutions or improving the existing ones [4]. Regarding these concepts, empirical experimentation has demonstrated that there is a strong relationship between the exploration-exploitation capacity of a certain search method and its convergence rate. In particular, while exploitation strategies are known to enhance convergence speed toward a global optimum, they are also known to increase the probability of entrapment into local optima. Conversely, search strategies that favor exploration over exploitation tend to increase the probability of finding regions within the search space where the global optimum is more likely to be located, this at the cost of deterioration on the algorithm's convergence speed [13].

In recent years, questions about how exploration and exploitation of solutions is achieved in metaheuristic optimization algorithms has remained as an open subject, and although seemingly trivial, it has remained as a source of disagreement among many researchers [11,14]; Although many ideas and concepts may seem opposite, there seems to be a common agreement within the research community on the idea that a good ratio between exploration and exploitation is essential to ensure good performance in this kind of search methods. Naturally, the question that often arises from such an idea is: Which are the optimal exploration and exploitation rates required to conduct an efficient search process? Given that metaheuristic search methods can be very different in terms of search strategy, it is difficult (if not impossible) to give an appropriate exploration/exploitation rate that works for every single method in existence; in fact, it becomes evident that understanding the workings of the mechanisms implemented by these methods (and how these contribute to the search process) is necessary to devise an efficient search strategy.

In most population-based search methods, for example, selection mechanisms that allow choosing prominent solutions from among the available population (either to integrate them on the next cycle of the searching process or implement some solution update strategy) are commonly applied in order to balance elitism and diversity [15]. In the case of greedy selection mechanisms, for example, it is ensured that the individuals with the best-found solutions among all candidate solutions remain intact for the next generation and is known to improve convergence speed toward promising solutions. Also, in algorithms like DE [16], where an individual greedy selection mechanism is applied, new solutions are accepted only if it improves the solution(s) that originate them. The appeal of this selection strategy is that it has the potential to improve the exploration-exploitation ratio of solutions, as it forces an initial (diverse) population to improve individually from its starting point [17].

On the other hand, there are search algorithms that do not implement any kind of selection strategy at all, accepting every new solution independently of their quality. While these methods do not enforce the selection of prominent individuals during their search process, the implementation of other search mechanisms is necessary in order to balance the exploration-exploitation rate. In the case of algorithms such as PSO [18] or GWO [19], search mechanisms that apply some kind of "attraction operators" are considered in order to enhance their exploitation capabilities. These mechanisms seek to improve a population of solutions by, either moving them toward the location of seemingly "good" individuals within said population, or toward the location of the current best solutions found so far by the search process. The way in which solutions are chosen as attractors, and how other solutions are attracted to these attractors entirely depends on the design of the search method itself. In the case of the PSO algorithm, for example, individuals are not only set to experience an attraction toward the global best solution at a given iteration (cycle) of the search process, but also toward the best solution(s) recorded by each particle as the search process evolves (personal best solution). While this approach is often considered a well-balanced attraction mechanism regarding convergence and solutions diversity, implementing this kind of search strategy requires the allocation of additional memory, which could be undesired depending on the intended application(s).

Furthermore, some other algorithms implement attraction mechanisms which consider the composite effect of more than one attractor in order to discover new solutions. These set of attractors can be comprised of a subset of all currently available individuals, or even, by the whole population. Also, there are methods that implement more complex attraction schemes, which consider not only very specific members among the available population but also other particular properties. In FA [20], for example, the attractiveness experienced by each individual is not only dependent on the quality of other members, as it is also set to be inversely proportional to the distance that separates them, hence, the longer the distance, the lower the attraction. Similarly, in GSA, attractiveness is dictated by the so-called "gravitation force" exerted among particles within the feasible search space, and the magnitude of such attractions depends not only on the fitness value of each solution but also on the distance separating them [21].

Moreover, there search methods that do not contemplate attraction mechanisms as part of their search strategy; instead, these methods generate new solutions by means of pure random walks or by taking other criterions into account (i.e. the distance between solutions, as in the case of DE). In methods such as GA, some solutions are generated by "mixing" the information of randomly chosen solution (crossover), whereas others are generated by adding perturbations to currently existing solutions (mutation). In this regard, it's worth mentioning that, while crossover and mutation operators in evolutionary algorithms are often seen as exploration and exploitation strategies, respectively, an indeep observation over the crossover behaviors may suggest that at the start of the search process (where the population is still diverse), crossover operators may indeed favor the exploration of solutions, whereas, toward the end of the process (where population has lost diversity), exploration capabilities are dramatically reduced. In a similar manner, mutation operators that modify existing solutions by applying large amounts of perturbations could be perceived as an exploration mechanism, as solutions might be generated over much larger proportions of the feasible solution space. With the previous being said, it is complicated to roughly classify crossover and mutation as either exploration or exploitation operators, as their intended behavior could be easily altered by adjusting their respective crossover and mutation rates.

Finally, it is worth noting that there are metaheuristic algorithms that consider the iterative stop criterion (maximum number of iterations), as well as the iterations progress as part of their search strategy. These mechanisms are mostly used to indirectly adjust the exploration-exploitation rate by modifying several parameters employed by the algorithm's search operators as the search process evolves, this with the purpose of avoiding premature convergence. However, as a result of this constant adjustment on the exploration-exploitation rate, the algorithm's convergence speed may suffer a significant impact, which could be undesired depending on certain situations [6].

3. Evaluation of the balance

Metaheuristic algorithms use a group of candidate solutions to explore the search space with the objective to find satisfactory solutions for an optimization problem. In general, search agents with the best solutions tend to direct the search process towards them. As a consequence of this attraction, the distance among search agents decreases while the effect of exploitation increases. On the other hand, when the distance among search agents increases, the effect of the exploration process is more evident.

To calculate the increase and decrease in distance among search agents, a diversity measurement known as the dimension-wise diversity measurement [10] is considered. Under this method the population diversity is defined as follows:

$$Div_j = \frac{1}{n}\sum_{i=1}^{n}\left|median(x^j) - x_i^j\right| \tag{1}$$

$$Div = \frac{1}{m}\sum_{j=1}^{m} Div_j \tag{2}$$

where $median(x^j)$ represents the median of dimension j in the whole population. x_i^j is the dimension j of search agent i. n corresponds to the number of search agents in the population while m symbolizes the number of design variables of the optimization problem.

The diversity in each dimension Div_j is defined as the distance between the dimension j of every search agent and the median of that dimension, averaged. The diversity of the entire population Div is then calculated by averaging every Div_j in each dimension. Both values are computed in every iteration.

The complete balance response is characterized as the percentage of exploration and exploitation invested by certain metaheuristic scheme. These values are computed in each iteration by means of the following models:

$$XPL\% = \left(\frac{Div}{Div_{max}}\right) \times 100 \tag{3}$$

$$XPT\% = \left(\frac{|Div - Div_{max}|}{Div_{max}}\right) \times 100 \tag{4}$$

where Div_{max} represents the maximum diversity value found in the entire optimization process.

The percentage of exploration $XPL\%$ represents the level of exploration as the relationship between the diversity in each iteration and the maximum reached diversity. The percentage of exploitation $XPT\%$ corresponds to the level of exploitation. It is calculated as the complemental percentage to $XPL\%$ because the difference between the maximum diversity and the current diversity of an iteration is produced as a consequence of the concentration of search agents. As can be seen, both elements $XPL\%$ and $XPT\%$ are mutually conflicting and complementary.

In the evaluation of the balance response, the use of the median value avoids inconsistencies by using a reference element. Another interesting property of the balance response is that the values of $XPL\%$ and $XPT\%$ are also influenced by the maximum diversity Div_{max} found during the entire optimization process. In fact, this value is used as a reference to evaluate the rate of exploration and exploitation. However, one disadvantage, of the dimension-wise diversity index is its filtering effect produced by the average combination of all dimensions. Therefore, small abrupt changes in diversity are partially smoothed. Under such conditions, the index used to evaluate the balance presents a very small tendency to exploitation since some very small abrupt changes in diversity are partially overlooked.

In order to illustrate the balance evaluation, a graphical example has been conducted. In the example a hypothetical metaheuristic scheme is used to optimize a simple two-dimensional function which is defined as follows:

$$f(x_1,x_2) = 3(1-x_1)^2 e^{-\left(x_1^2 - x_2^2\right)} - 10\left(\frac{x_1}{5} - x_1^3 - x_2^5\right) e^{\left(-x_1^2 - x_2^2\right)} - 1\Big/3e^{\left(-(x_1+1)^2 - x_2^2\right)} \tag{5}$$

Assuming the interval $-3 \le x_1, x_2 \le 3$, the function has a global maximum and two local maxima. Fig. 1(b) shows a three-dimensional plot of this function. A good ratio between the diversification of solutions in the search space and intensification of the best-found solutions symbolize a conflicting objective which should be examined when the performance of a metaheuristic scheme is analyzed. In spite of their differences, metaheuristic approaches maintain a standard design scheme. In the initial stage of the search strategy, the algorithm promotes the diversification, which implies the production of candidate solutions in diverse locations of the search space. As the iterations progress, the exploitation must be intensified to refine the quality of their solutions. Fig. 1(b) shows an example of the performance behavior during 450 iterations produced by a hypothetical method in terms of the balance evaluation defined by Eq. (3) and (4). From the Figure, six points (c), (d), (e), (f), (g) and (h) have been selected to exemplify the solution diversity (distribution) and their respective balance evaluations. Point (c) represents an early stage of the algorithm (about the 20 iteration) where the balance evaluation has as values $XPL\% = 90$ and $XPT\% = 10$. Under such percentages, the hypothetical method behaves with a clear tendency to the exploration of the search space. This effect can be illustrated by Fig. 1(c) in which the solutions maintain a high dispersion of the search space. Point (d) corresponds to the 60 iterations. In this position, the balance evaluation maintains a value of $XPL\% = 70$ along with $XPT\% = 30$. In this behavior, the algorithm presents mainly exploration with a low level of exploitation. The solution configuration of this behavior can be exemplified in Fig. 1(d) where the diversity is high with some groups of similar solutions. Points (e) and (f) correspond to the 150 and 250 iterations, respectively, where the balance evaluations have as values $\% = 25$, $XPT\% = 75$ and $XPL\% = 08$, $XPT\% = 92$, respectively. Under such percentages, the algorithm behavior has been inverted promoting more the exploitation than the exploration. Fig. 1(e) and (f) show the solution distribution for points (e) and (f) from Fig. 1(b). Under such configurations, the solutions are distributed in several clusters diminishing the total diversity. Finally, points (g) and (h) represent the last stages of the hypothetical method. In such positions, the algorithm maintains a clear tendency to the exploitation of the best-found solutions

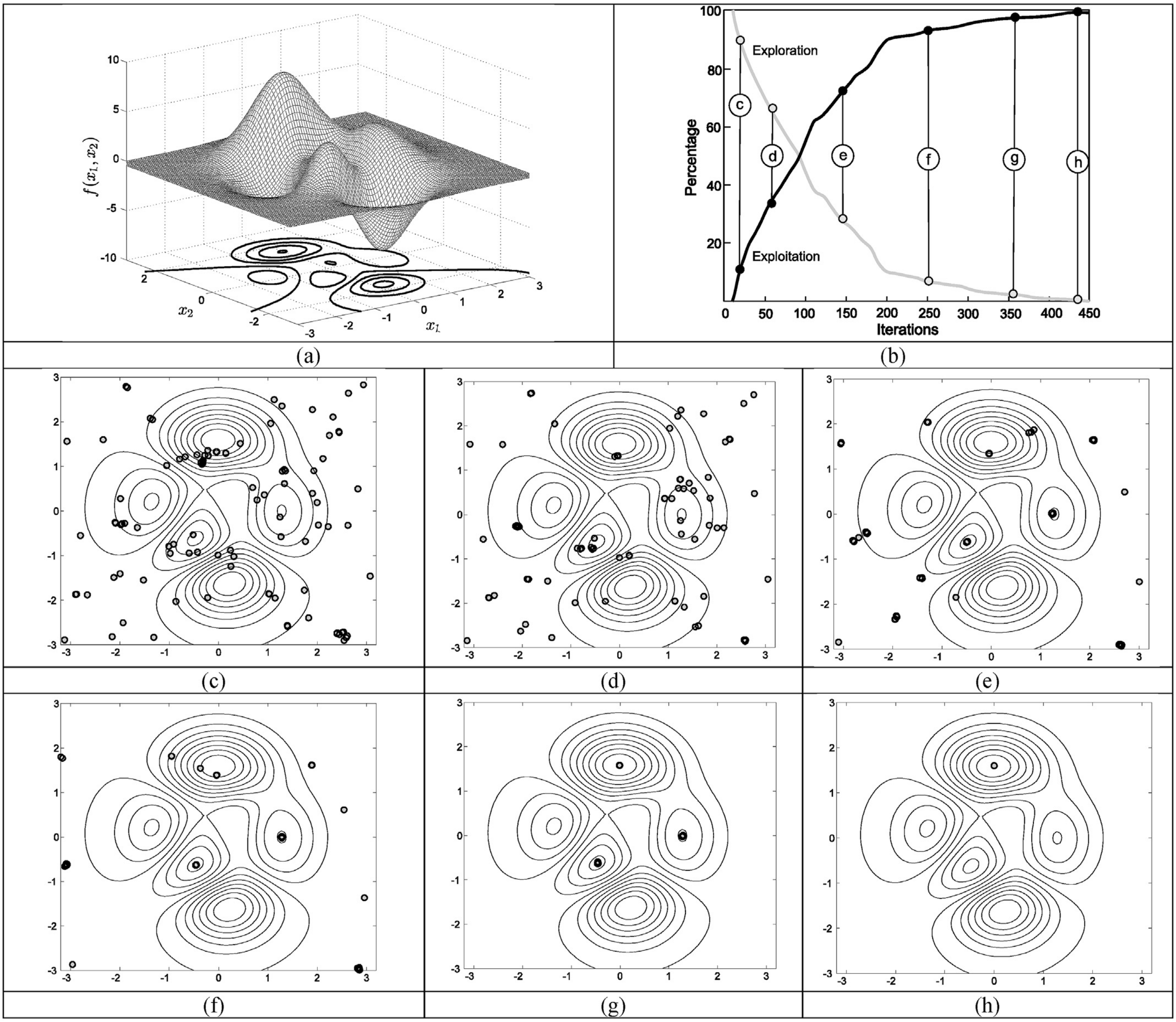

Fig. 1. Performance behavior during 450 iterations produced by a hypothetical method in terms of the balance evaluation defined by Eq. (3) and (4).

without considering any exploration mechanism. Fig. 1(g) and (h) correspond to the solution distribution for points (g) and (h).

4. Experiments

To analyze the balance between exploration and exploitation in many of the most popular and important metaheuristic algorithms, we examined the performance of 12 state-of-the-art optimization techniques: Artificial Bee Colony (ABC) [22,23], Bat Algorithm (BA) [24], Covariance Matrix Adaptation Evolution Strategy (CMA-ES) [25], Crow Search Algorithm (CSA) [26], Differential Evolution (DE) [16], Firefly Algorithm (FA) [27], Grey Wolf Optimization (GWO) [19], Social Spiders Optimization (SSO) [28], Teaching-Learning Based Optimization (TLBO) [29], Moth Flame Optimization (MFO) [30], Particle Search Optimization (PSO) [31] and Whale Optimization Algorithm (WOA) [32] in 43 well-known benchmark test functions (see Appendix A). The algorithms have been selected with the objective of covering a wide range of search mechanisms and design methodologies. Another criterion has been to include consolidate schemes and recently proposed methods. The availability of source code has been also an important factor as well as the popularity and performance reported in the literature. For all the algorithms the maximum function evaluations were 50,000 and unless otherwise stated, the population was 50. In some of the following tables, five different indicators are displayed: The average, median and standard deviation of the best-found solutions from 30 individual runs (**AB, MD, SD**), the average percentage of time spent exploring (**%XPL**), the average percentage of time spent exploiting (**%XPT**). To validate the performance differences among the algorithms, the Friedman test [33,40] has been considered. This study is a non-parametric analysis that aims to detect significant differences among the behavior of three or more related methods. During this test, the original results are firstly converted to ranks so that the best performing algorithm is assigned rank 1, the second rank 2, etc. The **average ranks** obtained by each algorithm are then one of the parameters used by the Friedman analysis. With this information, the test delivers a final result known as the ***p*-value**. If this value is small enough (less than 0.01 or 0.05 according to the significance level) we can determine that the null hypothesis is rejected, and in consequence, there is significant difference amon the behavior of the algorithms [33].

Table 1 contains several observable characteristics of each selected algorithm that influence exploration, exploitation or both. In this table

Table 1
Observable characteristics of selected algorithms.

Algorithms	Exploration/Exploitation		
	Selection mechanism	Attraction operators	Iteration dependent
ABC	Ind. Greedy	Multiple	NO
BA	Ind. Greedy	Global Best	NO
CMA-ES	Non-Greedy	N/A	NO
CSA	Non-Greedy	Personal	NO
DE	Ind. Greedy	N/A	NO
FA	Non-Greedy	Multiple	NO
GWO	Non-Greedy	Multiple	NO
MFO	Greedy	Multiple	YES
PSO	Non-Greedy	Multiple	NO
SSO	Non-Greedy	Multiple	NO
TLBO	Greedy	Multiple	NO
WOA	Non-Greedy	Global Best	NO

the selection mechanism, the number of attraction operators and if it considers the number of iterations as a part of the search strategy is detailed.

These algorithms were chosen according to their importance to the field, reported performance, novelty or accessibility of the source code. Table 2 shows the algorithm-specific settings taken from the literature.

