Multimodal Optimization Using Niching Differential Evolution with Index-based Neighborhoods

Michael G. Epitropakis[*], Vassilis P. Plagianakos[†], and Michael N. Vrahatis[*]
[*]Computational Intelligence Laboratory, Department of Mathematics, University of Patras, GR-26110 Patras, Greece,
Email: {mikeagn,vrahatis}@math.upatras.gr
[†]Department of Computer Science and Biomedical Informatics, University of Central Greece, GR-35100 Lamia, Greece,
Email: vpp@ucg.gr

***Abstract*—A new family of Differential Evolution mutation strategies (DE/nrand) that are able to handle multimodal functions, have been recently proposed. The DE/nrand family incorporates information regarding the real nearest neighborhood of each potential solution, which aids them to accurately locate and maintain many global optimizers simultaneously, without the need of additional parameters. However, these strategies have increased computational cost. To alleviate this problem, instead of computing the real nearest neighbor, we incorporate an index-based neighborhood into the mutation strategies. The new mutation strategies are evaluated on eight well-known and widely used multimodal problems and their performance is compared against five state-of-the-art algorithms. Simulation results suggest that the proposed strategies are promising and exhibit competitive behavior, since with a substantial lower computational cost they are able to locate and maintain many global optima throughout the evolution process.**

I. Introduction

Handling multimodal functions is a very important and challenging task in evolutionary computation community, since most hard real-world problems exhibit highly multimodal landscapes. They are likely to have several global and/or local minima, and in many cases it is desirable to accurately locate as many as possible. To this end, several Evolutionary Algorithms (EAs) have been recently extended to handle such landscapes through the concept of the niche formation. Niche formation is a common biological phenomenon [1]. A niche can be defined as a subspace in the environment that can support different types of life. In general, niches indirectly impose reproduction restrictions to aid the differentiation of the species and thus maintain their diversity. Many natural environments can lead to niche formation, such as remote islands, high mountains and isolated valleys. Many well-known EAs have been developed, to mimic the biological niche formation and take advantage of its characteristics. Theses methodologies are characterized as Niching methods [1]. Niching methods tend to maintain the diversity within their population and allow a parallel convergence into multiple solutions. Various niching techniques have been proposed and successfully applied to different EAs, such as, crowding [2], [3], fitness sharing [3], [4], clearing [5], specialized evolution operators [6], clustering [7], stretching and deflation [8], [9], parallelization [10], restricted tournament selection [11], [12], and speciation [13].

In the paper at hand, we consider the Differential Evolution (DE) algorithm which has been proposed by Storn and Price [14]. DE has been successfully applied in a plethora of optimization problems [14]–[17]. In this work, the objective is to locate as many global optimizers of a multimodal function as possible. The DE literature includes several different variants that incorporate the aforementioned niching techniques and attempt to handle multimodal landscapes. In particular, Thomsen extends DE with both a crowding and a fitness sharing technique, namely Crowding DE (CDE) and Sharing DE [18], and shows that the CDE variant is a more promising approach, since outperforms the Sharing DE in all tested problems [18]. In turn, Species-based DE (SDE) [19], [20] incorporates the speciation concept to handle multimodal functions. SDE locates multiple global optima simultaneously through the adaptive formation of multiple species, which are evolved through DE. Although SDE is computationally more efficient than the Crowding DE, it incorporates a user-specified and problem dependent parameter called *species radius*, which should be properly chosen.

Additionally, DE using local selection (DELS) [21] employs a new mutation strategy that divides the mutation operation into the local and the global mutation stages. It selects a different mutation strategy, with a pre-specified probability, to perform either a global or a local mutation. The global mutation enhance the exploratory ability of the algorithm, while the local mutation its exploitative behavior. DELS has been further hybridized with a multi-start gradient-based local search, as well as with the crowding technique [22]. In turn, Zaharie proposed a parallel approach that utilizes an “island model” approach to locate in parallel many global optima [10], while in [12] a DE extension with an ensemble of the restricted tournament selection (ERTS-DE) has been proposed. Finally, several other EAs have been proposed, which attempt to handle multimodal landscapes [9], [23]–[26].

Recently, we have introduced a new family of mutation strategies that are able to efficiently handle multimodal functions, namely the DE/nrand family [6]. The DE/nrand family incorporates information regarding the real nearest neighborhood of each potential solution, which aids them to accurately locate and maintain many global optimizers, without the need of additional parameters. Nevertheless, the calculation of the distances between the individuals demands a high computational cost. To alleviate this problem, we incorporate an index-based neighborhood into the mutation strategies. To this end, we manage to reduce the computational cost of the mutation strategies and still be able to efficiently locate and maintain multiple global optima of a multimodal function. Here, we incorporate two index-based neighborhoods; a two side ring and a Von Neumann neighborhood [27], [28]. The new mutation strategies are evaluated on eight well-known and widely used multimodal problems and their performance is compared against five state-of-the-art algorithms. Simulation results suggest that the proposed strategies are promising and exhibit competitive behavior.

The rest of the paper is structured as follows: Section II briefly describes the motivation behind the development of the proposed mutation strategies. A brief description of the new mutation strategies are presented in Section III. Next, the outcomes of an