As the total number of graphics depicting the rate of exploration and exploitation employed by every algorithm in every function is close to 500, the experimental results are divided into five sub-sections according to the type of functions: multimodal, unimodal, composite and shifted. The results from each sub-section are from the best performing algorithms to better illustrate what behaviors are being more effective in each type of function.

The experiments are divided into six sub-sections. In the first section (4.1), the balance of metaheuristic methods is analyzed when they face multimodal functions. In the second section (4.2), the study considers unimodal functions. In the third section (4.3), the balance responses over hybrid functions are discussed. In the fourth section (4.4), results over shifted functions are analyzed. In the fifth section (4.5), the balance response of the Random Search algorithm is examined. Finally, in the sixth section (4.6), a diversity analysis is conducted.

4.1. Results of multimodal test functions

Detailed information of each multimodal function, which is characterized by having multiple local optima, is presented in Appendix [A.I] Table AI. Table 3 presents the comparison results of the selected algorithms in the optimization of multimodal functions. Table 4 presents the statistical analysis outcomes by considering Friedman test. The best results are highlighted in boldface. The results displayed in both tables suggest that the best performing algorithms are CMAES, TLBO and WOA.

Fig. 2 shows the evolution of the exploration and the exploitation effects obtained by the top three algorithms in each of the 24 multimodal functions through all iterations. In order to visualize both effects as a characteristic, a new graph called incremental-decremental has been also added. In this graph, an increment is presented when the value of the exploration effect is higher or equal than the exploitation action ($XPL\% \geq XPT\%$). On the contrary, a decrement is produced when the exploitation value is superior to the exploration effect ($XPL\% < XPT\%$). In the case of negative values, they are assumed as zero. Under such conditions, both effects are clearly visible through the values and the extension of the graph. High values correspond to an extensive exploration action while low magnitudes refer to a strong exploitation effect. Likewise, the duration of the graph in high or low values reflexes the sustained effect of exploration or exploitation in the search strategy. The maximum value of the incremental-decremental graph takes place when both the effects of exploration and exploitation present the same level ($XPL\% = XPT\%$).

Tables 3 and 4 show that CMA-ES surpasses all other algorithms in four functions. According to Fig. 2, it exploited the search space 96.0574% of the time and explored 3.9426% of the time. TLBO won in 5 functions and tied in 3 with average exploitation of 91.7911% and 8.2089% exploration. Finally, WOA found the best solution on 7 functions and tied in 4, this algorithm employed a balance of 93.4241% exploitation and 6.5759% exploration. In all cases, the incremental-decremental graph shows that the effect of the exploration effect is very short while the exploitation action is prolonged during most of the time in the search strategy. These results suggest that the best balance for multimodal functions is closer to 90% exploitation and 10% exploration. These balances of exploration-exploitation have been produced as a consequence of the search mechanisms employed by each metaheuristic scheme. CMA-ES and WOA consider a non-greedy selection mechanism that provides a low exploration level. Conversely, TLBO used a greedy selection mechanism which slightly decreases the exploration rate. On the other hand, most of them use attraction operators toward the best global solution as a search strategy. Under such conditions, they promote exploitation at a high level.

There were four cases where CMA-ES, TLBO and WOA didn't find the best solutions, functions f_5, f_9, f_{10} and f_{21}. In these functions, the algorithms that got the best results were DE (f_5, f_{21}) and MFO (f_9, f_{10}). Fig. 3 shows the evolution of the exploration and the exploitation effects obtained in such functions by those algorithms. Beginning with f_5, we can already see that the DE algorithm focused less on exploitation compared to the top algorithms, and unlike in all the other multimodal functions, the best solution in f_{21} was obtained by DE with a focus on exploration, with a balance of 83.7028% of the time exploring and 16.2972% exploiting. In case of function f_5, the incremental-decremental graph shows that the exploration action maintains a wider effect during almost 200 iterations. On the contrary in function f_{21}, the incremental-decremental information demonstrates that the exploration effect is prolonged during almost the complete optimization process. To obtain such values, DE uses a search strategy that combines an increase of the exploration rate with the independence of a specific point attractor. Through these search mechanisms, DE can avoid being trapped in local minima and find better solutions compared to the other metaheuristic algorithms. Another remarkable aspect of DE is the high differences produced in its balances when it faces optimization problems. On the other hand, MFO obtains the same balance level as the best three algorithms. This fact is an effect of the multiple attractors considered in its search strategy. Since it uses several attractor points, its level of exploitation decreases, incrementing slightly the exploration of the search

Table 2
Parameter settings for each algorithm.

Algorithm	Parameters
ABC	Colony population = 124, Onlooker bees = 62, Employed bees = 62, Scout bee = 1, limit = 100 [23].
BA	Bat population = 50, alpha = 0.9, gamma = 0.9 [24].
CMA-ES	The algorithm has been configurated according to the guidelines provided by its author [34].
CSA	Crow population = 50, Flight length = 2, awareness probability = 0.1 [26].
DE	The weight parameter is set to $F = 0.75$ while the crossover probability is configurated to $CR = 0.2$ [35].
FA	The parameters set up for the randomness factor and the light absorption coefficient are set to $\alpha = 0.98^{it}$ and $\gamma = 1.0$ respectively, where *it* is the iteration number [36].
GWO	Wolf population = 30 [19].
MFO	The number of flames is set as $N_{flames} = round\left((N_{pop} - k) * \frac{N_{pop} - 1}{k_{max}}\right)$ where N_{pop} denotes the population size, k the current iteration and k_{max} the maximum number of iterations [30].
PSO	The learning factors are set to $c_1 = 2$ and $c_2 = 2$. On the other hand, the inertia weight factor is configurated to decrease linearly from 0.9 to 0.2 as the process evolves [37].
SSO	Number of females NFem = 50*rand, Number of males NMa = 50-NFem where rand is a number between 0.65 and 0.9.
TLBO	Population = 50.
WOA	Whale population = 30 [32].

Table 3
Best solutions found and balance employed in each multimodal function.

Function		ABC	BA	CMAES	CSA	DE	FA	GWO	MFO	PSO	SSO	TLBO	WOA
f_1	**AB**	1.72E+01	1.25E+01	2.06E+00	1.71E+01	5.12E-03	1.77E+00	1.01E-14	9.22E+00	5.44E+00	3.54E-01	5.33E-15	**3.08E-15**
	MD	1.73E+01	1.25E+01	7.11E-15	1.72E+01	4.84E-03	1.81E+00	8.88E-15	1.39E+01	2.38E-02	3.48E-01	5.33E-15	**3.55E-15**
	SD	4.69E-01	1.07E+00	6.30E+00	4.15E-01	9.34E-04	1.43E-01	3.24E-15	8.37E+00	7.98E+00	5.43E-02	**1.81E-15**	2.59E-15
	%XPL	87.4970	0.9934	2.0427	98.0255	13.4553	14.4189	0.4716	6.7257	44.4648	6.9182	1.0521	1.8393
	%XPT	12.5030	99.0066	97.9573	1.9745	86.5447	85.5811	99.5284	93.2743	55.5352	93.0818	98.9479	98.1607
	AB	2.86E+05	2.52E+01	6.67E-01	1.69E+05	1.46E+00	4.17E+00	**6.67E-01**	4.28E+04	1.80E+04	3.69E+00	**6.67E-01**	**6.67E-01**
f_2	**MD**	2.74E+05	3.92E+00	6.67E-01	1.70E+05	1.38E+00	3.92E+00	**6.67E-01**	8.24E+01	3.21E+02	3.34E+00	**6.67E-01**	**6.67E-01**
	SD	7.76E+04	4.93E+01	6.03E-15	4.70E+04	3.65E-01	1.20E+00	4.94E-06	1.16E+05	4.27E+04	1.46E+00	**3.41E-14**	8.19E-05
	%XPL	93.9632	3.1009	2.4524	98.5658	15.3386	19.1611	0.5150	5.7613	45.8991	19.1499	1.0151	2.5674
	%XPT	6.0368	96.8991	97.5476	1.4342	84.6614	80.8389	99.4850	94.2387	54.1009	80.8501	98.9849	97.4326
	AB	1.37E+02	7.76E+01	1.31E-03	1.70E+02	6.01E-03	1.10E+00	1.05E-03	1.19E-02	9.03E+00	1.66E-02	**0.00E+00**	1.34E-03
f_3	**MD**	1.39E+02	6.65E+01	0.00E+00	1.73E+02	3.98E-03	1.10E+00	0.00E+00	9.87E-03	8.86E-03	1.40E-02	**0.00E+00**	0.00E+00
	SD	1.61E+01	2.51E+01	3.47E-03	2.03E+01	5.66E-03	1.24E-02	4.40E-03	1.32E-02	2.75E+01	1.05E-02	**0.00E+00**	7.35E-03
	%XPL	78.6281	2.9400	2.0703	98.3754	9.9041	16.8046	0.4640	4.9714	38.7490	6.6239	0.9905	1.7932
	%XPT	21.3719	97.0600	97.9297	1.6246	90.0959	83.1954	99.5360	95.0286	61.2510	93.3761	99.0095	98.2068
	AB	4.98E-01	3.21E-06	5.29E-111	1.18E-01	2.42E-16	1.05E-09	4.92E-217	4.53E-08	2.17E-10	4.68E-05	1.98E-236	**0.00E+00**
f_4	**MD**	4.96E-01	3.07E-06	1.04E-112	1.07E-01	2.12E-16	8.43E-10	1.26E-221	1.19E-09	7.22E-12	3.98E-05	2.49E-238	**0.00E+00**
	SD	1.56E-01	1.00E-06	1.96E-110	4.14E-02	1.68E-16	7.67E-10	0.00E+00	1.23E-07	9.22E-10	3.14E-05	0.00E+00	**0.00E+00**
	%XPL	96.9063	7.1664	2.4101	98.1739	15.5385	18.7493	0.5353	7.6471	49.5202	55.0966	0.9975	2.3214
	%XPT	3.0937	92.8336	97.5899	1.8261	84.4615	81.2507	99.4647	92.3529	50.4798	44.9034	99.0025	97.6786
	AB	8.36E+01	7.82E+00	2.12E-01	5.83E+01	**2.52E-05**	3.47E-01	1.11E+00	2.48E+01	4.94E+00	3.09E-01	4.74E-01	9.66E-02
f_5	**MD**	8.47E+01	6.57E+00	1.50E-32	5.84E+01	**2.43E-05**	9.07E-02	1.09E+00	2.33E+01	3.50E+00	1.27E-01	4.92E-01	9.51E-02
	SD	1.21E+01	4.65E+00	5.06E-01	8.60E+00	**8.90E-06**	5.24E-01	2.14E-01	1.01E+01	4.94E+00	4.17E-01	2.22E-01	1.02E-01
	%XPL	95.0028	2.5900	2.6241	98.1753	17.3320	17.5826	2.6767	5.0162	47.1323	19.6133	1.2255	6.3444
	%XPT	4.9972	97.4100	97.3759	1.8247	82.6680	82.4174	97.3233	94.9838	52.8677	80.3867	98.7745	93.6556
	AB	1.91E-01	1.81E-04	5.57E-27	9.36E-02	**0.00E+00**	9.52E-08	7.04E-12	7.12E-07	0.00E+00	9.32E-07	**0.00E+00**	**0.00E+00**
f_6	**MD**	1.88E-01	5.85E-15	0.00E+00	9.09E-02	**0.00E+00**	8.42E-08	3.69E-12	0.00E+00	0.00E+00	7.50E-07	**0.00E+00**	**0.00E+00**
	SD	4.67E-02	9.76E-04	3.05E-26	2.18E-02	**0.00E+00**	4.01E-08	7.35E-12	3.90E-06	0.00E+00	6.86E-07	**0.00E+00**	**0.00E+00**
	%XPL	95.9426	3.5010	2.9087	98.4887	70.1519	13.0572	25.8766	4.0018	0.5549	20.5388	14.9194	24.2250
	%XPT	4.0574	96.4990	97.0913	1.5113	29.8481	86.9428	74.1234	95.9982	99.4451	79.4612	85.0806	75.7750
	AB	3.79E+04	6.36E+04	1.63E-87	5.81E+01	1.69E-111	1.03E-38	**0.00E+00**	4.66E-206	7.25E-33	4.39E-17	**0.00E+00**	**0.00E+00**
f_7	**MD**	5.61E+03	1.41E-04	1.80E-121	1.14E+01	1.17E-116	1.69E-42	**0.00E+00**	8.15E-225	1.36E-90	3.59E-22	**0.00E+00**	**0.00E+00**
	SD	1.10E+05	3.21E+05	6.12E-87	1.24E+02	8.88E-111	5.60E-38	**0.00E+00**	0.00E+00	3.97E-32	1.46E-16	**0.00E+00**	**0.00E+00**
	%XPL	86.4106	1.0759	3.5939	98.0479	12.4436	22.9480	0.6234	6.7044	37.5761	23.0048	1.3033	1.6233
	%XPT	13.5894	98.9241	96.4061	1.9521	87.5564	77.0520	99.3766	93.2956	62.4239	76.9952	98.6967	98.3767
	AB	1.15E+08	9.53E+06	**7.69E+01**	4.59E+07	2.00E+05	1.45E+03	2.27E+06	8.66E+01	9.20E+02	8.27E+01	2.64E+02	4.52E+02
f_8	**MD**	1.16E+08	1.03E+07	**7.65E+01**	4.51E+07	1.97E+05	1.22E+03	2.00E+06	8.73E+01	3.23E+02	8.20E+01	7.82E+01	1.45E+02
	SD	3.13E+07	3.34E+06	**3.07E+00**	9.43E+06	6.77E+04	9.15E+02	1.44E+06	3.74E+00	1.60E+03	2.51E+00	7.47E+02	1.19E+03
	%XPL	95.5115	3.4265	2.4201	98.5632	39.5521	18.0130	7.3906	6.4148	42.9836	8.8978	2.7120	10.5891
	%XPT	4.4885	96.5735	97.5799	1.4368	60.4479	81.9870	92.6094	93.5852	57.0164	91.1022	97.2880	89.4109
	AB	2.09E+08	2.82E+06	1.04E+02	7.46E+07	1.92E+04	1.22E+03	1.40E+05	**9.14E+01**	7.75E+02	1.21E+02	1.36E+02	4.60E+03
f_9	**MD**	2.08E+08	2.25E+06	1.04E+02	7.82E+07	1.75E+04	9.04E+02	1.25E+05	**9.18E+01**	2.38E+02	1.20E+02	1.22E+02	1.64E+02
	SD	6.05E+07	2.48E+06	4.61E+00	1.87E+07	7.84E+03	1.11E+03	7.09E+04	**3.72E+00**	1.54E+03	5.36E+00	5.87E+01	1.95E+04
	%XPL	95.3091	3.5499	2.3544	99.0277	24.8489	19.8155	3.7775	6.7386	48.9883	9.1842	2.0446	6.8941
	%XPT	4.6909	96.4501	97.6456	0.9723	75.1511	80.1845	96.2225	93.2614	51.0117	90.8158	97.9554	93.1059
	AB	6.18E+84	9.59E+83	1.89E+83	5.10E+83	3.06E+81	8.60E+81	3.60E+81	**1.66E+81**	4.40E+81	4.26E+81	1.82E+82	1.29E+82
f_{10}	**MD**	3.86E+84	8.54E+83	1.02E+83	4.32E+83	1.25E+81	6.39E+81	7.87E+80	**2.36E+80**	2.47E+81	9.35E+80	1.07E+82	3.02E+81
	SD	6.29E+84	7.77E+83	2.81E+83	3.87E+83	6.49E+81	7.58E+81	6.65E+81	**3.25E+81**	5.29E+81	8.50E+81	1.84E+82	2.56E+82
	%XPL	97.9017	1.0069	29.1786	99.1435	89.9032	51.7721	14.7970	8.4823	60.2020	33.6651	84.3472	12.5809
	%XPT	2.0983	98.9931	70.8214	0.8565	10.0968	48.2279	85.2030	91.5177	39.7980	66.3349	15.6528	87.4191
	AB	1.55E+83	2.85E+63	2.04E+02	2.90E+75	1.25E+02	8.65E+28	2.71E+02	7.61E+37	2.15E+49	8.15E+23	1.64E+02	**1.41E+02**
f_{11}	**MD**	8.93E+81	1.60E+54	3.03E+01	7.22E+73	9.93E+01	1.73E+21	1.15E+02	2.09E+25	4.00E+32	5.96E+17	5.41E+01	**5.09E+01**
	SD	4.24E+83	1.56E+64	4.10E+02	5.96E+75	7.70E+01	4.28E+29	3.81E+02	4.17E+38	1.18E+50	2.99E+24	2.53E+02	**2.52E+02**
	%XPL	97.5495	2.4453	0.5018	98.6073	20.5013	23.3910	0.9863	10.0874	58.0356	16.5102	1.1156	3.2459
	%XPT	2.4505	97.5547	99.4982	1.3927	79.4987	76.6090	99.0137	89.9126	41.9644	83.4898	98.8844	96.7541

	AB	4.22E+01	4.88E+01	**3.00E+01**	5.84E+01	3.00E+01	**3.00E+01**	**3.00E+01**	3.37E+01	3.47E+01	3.06E+01	**3.00E+01**	**3.00E+01**
f_{12}	**MD**	4.20E+01	4.80E+01	**3.00E+01**	5.90E+01	3.00E+01	**3.00E+01**	**3.00E+01**	3.50E+01	3.50E+01	3.00E+01	**3.00E+01**	**3.00E+01**
	SD	1.63E+00	3.54E+00	**0.00E+00**	2.91E+00	0.00E+00	**0.00E+00**	**0.00E+00**	3.70E+00	4.15E+00	7.74E-01	1.83E-01	1.83E-01
	%XPL	56.3797	2.2746	4.6431	98.0768	16.0563	17.8066	3.5525	9.1007	39.3777	27.7303	4.3927	4.7957
	%XPT	43.6203	97.7254	95.3569	1.9232	83.9437	82.1934	96.4475	90.8993	60.6223	72.2697	95.6073	95.2043
	AB	2.75E+03	1.06E-01	8.01E-05	1.93E+03	8.35E+01	1.31E+00	1.46E-06	4.56E+02	1.36E+03	2.96E+00	8.93E-06	**7.44E-07**
f_{13}	**MD**	2.88E+03	1.36E-02	7.46E-05	1.92E+03	8.25E+01	1.03E+00	3.72E-07	1.88E+02	5.62E+02	2.66E+00	1.05E-06	**1.88E-08**
	SD	7.50E+02	3.94E-01	3.69E-05	3.15E+02	2.08E+01	9.05E-01	2.44E-06	6.44E+02	1.48E+03	1.29E+00	1.75E-05	**1.85E-06**
	%XPL	90.3160	2.4176	1.0526	98.9323	70.1808	27.7344	1.1843	6.6424	22.9571	31.7263	1.1151	2.8662
	%XPT	9.6840	97.5824	98.9474	1.0677	29.8192	72.2656	98.8157	93.3576	77.0429	68.2737	98.8849	97.1338
	AB	4.28E+10	4.38E+09	**2.29E-28**	2.08E+10	6.42E+02	1.34E+04	1.22E+03	2.08E+09	1.16E+02	7.98E+00	1.08E-05	2.24E+02
f_{14}	**MD**	4.40E+10	4.08E+09	**2.14E-28**	2.03E+10	6.66E+02	1.16E+04	1.17E+03	2.90E-01	5.59E+01	7.04E+00	7.41E-07	1.61E+02
	SD	9.36E+09	2.48E+09	**6.23E-29**	4.80E+09	7.47E+01	5.72E+03	5.86E+02	1.14E+10	1.66E+02	2.86E+00	3.56E-05	2.32E+02
	%XPL	93.7338	4.5958	2.0125	98.6879	12.5087	15.4941	0.7743	8.0601	42.6197	9.9536	1.0140	2.1986
	%XPT	6.2662	95.4042	97.9875	1.3121	87.4913	84.5059	99.2257	91.9399	57.3803	90.0464	98.9860	97.8014
	AB	3.57E+01	1.31E+01	9.61E+00	2.43E+01	1.10E+01	9.92E+00	8.74E+00	1.19E+01	1.49E+01	1.16E+01	**8.78E+00**	9.10E+00
f_{15}	**MD**	4.69E+04	5.34E+01	**2.25E-09**	1.40E+04	1.35E+01	2.16E+01	1.46E+01	7.41E+03	2.23E-01	1.98E+01	1.52E-02	1.16E+01
	SD	4.70E+04	5.32E+01	**6.88E-15**	1.32E+04	1.32E+01	2.16E+01	1.42E+01	4.72E-03	1.57E-01	1.97E+01	3.32E-04	1.14E+01
	%XPL	94.4582	2.9616	11.8761	98.0231	37.4339	28.9769	3.3576	10.4171	49.7007	51.1412	17.1500	7.1632
	%XPT	5.5418	97.0384	88.1239	1.9769	62.5661	71.0231	96.6424	89.5829	50.2993	48.8588	82.8500	92.8368
	AB	1.24E+04	1.60E+01	**1.23E-08**	3.76E+03	1.28E+00	2.31E+00	4.90E+00	2.26E+04	2.16E-01	2.69E+00	5.52E-02	5.55E+00
f_{16}	**MD**	2.97E+02	6.57E+01	7.02E+01	2.58E+02	8.34E+01	5.90E+01	1.52E-01	1.36E+02	1.38E+02	5.71E+01	1.27E+01	**3.79E-15**
	SD	2.97E+02	6.17E+01	5.27E+01	2.65E+02	8.40E+01	5.45E+01	0.00E+00	1.25E+02	1.26E+02	5.72E+01	1.29E+01	**0.00E+00**
	%XPL	95.3699	4.8089	2.1825	98.2647	18.5119	17.4896	2.4108	5.9925	46.6230	19.3011	1.1184	5.3582
	%XPT	4.6301	95.1911	97.8175	1.7353	81.4881	82.5104	97.5892	94.0075	53.3770	80.6989	98.8816	94.6418
	AB	1.61E+01	3.21E+01	4.92E+01	1.60E+01	6.98E+00	1.91E+01	8.35E-01	3.52E+01	4.59E+01	1.21E+01	4.78E+00	**1.44E-14**
f_{17}	**MD**	2.38E+05	4.60E+02	1.13E+01	2.41E+05	7.54E+01	8.43E+01	2.70E+01	5.71E+04	8.45E+04	9.00E+01	2.31E+01	**2.64E+01**
	SD	2.46E+05	1.74E+02	1.59E-01	2.49E+05	7.51E+01	8.61E+01	2.71E+01	5.62E+04	5.81E+04	9.59E+01	2.31E+01	**2.63E+01**
	%XPL	89.3263	4.9894	4.2105	98.3033	46.8273	26.7953	1.3108	3.1251	50.2820	24.4155	5.9168	2.3435
	%XPT	10.6737	95.0106	95.7895	1.6967	53.1727	73.2047	98.6892	96.8749	49.7180	75.5845	94.0832	97.6565
	AB	6.46E+04	7.57E+02	3.19E+01	7.70E+04	2.21E+01	4.21E+01	7.33E-01	4.50E+04	5.50E+04	4.90E+01	7.63E-01	**3.01E-01**
f_{18}	**MD**	6.67E+01	3.29E+01	1.64E-13	4.97E+01	1.45E+01	9.38E+00	3.88E-25	5.29E+01	2.21E+01	1.70E-01	**4.54E-35**	3.35E+01
	SD	6.77E+01	3.31E+01	1.38E-13	5.08E+01	1.48E+01	9.60E+00	1.85E-25	5.07E+01	2.12E+01	1.65E-01	**4.03E-35**	2.80E+01
	%XPL	95.3792	3.5059	2.4252	98.6945	20.3476	18.4110	0.7932	6.3083	39.2793	23.6444	1.0447	1.8732
	%XPT	4.6208	96.4941	97.5748	1.3055	79.6524	81.5890	99.2068	93.6917	60.7207	76.3556	98.9553	98.1268
	AB	4.91E+00	4.22E+00	1.09E-13	3.43E+00	1.49E+00	2.33E+00	4.73E-25	1.14E+01	5.09E+00	2.72E-02	**2.75E-35**	3.05E+01
f_{19}	**MD**	2.59E+32	7.27E+27	1.03E+01	7.60E+29	3.11E-02	1.81E+01	2.41E-57	4.73E+02	7.42E+02	7.35E+04	4.36E-42	**2.80E-168**
	SD	2.49E+31	3.39E+22	1.23E-14	1.30E+29	3.02E-02	1.72E+01	5.61E-58	5.00E+02	8.00E+02	2.57E+02	3.38E-42	**5.92E-178**
	%XPL	96.9787	1.7034	2.3870	98.6170	48.5372	22.2809	0.7828	13.8845	64.5812	5.2652	1.1583	17.2994
	%XPT	3.0213	98.2966	97.6130	1.3830	51.4628	77.7191	99.2172	86.1155	35.4188	94.7348	98.8417	82.7006
	AB	6.87E+32	3.95E+28	5.66E+01	1.61E+30	6.65E-03	3.87E+00	4.40E-57	1.66E+02	2.06E+02	3.44E+05	2.93E-42	**0.00E+00**
f_{20}	**MD**	−4.98E+49	−1.43E+268	−5.82E+27	−8.05E+03	−1.21E+04	−9.12E+03	**−5.91E+03**	−8.56E+03	−8.21E+03	−7.28E+03	−7.40E+03	−1.18E+04
	SD	−1.02E+48	−3.81E+260	−3.35E+23	−7.81E+03	−1.22E+04	−8.99E+03	**−5.96E+03**	−8.35E+03	−8.04E+03	−7.21E+03	−7.76E+03	−1.23E+04
	%XPL	89.1435	1.5237	3.6850	98.1749	12.5994	20.5564	0.5686	5.5311	38.7221	19.2984	1.3860	2.6828
	%XPT	10.8565	98.4763	96.3150	1.8251	87.4006	79.4436	99.4314	94.4689	61.2779	80.7016	98.6140	97.3172
	AB	1.94E+50	6.55E+04	2.90E+28	8.68E+02	**4.21E+02**	6.92E+02	7.23E+02	9.93E+02	1.01E+03	5.62E+02	1.13E+03	1.18E+03
f_{21}	**MD**	1.49E+04	8.74E+03	0.00E+00	1.91E+04	**0.00E+00**	1.39E+01	0.00E+00	1.67E+03	1.33E+03	2.00E-01	0.00E+00	1.00E-01
	SD	1.47E+04	8.37E+03	0.00E+00	1.90E+04	**0.00E+00**	1.40E+01	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
	%XPL	1.1200	0.1531	0.8569	99.1186	83.7028	25.9312	19.3947	4.2543	8.3847	9.2638	43.7278	12.8460
	%XPT	98.8800	99.8469	99.1431	0.8814	16.2972	74.0688	80.6053	95.7457	91.6153	90.7362	56.2722	87.1540
	AB	2.27E+03	2.54E+03	0.00E+00	3.32E+03	0.00E+00	2.35E+00	0.00E+00	4.61E+03	3.46E+03	4.07E-01	**0.00E+00**	3.05E-01
f_{22}	**MD**	−6.58E+02	−9.93E+02	−1.04E+03	−7.24E+02	−1.17E+03	−1.07E+03	−9.31E+02	−1.02E+03	−1.08E+03	−1.07E+03	**−1.03E+03**	−1.17E+03
	SD	−6.59E+02	−9.91E+02	−1.04E+03	−7.18E+02	−1.17E+03	−1.08E+03	−9.47E+02	−1.02E+03	−1.08E+03	−1.07E+03	**−1.02E+03**	−1.17E+03
	%XPL	77.4735	1.9342	2.1311	98.3742	9.9359	15.8333	0.6335	6.0800	42.6055	6.2696	1.2032	2.3062
	%XPT	22.5265	98.0658	97.8689	1.6258	90.0641	84.1667	99.3665	93.9200	57.3945	93.7304	98.7968	97.6938
	AB	3.50E+01	3.56E+01	3.29E+01	2.40E+01	2.56E-03	4.08E+01	6.07E+01	4.52E+01	2.81E+01	2.81E+01	2.59E+01	**2.03E+01**
f_{23}	**MD**	1.36E+06	5.66E+05	−4.93E+03	1.19E+06	4.34E+05	9.24E+03	**−2.50E+02**	2.60E+05	7.59E+05	−3.10E+03	−1.02E+03	−3.77E+03
	SD	1.41E+06	5.20E+05	−4.93E+03	1.17E+06	4.22E+05	7.22E+03	**−2.10E+02**	1.22E+05	6.82E+05	−3.21E+03	−1.02E+03	−3.79E+03
	%XPL	96.0793	7.0659	2.1461	99.2405	30.4547	20.9524	20.5710	4.8452	42.1270	26.2275	4.9875	16.4037
	%XPT	3.9207	92.9341	97.8539	0.7595	69.5453	79.0476	79.4290	95.1548	57.8730	73.7725	95.0125	83.5963

(*continued on next page*)

Table 3 (*continued*)

Function		ABC	BA	CMAES	CSA	DE	FA	GWO	MFO	PSO	SSO	TLBO	WOA
AB	1.89E+05	2.13E+05	**5.24E-11**	1.72E+05	6.51E+04	7.72E+03	1.28E+02	2.79E+05	3.28E+05	3.52E+02	1.46E+02	2.91E+02	
f_{24}	**MD**	4.69E+04	5.34E+01	**2.25E-09**	1.40E+04	1.35E+01	2.16E+01	1.46E+01	7.41E+03	2.23E-01	1.98E+01	1.52E-02	1.16E+01
	SD	4.70E+04	5.32E+01	**6.88E-15**	1.32E+04	1.32E+01	2.16E+01	1.42E+01	4.72E-03	1.57E-01	1.97E+01	3.32E-04	1.14E+01
	%XPL	88.4427	5.6739	2.4577	97.6455	73.6445	17.4556	0.7390	4.6445	33.1138	4.5270	1.0758	5.6605
	%XPT	11.5573	94.3261	97.5423	2.3545	26.3555	82.5444	99.2610	95.3555	66.8862	95.4730	98.9242	94.3395

space. This behavior is shown in Fig. 3 where multiple exploration peaks appear along the optimization process. The multiple attraction points allow jumping in different zones even though it is continuously focusing on exploitation. This operation eventually permits to find better solutions than other methods.

Fig. 4 shows the balance levels of the worst-performing algorithms ABC and CSA. According to Fig. 4, it is clear that both schemes use excessive exploration in their search processes. Such an effect is visible from the incremental-decremental graph where its value increases during the whole optimization process. This lack of balance results in worse performance. It is not evident a direct relationship of this bad performance with their search mechanisms. However, it is clear that their well-known slow convergence as a consequence of their types of selection mechanisms and attraction operators could be responsible.

4.2. *Results of unimodal test functions*

This subsection details the analysis considering functions with only one optimum. Detailed information on each unimodal function can be found in Appendix [A.II] Table AII. Table 5 presents a comparison of the selected algorithms in unimodal functions. Table 6 presents the statistical analysis outcomes by applying Friedman test. The best results are highlighted in boldface. The results displayed in these tables suggest that the most prominent algorithm is WOA with an average rank of 1.

The second and third best algorithms in the unimodal test, TLBO and GWO, were compared directly to WOA in Fig. 5 to see the contrast between them. According to the incremental-decremental graph, all three methods on average exploited the search space above 98% of the time. The algorithm with the best results for all unimodal functions, WOA, spent 98.1318% of the time exploiting. The other algorithms, TLBO and GWO, differed very little, spending 98.9133% and 99.4177% of the time exploiting respectively. In terms of the characteristics of the algorithms, WOA utilizes a non-greedy selection mechanism and has a single attraction operator guided by the best solution. GWO employs a non-greedy selection mechanism. On the other hand, TLBO considers a greedy selection mechanism. Both methods GWO and TBL use as an attraction point the mean value of some individuals of the population to update its solutions. These three schemes maintain similar search mechanisms, as a consequence, they present similar performance in terms of solution quality. They also present very similar balances in exploration-exploitation such as it is shown in Fig. 5. It is important to point out that the WOA algorithm produces a rough balance response.

The worst four performing algorithms in unimodal functions are displayed in Fig. 6. ABC and CSA are once again in the bottom and both spent a high amount of time exploring without exploiting the search space. Although BA and MFO maintain a behavior very close to the one used by the top three algorithms, they present a bad performance in terms of the solution quality. In spite of their good balance level, it seems that is also important the search mechanisms obtain a better performance.

4.3. *Results of hybrid test functions*

Hybrid functions are formulations with complex behaviors that are produced from the combination of several multimodal functions. A specific implementation of composite functions can be consulted in Ref. [38]. Detailed information on each hybrid function can be found in Appendix [A.III] Table AIII. Table 7 presents a comparison of the selected algorithms in hybrid functions. This table is consistent with the results from previous tests. Table 8 presents the statistical analysis outcomes by applying Friedman test. The best results are highlighted in boldface. Considering both tables, WOA and TLBO were the algorithms that had the best performance, with CMA-ES being the distant third.

The results from the aforementioned trio of algorithms are averaged and displayed in Fig. 7. The algorithms show consistency in balance employed when compared with previous tests. According to the

Table 4
Ranks obtained and *p*-value of the Friedman test in multimodal functions.

Function	ABC	BA	CMAES	CSA	DE	FA	GWO	MFO	PSO	SSO	TLBO	WOA
Average rank	11.3333	8.9167	3.5833	10.9583	4.6875	6.1458	4.4583	7.4583	7.3125	6.0417	3.4792	3.625
***p*-value**	6.94e-28											

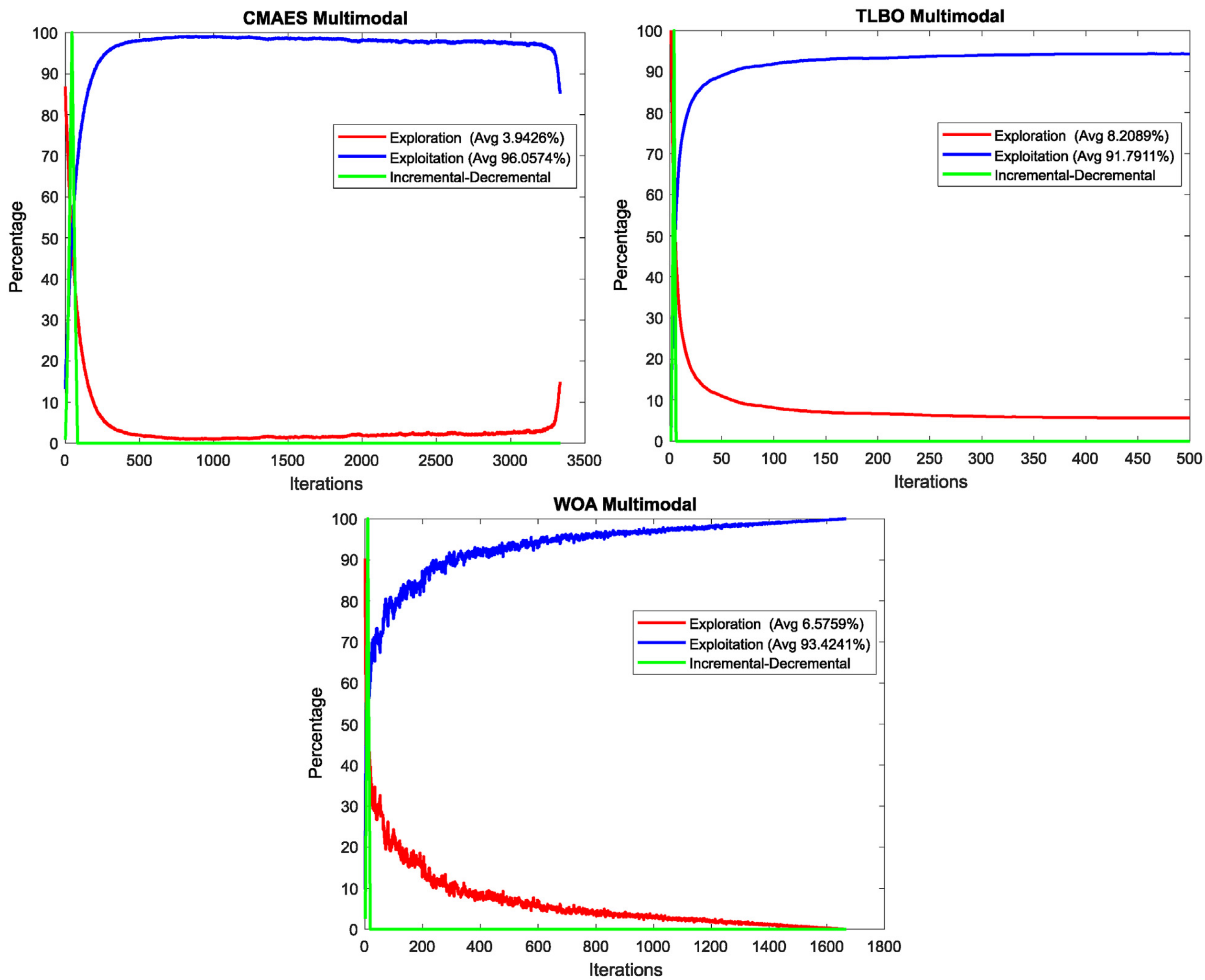

Fig. 2. Average balance employed by the top three algorithms in multimodal functions.

incremental-decremental graph, all schemes spent above 90% of the runtime exploiting the search space. In almost all hybrid functions the WOA and TLBO algorithms managed to find the global optimum. The only significant difference can be seen in function f_{30} where WOA surpass the other algorithms by a large margin, and f_{32} where CMA-ES found the best solutions with the least standard deviation. As previously mentioned, the reason for this difference is the search mechanisms used for exploration and exploitation. It is important to point out that the WOA algorithm once again produces a rough balance response.

Fig. 8 contains the evolution of the balance employed by the bottom three algorithms in hybrid functions. Once again, according to the incremental-decremental graph, ABC and CSA utilized high amounts of exploration to detrimental results. The BA balance between exploration and exploitation was close to the one employed by WOA, the best performing algorithm in hybrid functions. This is a good example of how the difference in the quality of the specific search mechanism of each algorithm affects greatly the performance, even though the compared algorithms invest the same amount of time exploring and exploiting the search space.

4.4. Results of shifted test functions

To see how the algorithms are affected when the problems to solve don't have their global optimum in positions close to zero, shifted functions are utilized. These are known functions but shifted to the left or right to have their global optimum in a different position. A specific implementation of shifted functions can be consulted in Ref. [38]. Detailed information of each shifted function can be found in Appendix [A.IV] Table AIV. Table 9 presents a comparison of the selected algorithms in shifted functions. Table 10 presents the statistical analysis outcomes by applying Friedman test. The best results are highlighted in boldface. Of the top algorithms from the previous tests, only CMA-ES was impervious to shifted functions and won in 7 of the 9 functions with an average rank of 1.8889. WOA and GWO suffered greatly on this set of functions while TLBO still suffered but less so. This allowed DE to be the one with the best performance in f_{34}, while TLBO still managed to be the

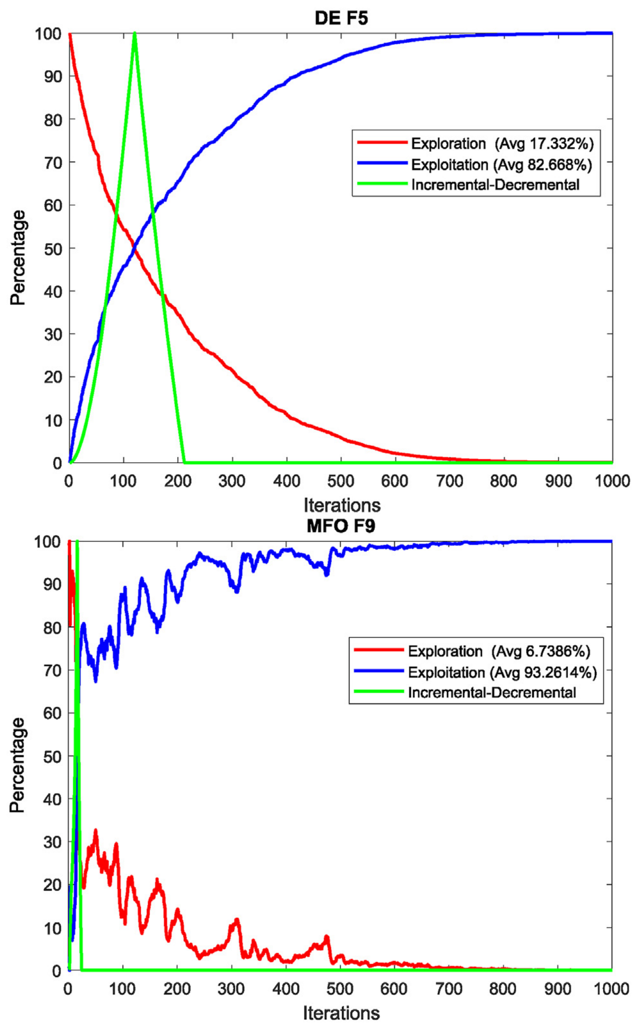

Fig. 3. Balance employed by DE in f_5 and f_{21}, and MFO in f_9 and f_{10}.

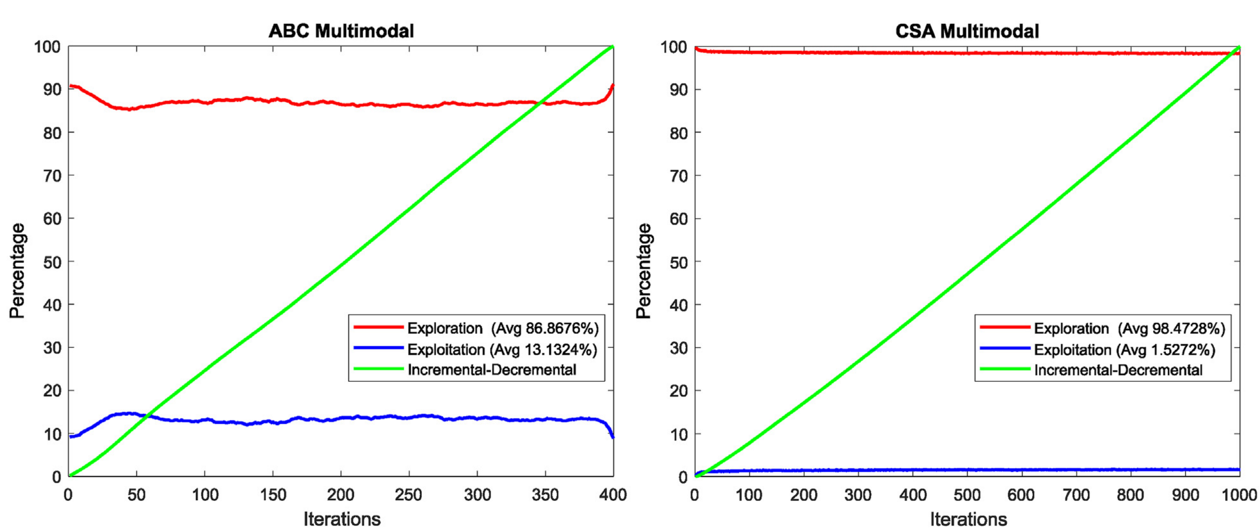

Fig. 4. Average balance employed by the bottom two algorithms in multimodal functions.

Table 5
The best solutions found and balance employed in each unimodal function.

Function		ABC	BA	CMAES	CSA	DE	FA	GWO	MFO	PSO	SSO	TLBO	WOA
f_{25}	**AB**	7.84E+04	3.67E+04	9.47E-37	1.18E+05	1.21E-03	7.10E+01	4.02E-100	1.23E+04	3.38E+04	1.88E+00	7.60E-86	**2.36E-250**
	MD	7.85E+04	3.33E+04	5.29E-37	1.16E+05	1.20E-03	6.88E+01	4.98E-101	4.29E+03	3.01E+04	1.70E+00	4.10E-86	**7.96E-265**
	SD	9.04E+03	1.42E+04	9.32E-37	1.46E+04	2.75E-04	1.17E+01	1.54E-99	1.87E+04	2.39E+04	7.23E-01	9.83E-86	**0.00E+00**
	%XPL	79.9258	2.0399	2.2269	98.8173	10.1382	16.6425	0.4699	6.7960	39.3438	7.1850	1.0403	1.7461
	%XPT	20.0742	97.9601	97.7731	1.1827	89.8618	83.3575	99.5301	93.2040	60.6562	92.8150	98.9597	98.2539
f_{26}	**AB**	2.85E+06	1.59E+06	5.43E-34	3.99E+06	2.75E-02	2.96E+03	4.42E-98	1.09E+06	1.69E+06	2.21E+02	4.32E-84	**7.05E-251**
	MD	2.85E+06	1.36E+06	2.32E-34	4.00E+06	2.77E-02	2.97E+03	1.24E-99	1.09E+06	1.57E+06	1.58E+02	2.15E-84	**1.51E-264**
	SD	4.42E+05	7.66E+05	8.08E-34	5.07E+05	7.20E-03	4.48E+02	1.00E-97	9.16E+05	8.83E+05	1.93E+02	9.66E-84	**0.00E+00**
	%XPL	89.6177	3.6873	2.3025	98.3686	11.9995	19.2656	0.5263	5.9581	37.2827	6.0095	1.0183	1.6362
	%XPT	10.3823	96.3127	97.6975	1.6314	88.0005	80.7344	99.4737	94.0419	62.7173	93.9905	98.9817	98.3638
f_{27}	**AB**	3.86E+01	1.54E-05	2.45E-43	5.12E+01	6.33E-07	3.14E-02	2.25E-103	6.99E+00	1.75E+00	9.50E-02	5.25E-89	**4.47E-250**
	MD	4.03E+01	1.54E-05	1.96E-43	5.13E+01	5.77E-07	3.14E-02	1.74E-104	1.52E-07	2.13E-06	9.43E-02	2.49E-89	**1.08E-267**
	SD	6.69E+00	2.27E-06	1.81E-43	7.05E+00	2.10E-07	3.60E-03	4.39E-103	1.53E+01	6.65E+00	1.43E-02	7.54E-89	**0.00E+00**
	%XPL	77.6193	1.8221	1.9656	98.5818	9.6177	14.8480	0.4756	5.5791	43.1609	26.0592	0.9471	1.7297
	%XPT	22.3807	98.1779	98.0344	1.4182	90.3823	85.1520	99.5244	94.4209	56.8391	73.9408	99.0529	98.2703
f_{28}	**AB**	1.85E+03	1.16E+01	9.94E-41	2.47E+03	2.85E-05	1.58E+00	9.20E-102	4.43E+02	8.23E+02	1.94E+00	3.41E-87	**1.25E-253**
	MD	1.80E+03	4.09E+00	6.99E-41	2.54E+03	2.73E-05	1.59E+00	8.48E-103	2.50E+02	8.00E+02	1.97E+00	1.40E-87	**2.19E-267**
	SD	2.54E+02	1.70E+01	9.10E-41	3.71E+02	1.06E-05	1.95E-01	3.68E-101	5.50E+02	5.24E+02	4.85E-01	4.08E-87	**0.00E+00**
	%XPL	81.7367	3.4067	2.1688	98.4859	10.7318	16.4721	0.4746	5.8278	40.2838	16.9878	0.9162	1.9663
	%XPT	18.2633	96.5933	97.8312	1.5141	89.2682	83.5279	99.5254	94.1722	59.7162	83.0122	99.0838	98.0337
f_{29}	**AB**	7.55E-02	1.30E-08	5.08E-09	7.70E-03	1.31E-22	6.31E-08	**0.00E+00**	8.48E-19	3.51E-13	3.26E-05	2.44E-197	**0.00E+00**
	MD	7.33E-02	1.12E-08	4.28E-09	7.86E-03	3.35E-23	5.47E-08	**0.00E+00**	2.90E-23	6.81E-16	3.22E-05	5.98E-199	**0.00E+00**
	SD	3.51E-02	8.50E-09	2.89E-09	4.56E-03	3.62E-22	4.56E-08	**0.00E+00**	4.19E-18	1.88E-12	1.79E-05	0.00E+00	**0.00E+00**
	%XPL	95.8100	5.5193	0.2584	98.8009	27.2130	27.8819	0.9651	11.4744	52.5785	54.9875	1.5114	2.2625
	%XPT	4.1900	94.4807	99.7416	1.1991	72.7870	72.1181	99.0349	88.5256	47.4215	45.0125	98.4886	97.7375

one with the best performance in f_{36}.

To see the impact that this set of functions has in the best algorithms from previous tests, Fig. 9 shows the evolution of the average balance in shifted functions for CMA-ES, GWO, TLBO and WOA. CMA-ES demonstrated consistency with the other test in the balance utilized, and it also demonstrated consistency with the results obtained. The case of GWO and WOA is interesting, these algorithms cannot reach the performance displayed in previous tests. They cannot also obtain the exploration/exploitation balances demonstrated with previous functions. Both methods, according to the incremental-decremental graph present a prolonged exploration effect in contrast with CMA-ES and TLBO. This suggests that the balance achieved by these two algorithms is not predetermined, but it depends on the nature of their search mechanisms. Due to its search strategy, it is well-known that GWO and WOA maintain a good performance in problems where the optimal point is in the origin. However, they present critical flaws when they face test functions with a shifted optimal point. Therefore, if GWO and WOA optimize problems with an optimal value centered in the origin, they are able to rapidly find the zone with the global solution and then focus on exploiting it, resulting in exploiting rates of over 90%. However, when the functions are shifted, they cannot find quickly enough the zone with the global solution. Under such conditions, their exploiting rate is degraded finding solutions of low quality in comparison with non-shifted functions. On the contrary, TLBO maintains the same balance between exploration and exploitation from previous test functions. Nevertheless, its performance in terms of quality is slightly lower than the non-shifted functions. This seems to indicate that the TLBO algorithm maintains a fixed balance without considering the function type. It is important to point out that the WOA algorithm produces also a rough balance response.

Fig. 10 shows the evolution of the balance obtained by DE in f_{34} and the balance performed by TLBO in f_{36}. These problems represent the only functions where CMA-ES does not obtain the best performance. For this reason, they are analyzed individually. In case of function f_{34}, most of the algorithms reach a very close performance in comparison with the already obtained in case of the non-shifted versions. Nevertheless, TLBO, GWO and WOA are unable to get their previous performance. Under such circumstances, DE becomes the best one. As it can be seen from the incremental-decremental graph, the exploration effect lasts around 170 iterations from a total of 1000. In case of functionf_{36}, even though the TLBO algorithm does not reach the same performance obtained by the non-shifted version of the function, it maintains a good enough performance for function f_{36}. In this function, according to the incremental-decremental graph, the TLBO scheme maintains a strong exploitation effect.

Lastly, Fig. 11 illustrates the evolution of the balance utilized by the bottom three algorithms in shifted functions. The pattern and algorithms in the bottom three are consistent with previous results.

4.5. *The balance response of the Random Search algorithm*

To test for a possible bias towards exploration or exploitation, an experiment has been conducted. In the test, the Random Search algorithm [39] which generates a population with a fixed standard deviation in each iteration has been analyzed. Under this scheme, it is possible to see if there is any sign of bias when a fixed diversity is employed. In Table 11, **it is shown the** maximum diversity, mean diversity and exploration and exploitation rate considering function f_{27}. In the test, the Random Search algorithm has been set with a standard deviation of 2.88 for each random movement considering also 1000 iterations.

From Fig. 12, it can be seen how the fixed standard deviation of the algorithm affects significantly the measured balance. The distance between search agents is always inside the standard deviation of the random movement. This causes the population diversity to maintain almost the same value in all iterations. As the diversity in each iteration is close to the standard deviation, which is also close to the maximum diversity, an average exploration rate of 95.5237 is produced. Under

Table 6
Ranks achieved and *p*-value of the Friedman test in unimodal functions.

Function	ABC	BA	CMAES	CSA	DE	FA	GWO	MFO	PSO	SSO	TLBO	WOA
Average rank	11.2000	8.2000	4.60000	11.8000	4.8000	7.2000	2	8	8.8000	7.4000	3	1
***p*-value**	5.20e-07											

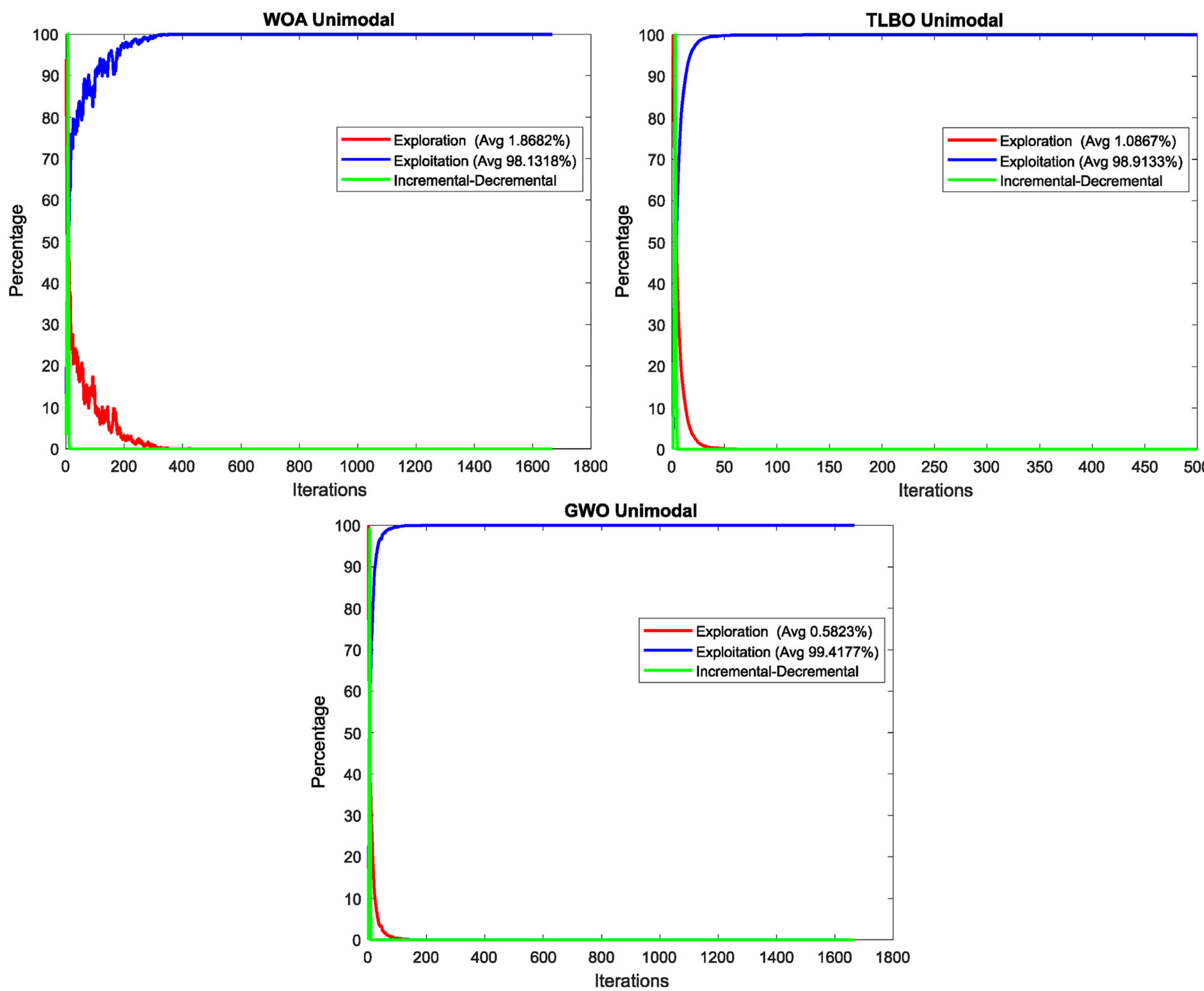

Fig. 5. Average balance employed by the top three algorithms in unimodal functions.

such conditions, as it is shown by the incremental-decremental graph, the exploration effect is maintained during the complete process.

The Random Search algorithm does not implement any other search mechanism. It produces new solutions through a random re-initialization. According to this experiment, it is apparent that no bias is influencing the results of the metric utilized.

4.6. Diversity analysis

In order to complement the analysis, an experimental test of the diversity over the best-performing algorithms CMA-ES, GWO, TLBO and WOA is conducted. In the experiment, the diversity defined in Eq. (2) is calculated and reported during the optimization of a determined benchmark function. In comparison, the functions f_6 and f_{37} are considered. These functions have been selected for being representative of the different behaviors presented in multimodal optimization. Fig. 13 shows the evolution of diversity during the optimization procedure for functions f_6 and f_{37} illustrated in Fig. 13(a) and (b), respectively. The axis x corresponds to the number of iterations, and the axis y the diversity measure.

After an analysis of Fig. 13, it is clear that all algorithms begin with a big diversity as a consequence of their random initialization. As the

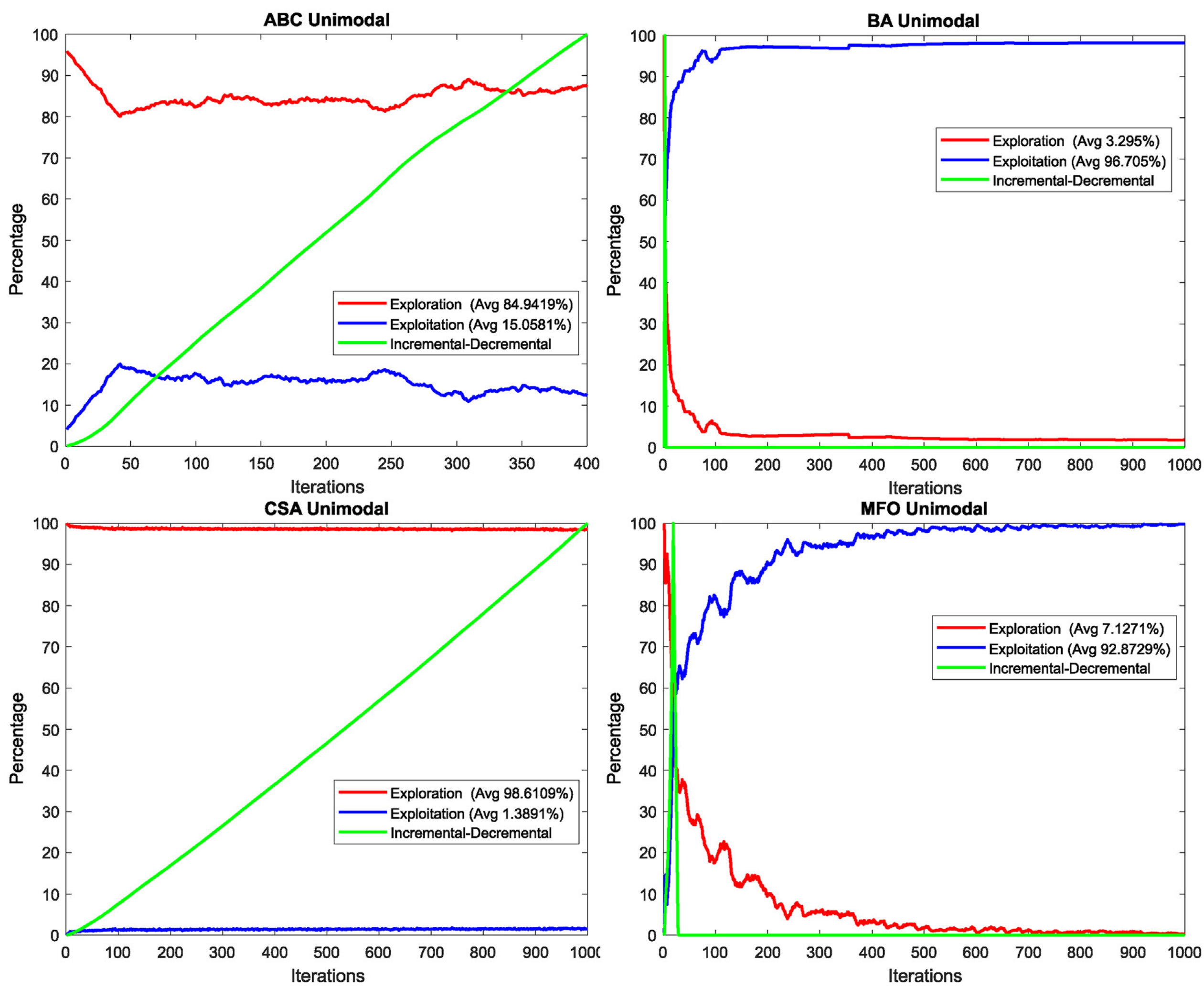

Fig. 6. Average balance employed by the bottom four algorithms in unimodal functions.

Table 7
Best solutions found and balance employed in each hybrid function.

Function		ABC	BA	CMAES	CSA	DE	FA	GWO	MFO	PSO	SSO	TLBO	WOA
f_{30}	**AB**	8.57E+04	1.12E+04	3.68E-19	1.99E+04	4.42E-03	1.95E+01	2.39E-58	2.01E+04	2.48E+04	2.92E-01	1.35E-43	**2.33E-175**
	MD	7.82E+04	8.84E+03	2.45E-19	2.05E+04	4.38E-03	1.98E+01	1.85E-58	2.01E+04	2.01E+04	2.88E-01	1.16E-43	**7.02E-181**
	SD	2.77E+04	7.55E+03	4.81E-19	2.33E+03	7.50E-04	1.94E+00	2.24E-58	1.86E+04	1.53E+04	4.99E-02	9.99E-44	**0.00E+00**
	%XPL	84.7161	4.8273	3.4316	98.5073	10.4171	16.2454	0.4767	9.1888	44.0224	8.2539	1.0883	1.8183
	%XPT	15.2839	95.1727	96.5684	1.4927	89.5829	83.7546	99.5233	90.8112	55.9776	91.7461	98.9117	98.1817
	AB	7.98E+02	4.00E+02	8.22E+01	7.21E+02	2.90E+01	7.98E+01	2.90E+01	1.80E+02	1.55E+02	7.31E+01	**2.90E+01**	**2.90E+01**
f_{31}	**MD**	7.98E+02	4.03E+02	7.78E+01	7.18E+02	2.90E+01	7.61E+01	2.90E+01	1.18E+02	8.95E+01	7.64E+01	**2.90E+01**	**2.90E+01**
	SD	8.92E+01	6.50E+01	1.84E+01	6.35E+01	2.28E-02	2.05E+01	5.66E-05	1.27E+02	1.26E+02	1.63E+01	**3.30E-12**	7.57E-05
	%XPL	88.0123	2.9059	2.2306	98.6365	15.2593	18.9380	0.5138	5.8204	42.3145	7.3259	1.1961	2.5734
	%XPT	11.9877	97.0941	97.7694	1.3635	84.7407	81.0620	99.4862	94.1796	57.6855	92.6741	98.8039	97.4266
	AB	2.23E+08	2.62E+06	**3.20E+01**	7.13E+07	4.76E+02	2.45E+02	3.20E+01	7.57E+01	2.40E+02	3.46E+01	**3.20E+01**	7.18E+01
f_{32}	**MD**	2.19E+08	1.50E+06	**3.20E+01**	7.08E+07	4.50E+02	2.45E+02	3.20E+01	7.18E+01	2.31E+02	3.45E+01	**3.20E+01**	6.77E+01
	SD	5.51E+07	2.66E+06	**1.73E-14**	1.92E+07	7.16E+01	4.54E+01	7.26E-06	1.59E+01	7.26E+01	8.49E-01	2.09E-10	3.31E+01
	%XPL	94.5070	3.4629	2.4833	98.4313	20.8563	18.7752	0.5362	7.1578	46.5950	11.0359	1.2826	4.9243
	%XPT	5.4930	96.5371	97.5167	1.5687	79.1437	81.2248	99.4638	92.8422	53.4050	88.9641	98.7174	95.0757
	AB	5.54E+03	5.84E+02	7.82E+01	9.21E+02	2.90E+01	5.95E+01	**2.90E+01**	8.95E+02	9.92E+02	1.63E+02	**2.90E+01**	**2.90E+01**
f_{33}	**MD**	5.51E+03	4.89E+02	7.73E+01	9.25E+02	2.90E+01	5.16E+01	**2.90E+01**	8.96E+02	9.24E+02	1.58E+02	**2.90E+01**	**2.90E+01**
	SD	3.22E+03	3.15E+02	2.35E+01	1.13E+02	9.70E-03	2.50E+01	7.14E-15	5.67E+02	5.46E+02	4.43E+01	4.12E-15	**2.64E-15**
	%XPL	90.3856	5.0770	3.6581	98.5816	12.9629	17.6859	0.5115	5.5055	46.0492	10.8563	1.2409	1.5329
	%XPT	9.6144	94.9230	96.3419	1.4184	87.0371	82.3141	99.4885	94.4945	53.9508	89.1437	98.7591	98.4671

Table 8
Ranks achieved and *p*-value of the Friedman test in hybrid functions.

Function	ABC	BA	CMAES	CSA	DE	FA	GWO	MFO	PSO	SSO	TLBO	WOA
Average rank	12	9	4.5000	10.2500	5.5000	6.5000	2.6250	8.5000	9.2500	5.5000	2.1250	2.2500
***p*-value**	1.12e-04											

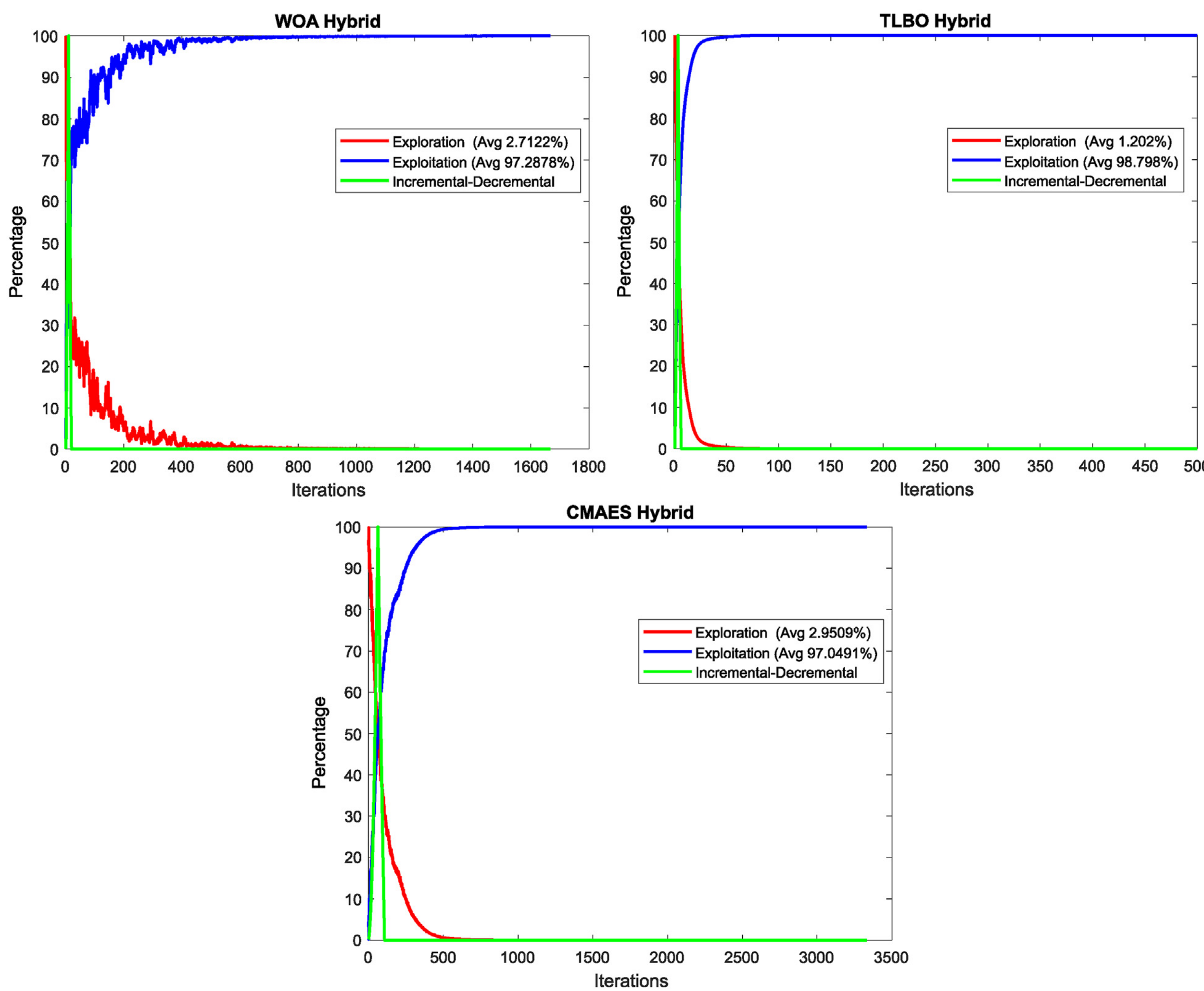

Fig. 7. Average balance employed by the top three algorithms in hybrid functions.

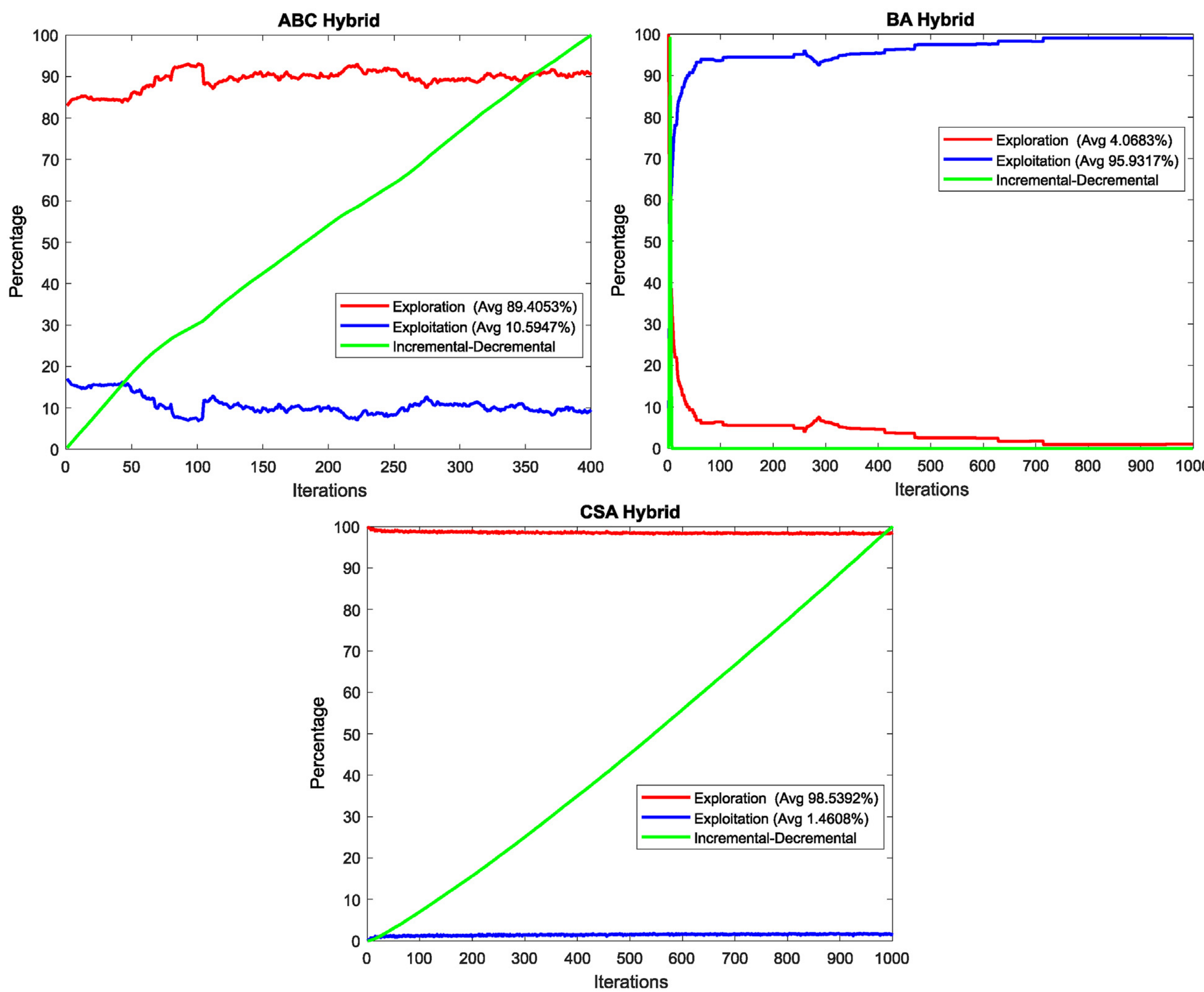

Fig. 8. Average balance employed by the bottom three algorithms in hybrid functions.

Table 9
Best solutions found and balance employed in each shifted function.

Function		ABC	BA	CMAES	CSA	DE	FA	GWO	MFO	PSO	SSO	TLBO	WOA
f_{34}	**AB**	1.73E+01	1.31E+01	2.10E+00	1.73E+01	**5.06E-03**	1.82E+00	7.18E+00	7.00E+00	7.51E+00	4.06E-01	4.74E+00	5.43E+00
	MD	1.74E+01	1.31E+01	**7.11E-15**	1.74E+01	4.65E-03	1.83E+00	8.29E+00	8.29E-01	7.25E-01	3.60E-01	4.55E+00	5.10E+00
	SD	4.73E-01	8.91E-01	6.40E+00	5.12E-01	**1.64E-03**	1.24E-01	3.64E+00	8.42E+00	8.29E+00	1.87E-01	9.31E-01	2.07E+00
	%XPL	87.1690	2.6385	2.0468	98.0358	12.9051	15.4772	13.5003	5.4988	43.1621	8.1621	1.2565	16.0797
	%XPT	12.8310	97.3615	97.9532	1.9642	87.0949	84.5228	86.4997	94.5012	56.8379	91.8379	98.7435	83.9203
	AB	2.87E+03	1.58E-02	**7.69E-05**	1.86E+03	9.25E+01	1.21E+00	2.20E+01	6.80E+02	1.52E+03	3.05E+00	1.09E-02	1.64E+00
f_{35}	**MD**	2.81E+03	1.11E-02	**7.60E-05**	1.81E+03	9.12E+01	1.05E+00	1.72E+01	2.22E+02	4.84E+02	2.75E+00	8.48E-03	1.24E+00
	SD	6.08E+02	1.64E-02	**2.92E-05**	4.86E+02	2.55E+01	7.00E-01	1.43E+01	1.03E+03	1.60E+03	1.40E+00	7.73E-03	1.23E+00
	%XPL	92.0951	6.0810	0.7667	98.9009	64.1413	23.1880	29.9811	5.0130	26.4855	30.5526	1.6696	26.0660
	%XPT	7.9049	93.9190	99.2333	1.0991	35.8587	76.8120	70.0189	94.9870	73.5145	69.4474	98.3304	73.9340
	AB	3.01E+02	6.01E+01	7.19E+01	2.61E+02	8.34E+01	5.82E+01	1.30E+02	1.40E+02	1.35E+02	5.89E+01	**1.76E+01**	4.10E+01
f_{36}	**MD**	3.01E+02	5.47E+01	5.22E+01	2.62E+02	8.24E+01	5.21E+01	8.18E+01	1.35E+02	1.38E+02	5.87E+01	**1.71E+01**	1.75E+01
	SD	1.26E+01	2.63E+01	5.23E+01	1.41E+01	8.56E+00	1.93E+01	8.68E+01	2.96E+01	3.58E+01	1.24E+01	**4.51E+00**	5.01E+01
	%XPL	89.7716	1.3032	3.9072	97.9230	45.4431	24.7250	47.5097	3.2421	40.8761	23.8254	5.7889	27.1777
	%XPT	10.2284	98.6968	96.0928	2.0770	54.5569	75.2750	52.4903	96.7579	59.1239	76.1746	94.2111	72.8223
	AB	3.32E+05	3.07E+03	**7.77E-01**	5.08E+05	7.05E+01	7.15E+01	1.96E+02	6.00E+04	1.53E+05	1.11E+02	2.80E+01	1.35E+00
f_{37}	**MD**	3.43E+05	1.39E+03	**1.11E-01**	5.24E+05	7.02E+01	5.27E+01	1.48E+02	5.48E+04	1.63E+05	1.09E+02	2.62E+01	7.87E-01
	SD	8.43E+04	3.75E+03	**2.85E+00**	1.13E+05	1.82E+01	4.03E+01	1.65E+02	4.06E+04	6.35E+04	6.34E+01	8.04E+00	1.59E+00
	%XPL	93.8950	7.8154	2.7508	98.7681	19.1539	19.5105	48.0801	6.0512	19.8387	24.8886	1.6766	24.1191
	%XPT	6.1050	92.1846	97.2492	1.2319	80.8461	80.4895	51.9199	93.9488	80.1613	75.1114	98.3234	75.8809
	AB	7.82E+04	3.48E+04	**1.13E-27**	1.14E+05	1.12E-03	7.44E+01	7.44E+03	1.85E+04	3.41E+04	1.90E+00	6.36E-06	1.10E+02
f_{38}	**MD**	7.78E+04	3.24E+04	**1.21E-27**	1.15E+05	1.02E-03	7.59E+01	5.41E+03	8.59E+03	3.01E+04	1.94E+00	6.51E-07	9.19E+01
	SD	1.20E+04	1.14E+04	**2.77E-28**	1.79E+04	4.19E-04	1.11E+01	5.54E+03	2.34E+04	2.24E+04	6.15E-01	2.73E-05	6.93E+01
	%XPL	80.7225	5.2696	2.3067	98.7206	10.4484	16.7563	11.9413	6.4235	39.5787	7.3420	1.3278	15.7178
	%XPT	19.2775	94.7304	97.6933	1.2794	89.5516	83.2437	88.0587	93.5765	60.4213	92.6580	98.6722	84.2822
	AB	2.73E+06	1.59E+06	**4.33E-26**	4.18E+06	3.15E-02	2.90E+03	7.74E+05	1.60E+06	1.41E+06	2.60E+02	1.47E-04	2.76E+05
f_{39}	**MD**	2.72E+06	1.47E+06	**1.01E-26**	4.21E+06	3.16E-02	3.00E+03	4.45E+05	1.26E+06	1.25E+06	1.50E+02	5.01E-05	1.85E+04
	SD	3.81E+05	5.22E+05	**7.64E-26**	5.55E+05	1.04E-02	4.50E+02	8.60E+05	1.58E+06	1.11E+06	2.78E+02	2.73E-04	1.09E+06
	%XPL	89.2405	2.9254	2.6224	98.4350	12.6352	17.9422	31.7209	7.0813	33.1841	7.7982	1.2846	26.3112
	%XPT	10.7595	97.0746	97.3776	1.5650	87.3648	82.0578	68.2791	92.9187	66.8159	92.2018	98.7154	73.6888
	AB	2.22E+32	5.11E+29	**7.24E-03**	5.98E+29	3.16E-02	1.84E+01	2.89E+02	4.90E+02	7.83E+02	1.08E+04	5.15E-01	3.45E+19
f_{40}	**MD**	1.72E+31	8.92E+22	**4.62E-14**	1.76E+29	3.02E-02	1.75E+01	2.81E+02	5.00E+02	8.00E+02	3.31E+02	3.88E-03	7.32E+12
	SD	5.78E+32	2.80E+30	**3.97E-02**	8.94E+29	8.41E-03	3.31E+00	6.80E+01	2.07E+02	2.25E+02	4.03E+04	2.69E+00	1.61E+20
	%XPL	87.7802	1.9500	3.5449	98.2999	14.0029	23.9210	10.3306	6.6679	38.9743	15.4068	1.5227	12.8716
	%XPT	12.2198	98.0500	96.4551	1.7001	85.9971	76.0790	89.6694	93.3321	61.0257	84.5932	98.4773	87.1284
	AB	3.89E+01	1.52E-05	**7.87E-29**	4.89E+01	5.73E-07	3.17E-02	1.16E-02	2.62E+00	2.62E+00	1.09E-01	2.10E-09	1.07E-02
f_{41}	**MD**	3.84E+01	1.55E-05	**7.57E-29**	4.94E+01	5.70E-07	3.31E-02	1.11E-02	7.83E-08	1.97E-06	1.05E-01	2.22E-10	8.50E-03
	SD	6.80E+00	1.68E-06	**1.47E-29**	5.76E+00	1.44E-07	3.41E-03	3.88E-03	8.00E+00	8.00E+00	2.57E-02	6.54E-09	8.84E-03
	%XPL	79.9896	4.5125	2.1102	98.1520	9.6874	16.0075	47.7530	6.3734	39.7010	28.9529	1.1754	26.0626
	%XPT	20.0104	95.4875	97.8898	1.8480	90.3126	83.9925	52.2470	93.6266	60.2990	71.0471	98.8246	73.9374
	AB	1.74E+03	1.96E+01	**7.64E-28**	2.57E+03	2.79E-05	1.67E+00	1.70E+00	4.13E+02	5.90E+02	1.75E+00	6.04E-07	8.74E-01
f_{42}	**MD**	1.76E+03	9.75E+00	**7.45E-28**	2.54E+03	2.56E-05	1.67E+00	1.36E+00	2.50E+02	5.50E+02	1.80E+00	1.58E-08	6.74E-01
	SD	2.02E+02	2.59E+01	**2.89E-28**	3.58E+02	9.89E-06	2.47E-01	1.16E+00	4.74E+02	3.86E+02	3.80E-01	2.65E-06	8.38E-01
	%XPL	78.9622	2.7921	2.2211	97.9084	10.5863	17.3523	47.1852	5.9450	37.1032	18.8607	1.2538	26.1591
	%XPT	21.0378	97.2079	97.7789	2.0916	89.4137	82.6477	52.8148	94.0550	62.8968	81.1393	98.7462	73.8409

Table 10
Ranks achieved and *p*-value of the Friedman test in shifted functions.

Function	ABC	BA	CMAES	CSA	DE	FA	GWO	MFO	PSO	SSO	TLBO	WOA
Average rank	11.4444	7.4444	1.8889	11.5555	3.7778	4.5556	6.7778	8.6667	9	5.4444	2.4444	5
***p*-value**	2.25e-12											

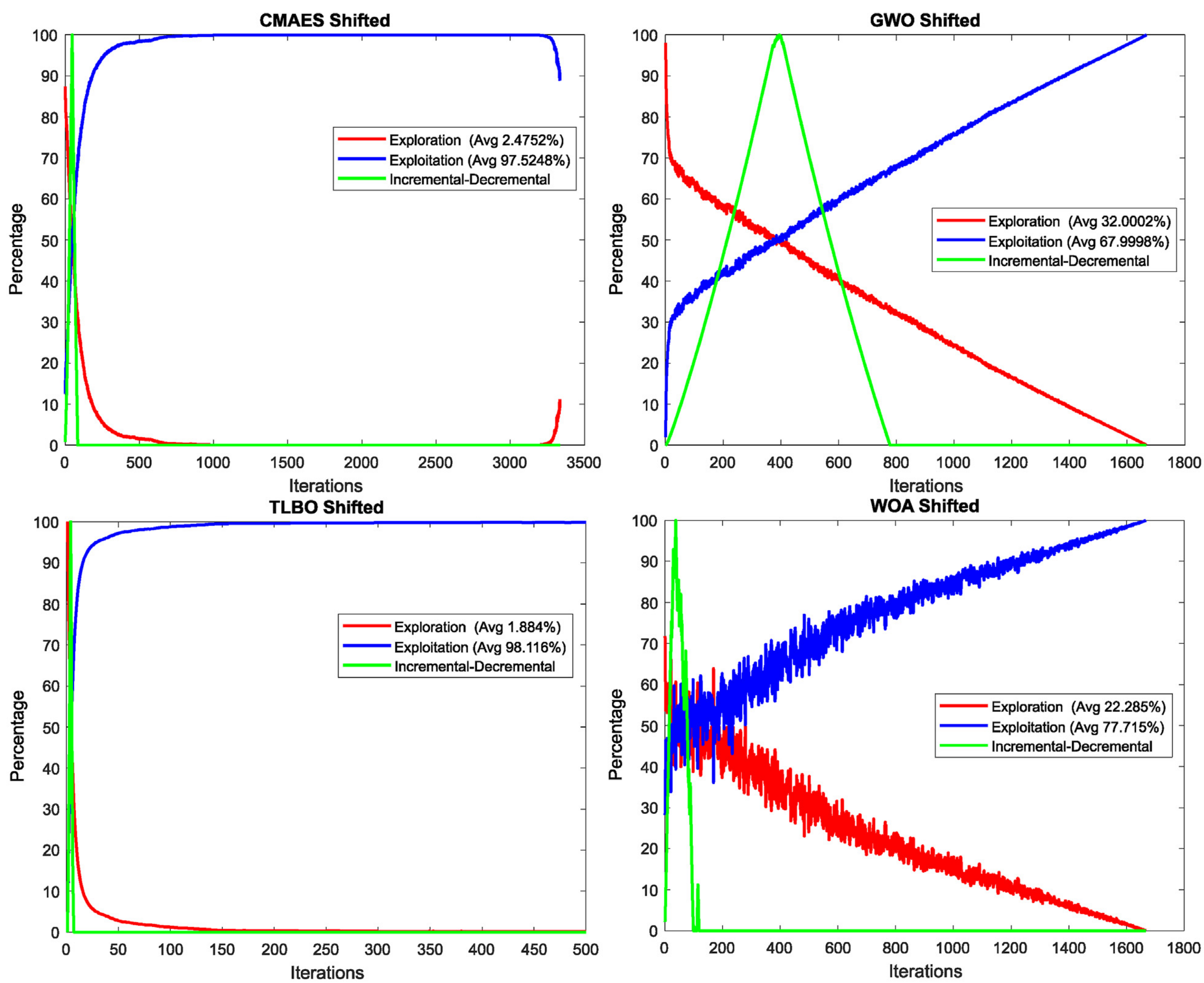

Fig. 9. Average balance employed by CMA-ES, GWO, TLBO and WOA algorithms in shifted functions.

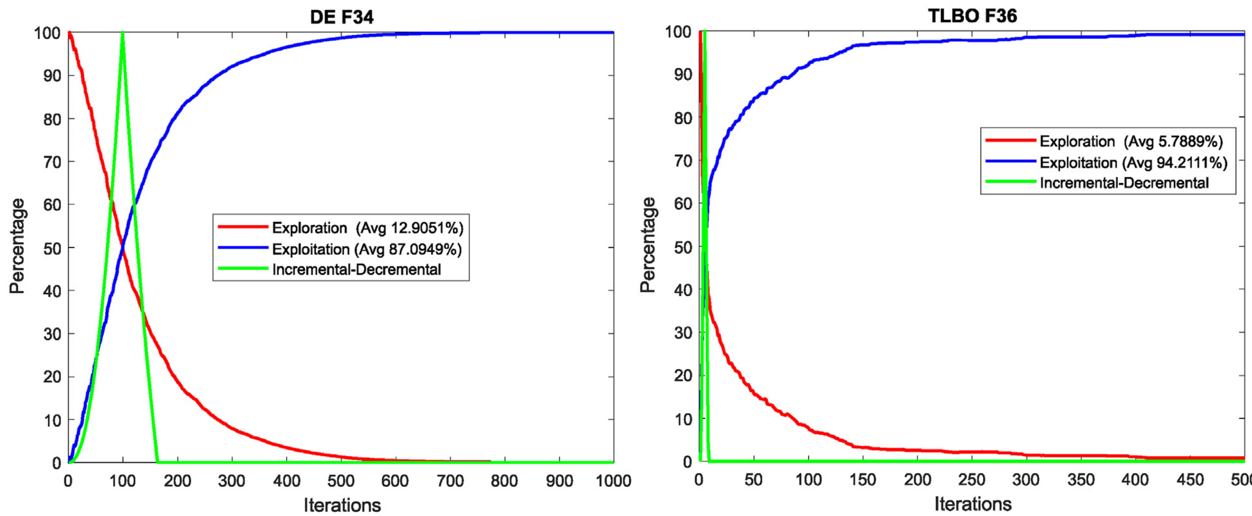

Fig. 10. Balance employed by DE in f_{34} and TLBO in f_{36}.

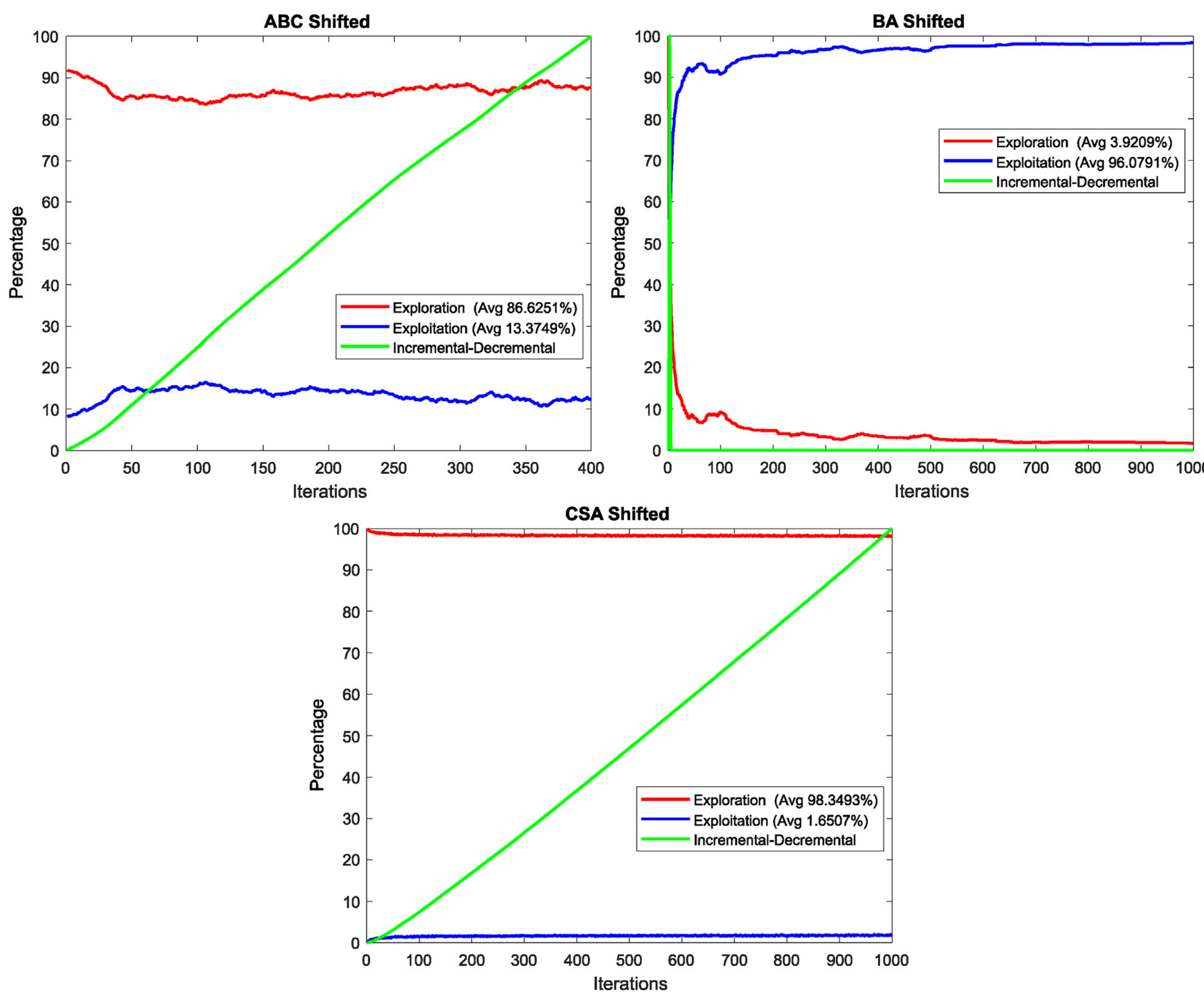

Fig. 11. Average balance employed by the bottom three algorithms in shifted functions.

Table 11
Results from the Random Search algorithm in the control test.

Div_{max}	Div_{mean}	%XPL	%XPT
2.57E+00	2.45E+00	95.52	4.48

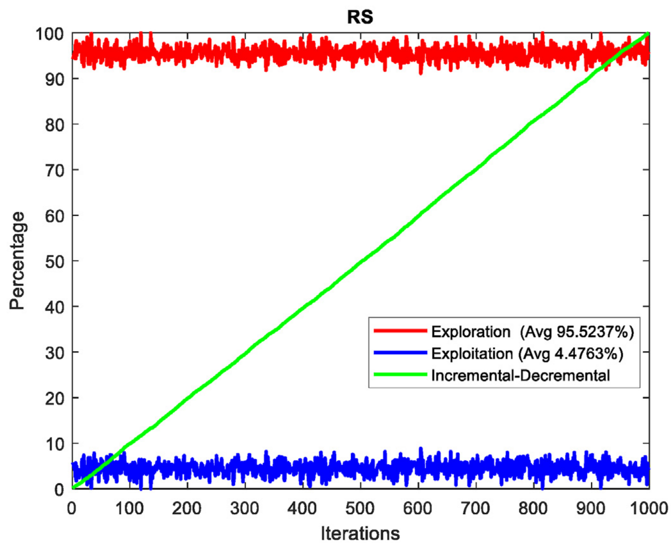

Fig. 12. The average balance of Random Search algorithm.

iterations increase, the population diversity diminishes. According to the Figure, the WOA algorithm presents in its behavior high oscillations which reflex its better conduction of exploration-exploitation. On the other hand, the other schemes present different diversity levels. CMA-ES and TLBO show the smoothest diversity responses.

5. Results analysis

According to this analysis, several interesting patterns have been identified. In the study, the best results have been obtained by WOA, CMA-ES, GWO and TLBO through an exploitation rate of over 90% ($XPL\% = 10, XPT\% = 90$). In general terms, from all schemes, the WOA maintains the best performance indexes.

From the experiments, it is clear that the best performances are obtained when the balances maintain a response of $XPL\% = 10$ and $XPT\% = 90$. In order to produce this behavior, the exploration process should last only the first iterations, specifically from 100 to 300 iterations. Then, the rest of the search strategy is mainly guided by the exploitation process.

An important characteristic of observable in balance graphs is the roughness. This feature implies an erratic variation of the balance during the evolution process. Such variations are produced by small and abrupt changes of diversity as a consequence of the used search mechanisms. These changes slightly increase the exploration rate allowing it to escape from local minima even though the optimization process is in the exploiting stage. In general terms, rough balance responses exhibit a better algorithm performance. The WOA algorithm maintains always a rough balance response (Figs. 2, 5 and 7) which seems to be one of the causes of its good performance.

Another interesting observation of this study is that good performance behavior is produced through the combination of competitive search mechanisms and an adequate balance response. During the experiments, in some cases, two metaheuristic algorithms presented very different performances in terms of solution quality in spite of their similar balance responses. Only an appropriate balance response ($XPL\% = 10$, $XPT\% = 90$) is not enough to obtain good results. It is also necessary to have operators for the generation of promising solutions so that they take advantage of the adequate diversity conditions observed in the balance response.

The results in shifted functions represent a valuable case to analyze since many algorithms could not maintain the good performance obtained in non-shifted functions. From the best performing algorithms, only CMA-ES has obtained its competitive performance in the shifted and non-shifted versions. GWO and WOA not only have presented a bad performance in terms of quality solution but also, an inappropriate balance response. Due to its search strategy, it is well-known that GWO and WOA maintain a good performance in problems where the optimal point is in the origin. However, they present critical flaws when they face test functions with a shifted optimal point. Under such conditions, when an algorithm cannot find a promising zone to exploit, it needs to invest more time exploring the search space producing an unbalance in the rate exploration-exploitation. This fact affects the quality or precision of the final solutions.

6. Conclusions and future work

In this paper, an empirical evaluation of the balance between

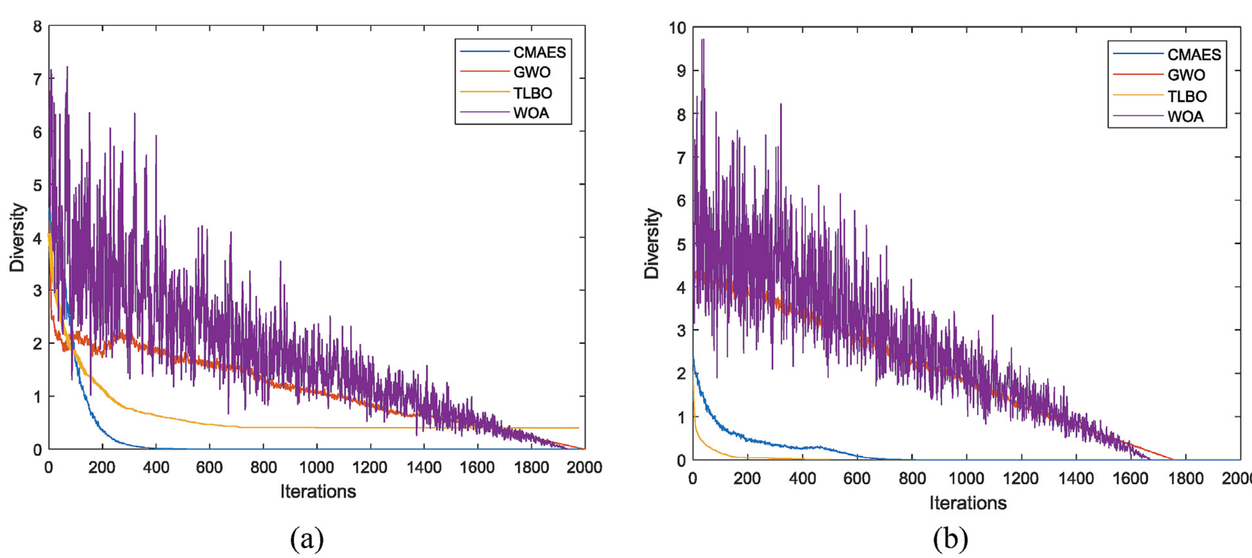

Fig. 13. Diversity behavior of the best performing algorithms CMA-ES, GWO, TLBO and WOA for functions (a) f_6 and (b) f_{37} .

exploration and exploitation on metaheuristic algorithms has been conducted. In the study, a dimension-wise diversity measurement is used to assess the balance of each scheme considering a representative set of 42 benchmark problems that involve multimodal, unimodal, composite and shifted functions. In the majority of the 42 functions (multimodal, unimodal, hybrid and shifted) the balance that produced the best results was above 90% exploitation and less than 10% exploration.

In shifted functions specifically, many algorithms maintain difficulties in maintaining the same performance that they had on to the non-shifted versions. Some of them had also problems to obtain the same balance between exploration and exploitation.

It has been observed in this study that good performance behavior is produced through the combination of competitive search mechanisms and an adequate balance response. Only an appropriate balance response is not enough to obtain good results. It is also necessary to have operators for the generation of promising solutions so that they take advantage of the adequate diversity conditions observed in the balance response.

According to this analysis, it can be formulated as a future work that the metaheuristic schemes could improve their results by gradually reducing the number of their search agents.

In general, a metaheuristic algorithm with N search agents invests N function evaluations in each iteration. In a competitive metaheuristic method, its exploitation phase lasts 90% of its execution. In this phase, the diversity of the population is small since most of the solutions are grouped or clustered in the best positions of the search space. Under such conditions, it is better to consider in the search strategy only the best B solutions from the existent N ($B \ll N$). Therefore in some cases, the computational complexity can be reduced.

CRediT authorship contribution statement

Bernardo Morales-Castañeda: Conceptualization, Investigation, Writing - original draft. **Daniel Zaldívar:** Resources, Data curation. **Erik Cuevas:** Resources, Writing - review & editing. **Fernando Fausto:** Visualization, Methodology. **Alma Rodríguez:** Software, Validation.

Appendix A

The set of benchmark test functions implemented in the experiments is described in Tables A.I, A.II, A.III and A.IV where $f(\mathbf{x}^*)$ is the optimum value of the function, $\mathbf{x}^*$ the optimum position and $S \in \mathbb{R}^n$ the search space. The benchmark test functions are classified in unimodal A.I, multimodal A.II, composite A.III and shifted A.IV.

Table A I
Multimodal test benchmark functions considered in the experiments.

f_i	Name	Function	S	Dim	Minimum
f_1	Ackley	$f(x) = -20e^{-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}} - e^{\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi x_i)} + 20 + e$	$[-30, 30]^n$	$n = 30$	$f(\mathbf{x}^*) = 0$; $\mathbf{x}^* = (0, \ldots, 0)$
f_2	Dixon Price	$f(x) = (x_i - 1)^2 + \sum_{i=2}^{n} i(2x_i^2 - x_{i-1})^2$	$[-10, 10]^n$	$n = 30$	$f(\mathbf{x}^*) = 0$; $\mathbf{x}^* = 2^{-\frac{2^i - 2}{2^i}}$ for $i = 1, \ldots, n$
f_3	Griewank	$f(x) = \frac{1}{4000}\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n}\cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	$[-600, 600]^n$	$n = 30$	$f(\mathbf{x}^*) = 0$; $\mathbf{x}^* = (0, \ldots, 0)$
f_4	Infinity	$f(x) = \sum_{i=1}^{n} x_i^6\left[\sin\left(\frac{1}{x_i}\right) + 2\right]$	$[-1, 1]^n$	$n = 30$	$f(\mathbf{x}^*) = 0$; $\mathbf{x}^* = (0, \ldots, 0)$
f_5	Levy	$f(x) = \cos2\ (\pi w_1) + \sum_{i=1}^{n-1}(w_i - 1)^2(1 + 10\sin\pi w_i + 1) + (w_n - 1)^2\ (1 + sin^2 2\pi w_n)$ $w_i = 1 + \left(\frac{x_i + 1}{4}\right)$	$[-10, 10]^n$	$n = 30$	$f(\mathbf{x}^*) = 0$; $\mathbf{x}^* = (1, \ldots, 1)$
f_6	Mishra11	$f(\mathbf{x}) = \left[\frac{1}{n}\sum_{i=1}^{n}\lvert x_i\rvert - \left(\prod_{i=1}^{n}\lvert x_i\rvert\right)^{\frac{1}{n}}\right]^2$	$[-10, 10]^n$	$n = 30$	$f(\mathbf{x}^*) = 0$; $\mathbf{x}^* = (0, \ldots, 0)$
f_7	Multimodal	$f(x) = \sum_{i}^{n}\lvert x_i\rvert \times \prod_{i}^{n}\lvert x_i\rvert$	$[-10, 10]^n$	$n = 30$	$f(\mathbf{x}^*) = 0$; $\mathbf{x}^* = (0, \ldots, 0)$
f_8	Penalty1	$f(x) = \frac{\pi}{30}\{10\sin^2(\pi y_1) + \sum_{i=1}^{n-1}(y_i - 1)^2[1 + 10\sin^2(\pi y_i + 1)] + (y_n - 1)^2\} + \sum_{i=1}^{n} u(x_i, 10, 100, 4)$; $y_i = 1 + \frac{x_i + 1}{4}$; $u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m, x_i > a \\ 0, -a \le x_i \le a \\ k(-x_i - a)^m, x_i < -a \end{cases}$	$[-50, 50]^n$	$n = 30$	$f(\mathbf{x}^*) = 0$; $\mathbf{x}^* = (-1, \ldots, -1)$
f_9	Penalty2	$f(x) = 0.1\{\sin^2(3\pi x_1) + \sum_{i=1}^{n-1}(x_i - 1)^2[1 + \sin^2(3\pi x_{i+1})] + (x_n - 1)^2[1 + \sin^2(2\pi x_n)]\} + \sum_{i=1}^{n} u(x_i, 5, 100, 4)$; $u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m, x_i > a \\ 0, -a \le x_i \le a \\ k(-x_i - a)^m, x_i < -a \end{cases}$	$[-50, 50]^n$	$n = 30$	$f(\mathbf{x}^*) = 0$; $\mathbf{x}^* = (1, \ldots, 1)$
f_{10}	Perm1	$f(x) = \sum_{k=1}^{n}\left\{\sum_{j}^{n}(j^k + \beta)\left[\left(\frac{x_j}{\beta}\right)^2 - 1\right]\right\}^2$	$[-n, n]^n$	$n = 30$	$f(\mathbf{x}^*) = 0$; $\mathbf{x}^* = (1, 2, \ldots, n)$
f_{11}	Perm2	$f(x) = \sum_{k=1}^{n}\left\{\sum_{j}^{n}(j^k + \beta)\left[x_j^k - \frac{1}{j}\right]\right\}^2$	$[-n, n]^n$	$n = 30$	$f(\mathbf{x}^*) = 0$; $\mathbf{x}^* = (1, 1/2, \ldots, 1/n)$
f_{12}	Plateau		$[-5.12, 5.12]^n$		

(continued on next page)

Table A I (*continued*)

f_i	Name	Function	S	Dim	Minimum
		$f(x) = 30 + \sum_{i=1}^{n} \lvert x_i \rvert$		$n = 30$	$f(\mathbf{x}^*) = 30$; $\mathbf{x}^* = (0, \ldots, 0)$
f_{13}	Powell	$f(x) = \sum_{i=1}^{n/4}[(x_{4i-3} + 10x_{4i-2})^2 + 5(x_{4i-1} - x_{4i})^2 + (x_{4i-2} - 2x_{4i-1})^4 + 10(x_{4i-3} - x_{4i})^4]$	$[-4, 5]^n$	$n = 30$	$f(\mathbf{x}^*) = 0$; $\mathbf{x}^* = (0, \ldots, 0)$
f_{14}	Qing	$f(x) = \sum_{i=1}^{n}(x_i^2 - 1)^2$	$[-500, 500]^n$	$n = 30$	$f(\mathbf{x}^*) = 0$; $x_i^* = (\pm\sqrt{i})$ i = 1,2,…,n
f_{15}	Quartic	$f(x) = \sum_{i=1}^{n}\{(ix_i)^4 + rand[0,1)\}$	$[-1.28, 1.28]^n$	$n = 30$	$f(\mathbf{x}^*) = 0$; $\mathbf{x}^* = (0, \ldots, 0)$
f_{16}	Quintic	$f(x) = \sum_{i=1}^{n}\lvert x_i^5 - 3x_i^4 + 4x_i^3 + 2x_i^2 - x_i - 4\rvert$	$[-10, 10]^n$	$n = 30$	$f(\mathbf{x}^*) = 0$; $\mathbf{x}^* = (-1, \ldots, -1)$
f_{17}	Rastringin	$f(x) = 10n + \sum_{i=1}^{n}[x_i^2 - 10\cos(2\pi x_i)]$	$[-5.12, 5.12]^n$	$n = 30$	$f(\mathbf{x}^*) = 0$; $\mathbf{x}^* = (0, \ldots, 0)$
f_{18}	Rosenbrock	$f(x) = \sum_{i=1}^{n-1}[100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$	$[-5, 10]^n$	$n = 30$	$f(\mathbf{x}^*) = 0$; $\mathbf{x}^* = (1, \ldots, 1)$
f_{19}	Schwefel21	$f(x) = \max\lvert x_i\rvert$	$[-100, 100]^n$	$n = 30$	$f(\mathbf{x}^*) = 0$; $\mathbf{x}^* = (0, \ldots, 0)$
f_{20}	Schwefel22	$f(x) = \sum_{i=1}^{n}\lvert x_i\rvert + \prod_{i=1}^{n}\lvert x_i\rvert$	$[-100, 100]^n$	$n = 30$	$f(\mathbf{x}^*) = 0$; $\mathbf{x}^* = (0, \ldots, 0)$
f_{21}	Schwefel26	$f(x) = -\sum_{i=1}^{n}[x_i sin(\sqrt{\lvert x_i\rvert})]$	$[-500, 500]^n$	$n = 30$	$f(\mathbf{x}^*) = -418.98n$; $\mathbf{x}^* = (420.97, ..., 420.97)$
f_{22}	Step	$f(x) = \sum_{i=1}^{n} x_i^2$	$[-100, 100]^n$	$n = 30$	$f(\mathbf{x}^*) = 0$; $\mathbf{x}^* = (0, \ldots, 0)$
f_{23}	Styblinski Tang	$f(x) = \frac{1}{2}\sum_{i=1}^{n}(x_i^4 - 16x_i^2 + 5x_i)$	$[-5, 5]^n$	$n = 30$	$f(\mathbf{x}^*) = -39.1659n$; $\mathbf{x}^* = (-2.90, \ldots, 2.90)$
f_{24}	Trid	$f(x) = \sum_{i=1}^{n}(x_i - 1)^2 - \sum_{i=2}^{n} x_i x_{i-1}$	$[-n^2, n^2]^n$	$n = 30$	$f(\mathbf{x}^*) = -n(n+4)(n-1)/6$; $x_i^* = i(n+1-i)$i = 1, 2,…,n

Table A II
Unimodal test benchmark functions considered in the experiments.

f_i	Name	Function	S	Dim	Minimum
f_{25}	Rothyp	$f(x) = \sum_{i=1}^{n}\sum_{j=1}^{i} x_j^2$	$[-65.536, 65.536]^n$	$n = 30$	$f(x^*) = 0$; $x^* = (0, ..., 0)$
f_{26}	$f(x) = \sum_{i=1}^{n}(\sum_{j=1}^{i} x_i)^2$	$[-100, 100]^n$	$n = 30$	$f(\mathbf{x}^*) = 0$; $\mathbf{x}^* = (0, \ldots, 0)$	
f_{27}	Sphere	$f(x) = \sum_{i=1}^{n} x_i^2$	$[-5, 5]^n$	$n = 30$	$f(\mathbf{x}^*) = 0$; $\mathbf{x}^* = 0, \ldots, 0$
f_{28}	Sum Squares	$f(x) = \sum_{i=1}^{n} ix_i^2$	$[-10, 10]^n$	$n = 30$	$f(\mathbf{x}^*) = 0$; $\mathbf{x}^* = (0, \ldots, 0)$
f_{29}	Sum of Different Powers	$f(x) = \sum_{i=1}^{n}\lvert x_i\rvert^{i+1}$	$[-1, 1]^n$	$n = 30$	$f(\mathbf{x}^*) = 0$; $\mathbf{x}^* = (0, \ldots, 0)$

Table A III
Composite test benchmark functions considered in the experiments.

f_i	Name	Function	S	Dim	Minimum
f_{30}	Hybrid1	$f(x) = f_1(\mathbf{x}) + f_{20}(\mathbf{x}) + f_{27}(\mathbf{x})$	$[-100, 100]^n$	$n = 30$	$f(\mathbf{x}^*) = 0$; $\mathbf{x}^* = (0, \ldots, 0)$
f_{31}	Hybrid2	$f(x) = f_3(\mathbf{x}) + f_{17}(\mathbf{x}) + f_{18}(\mathbf{x})$	$[-100, 100]^n$	$n = 30$	$f(\mathbf{x}^*) = n - 1$; $\mathbf{x}^* = (0, \ldots, 0)$
f_{32}	Hybrid3	$f(x) = f_1(\mathbf{x}) + f_9(\mathbf{x}) + f_{18}(\mathbf{x}) + f_{26}(\mathbf{x})$	$[-100, 100]^n$	$n = 30$	$f(\mathbf{x}^*) = (1.1n) - 1$; $\mathbf{x}^* = (0, \ldots, 0)$
f_{33}	Hybrid4	$f(x) = f_1(\mathbf{x}) + f_3(\mathbf{x}) + f_{17}(\mathbf{x}) + f_{18}(\mathbf{x}) + f_{20}(\mathbf{x})$	$[-100, 100]^n$	$n = 30$	$f(\mathbf{x}^*) = n - 1$; $\mathbf{x}^* = (0, \ldots, 0)$

Table A IV
Shifted test benchmark functions considered in the experiments.

f_i	Name	Function	S	Dim	Minimum
f_{34}	Shifted Ackley	$f(x) = -20e^{-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n}(X_i - 10)^2}} - e^{\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi(X_i - 10))} + 20 + e$	$[-20, 40]^n$	$n = 30$	$f(\mathbf{x}^*) = 0$; $\mathbf{x}^* = (10, \ldots, 10)$
f_{35}	Shifted Powell		$[1, 10]^n$		

(*continued on next page*)

Table A IV (*continued*)

f_i	Name	Function	S	Dim	Minimum
		$f(x) = \sum_{i=1}^{n/4}[((x_{4i-3}-5)+10x_{4i-2}-5))(x_{4i-2}-5))^2+5x_{4i-1}-5)-x_{4i}-5))(x_{4i}-5))^2+x_{4i-2}-5)-2(x_{4i-1}-5))^4+10((x_{4i-3}-5)-(x_{4i}-5))^4]$		$n=30$	$f(\mathbf{x}^*)=0$; $\mathbf{x}^*=(5,\ldots,5)$
f_{36}	Shifted Rastringin	$f(x) = 10n+\sum_{i=1}^{n}[(x_i-15)^2-10\cos(2\pi(x_i-15))]$	$[9.88, 20.12]^n$	$n=30$	$f(\mathbf{x}^*)=0$; $\mathbf{x}^*=(15,\ldots,15)$
f_{37}	Shifted Rosenbrock	$f(x) = \sum_{i=1}^{n-1}[100x_{i+1}-100)-(x_i-100)^2)(x_i-100)(x_i-100)^2)^2+(x_i-100)-1)^2]$	$[96, 111]^n$	$n=30$	$f(\mathbf{x}^*)=0$; $\mathbf{x}^*=(101,\ldots,101)$
f_{38}	Shifted Rothyp	$f(x) = \sum_{i=1}^{n}\sum_{j=1}^{i}(x_j-20)^2$	$[-45.536, 85.536]^n$	$n=30$	$f(\mathbf{x}^*)=0$; $\mathbf{x}^*=(20,\ldots,20)$
f_{39}	Shifted Schwefel2	$f(x) = \sum_{i=1}^{n}(\sum_{j=1}^{i}(x_i-100))^2$	$[0, 200]^n$	$n=30$	$f(\mathbf{x}^*)=0$; $\mathbf{x}^*=(100,\ldots,100)$
f_{40}	Shifted Schwefel22	$f(x) = \sum_{i=1}^{n}\lvert(x_i-25)\rvert+\prod_{i=1}^{n}\lvert(x_i-25)\rvert$	$[-75, 125]^n$	$n=30$	$f(\mathbf{x}^*)=0$; $\mathbf{x}^*=(25,\ldots,25)$
f_{41}	Shifted Sphere	$f(x) = \sum_{i=1}^{n}(x_i-20)^2$	$[14.88, 25.12]^n$	$n=30$	$f(\mathbf{x}^*)=0$; $\mathbf{x}^*=(20,\ldots,20)$
f_{42}	Shifted Sum2	$f(x) = \sum_{i=1}^{n}i(x_i-30)^2$	$[20, 40]^n$	$n=30$	$f(\mathbf{x}^*)=0$; $\mathbf{x}^*=(30,\ldots,30)$

References

[1] K. Hussain, M.N. Mohd Salleh, S. Cheng, Y. Shi, Metaheuristic research: a comprehensive survey, Artif. Intell. Rev. (2018) 1–43.
[2] K. Sörensen, Metaheuristics-the metaphor exposed, Int. Trans. Oper. Res. 22 (1) (2015) 3–18.
[3] J. Xu, J. Zhang, "Exploration-exploitation tradeoffs in metaheuristics: survey and analysis, in: Proc. 33rd Chinese Control Conf, CCC 2014, 2014, pp. 8633–8638.
[4] M. Črepinšek, S.-H. Liu, M. Mernik, Exploration and exploitation in evolutionary algorithms, ACM Comput. Surv. 45 (3) (Jun. 2013) 1–33.
[5] S. Chen, A. Boluf, J. Montgomery, T. Hendtlass, An Analysis on the Effect of Selection on Exploration in Particle Swarm Optimization and Differential Evolution, 2019, pp. 3037–3044.
[6] F. Fausto, A. Reyna-Orta, E. Cuevas, Á.G. Andrade, M. Perez-Cisneros, From Ants to Whales: Metaheuristics for All Tastes, Springer Netherlands, 2019.
[7] X.-S. Yang, Swarm-based metaheuristic algorithms and No-Free-Lunch theorems, Theory New Appl. (2012). Swarm Intell.
[8] X.-S. Yang, Nature-Inspired mateheuristic algorithms: success and new challenges, J. Comput. Eng. Inf. Technol. 1 (1) (2012).
[9] H.P. Kriegel, E. Schubert, A. Zimek, The (black) art of runtime evaluation: are we comparing algorithms or implementations? Knowl. Inf. Syst. 52 (2) (2017) 341–378.
[10] S. Cheng, Y. Shi, Q. Qin, Q. Zhang, R. Bai, Population Diversity Maintenance in Brain 4 (2) (2015) 83–97.
[11] T. Al-Quraishi, J.H. Abawajy, M.U. Chowdhury, S. Rajasegarar, A.S. Abdalrada, Recent Advances on Soft Computing and Data Mining, vol. 700, Springer International Publishing, Cham, 2018. Scdm.
[12] M.N.M. Salleh, et al., Exploration and exploitation measurement in swarm-based metaheuristic algorithms: an empirical analysis, in: Advances in Intelligent Systems and Computing, vol. 700, 2018, pp. 24–32. Scdm.
[13] X.S. Yang, S. Deb, S. Fong, Metaheuristic algorithms: optimal balance of intensification and diversification, Appl. Math. Inf. Sci. 8 (3) (2014) 977–983.
[14] X.S. Yang, S. Deb, T. Hanne, X. He, Attraction and diffusion in nature-inspired optimization algorithms, Neural Comput. Appl. 19 (2015).
[15] H. Du, Z. Wang, W.E.I. Zhan, Elitism and distance strategy for selection of evolutionary algorithms, IEEE Access 6 (2018) 44531–44541.
[16] R. Storn, K. Price, "Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces, J. Global Optim. 11 (4) (1997) 341–359.
[17] T. Huang, Z.-H. Zhan, X. Jia, H. Yuan, J. Jiang, J. Zhang, Niching community based differential evolution for multimodal optimization problems, in: 2017 IEEE Symposium Series on Computational Intelligence, SSCI, 2017, pp. 1–8.
[18] R. Poli, J. Kennedy, T. Blackwell, Particle swarm optimization, Swarm Intell. 1 (1) (Oct. 2007) 33–57.
[19] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey Wolf optimizer," Adv. Eng. Software., vol. 69, pp. 46–61, Mar. 2014.
[20] X.S. Yang, X. He, Firefly algorithm: recent advances and applications, Int. J. Swarm Intell. 1 (1) (2013) 1–14.
[21] E. Rashedi, H. Nezamabadi-pour, S. Saryazdi, GSA: a gravitational search algorithm, Inf. Sci. (Ny) 179 (13) (2009) 2232–2248.
[22] D. Karaboga, An idea based on honey bee swarm for numerical optimization, Tech. Rep. TR06 (2005) 10. Erciyes Univ., no. TR06.
[23] D. Karaboga, B. Basturk, A powerful and efficient algorithm for numerical function optimization: Artificial bee colony (ABC) algorithm, J. Global Optim. 39 (3) (2007) 459–471.
[24] X.-S. Yang, A New Metaheuristic Bat-Inspired Algorithm, 2010, pp. 65–74.
[25] N. Hansen, A. Ostermeier, Completely derandomized self-adaptation in evolution strategies, Evol. Comput. 9 (2) (Jun. 2001) 159–195.
[26] A. Askarzadeh, A novel metaheuristic method for solving constrained engineering optimization problems: Crow search algorithm, Comput. Struct. 169 (2016) 1–12.
[27] X.-S. Yang, Firefly algorithms for multimodal optimization, in: Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 5792, LNCS, 2009, pp. 169–178.
[28] E. Cuevas, M. Cienfuegos, D. Zaldívar, M. Pérez-cisneros, A swarm optimization algorithm inspired in the behavior of the social-spider, Expert Syst. Appl. 40 (16) (2013) 6374–6384.
[29] R.V. Rao, V.J. Savsani, D.P. Vakharia, Teaching-Learning-Based Optimization: an optimization method for continuous non-linear large scale problems, Inf. Sci. (Ny) 183 (1) (2012) 1–15.
[30] S. Mirjalili, Moth-flame optimization algorithm: a novel nature-inspired heuristic paradigm, Knowl. Base Syst. 89 (2015) 228–249.
[31] B. Chopard, M. Tomassini, Particle swarm optimization, Nat. Comput. Ser. (2018) 97–102.
[32] S. Mirjalili, A. Lewis, The Whale optimization algorithm, Adv. Eng. Software 95 (2016) 51–67.
[33] J. Derrac, S. García, D. Molina, F. Herrera, A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms, Swarm Evol. Comput. 1 (1) (2011) 3–18.
[34] N. Hansen, The CMA Evolution Strategy: A Tutorial 102 (2016) 75–102, 2006.
[35] S. Das, P.N. Suganthan, Differential evolution: a survey of the state-of-the-art, IEEE Trans. Evol. Comput. 15 (1) (2011) 4–31.
[36] I. Fister, X.S. Yang, J. Brest, A comprehensive review of firefly algorithms, Swarm Evol. Comput. 13 (2013) 34–46.
[37] F. Marini, B. Walczak, Particle swarm optimization (PSO). A tutorial, Chemometr. Intell. Lab. Syst. 149 (Dec. 2015) 153–165.
[38] N.H. Awad, M.Z. Ali, J.J. Liang, B.Y. Qu, P.N. Suganthan, Problem Definitions and Evaluation Criteria for the CEC 2017 Special Session and Competition on Single Objective Real-Parameter Numerical Optimization, 2016.
[39] H. Mania, A. Guy, B. Recht, Simple Random Search Provides a Competitive Approach to Reinforcement Learning, 2018, pp. 1–22.
[40] J. Carrasco, et al., Recent trends in the use of statistical tests for comparing swarm and evolutionary computing algorithms: Practical guidelines and a critical review, Swarm Evol. Comput. 54 (2020), https://doi.org/10.1016/j.swevo.2020.100665.
[41] J. Del Ser, et al., Bio-inspired computation: Where we stand and what's next, Swarm Evol. Comput. 48 (2019), https://doi.org/10.1016/j.swevo.2019.04.008.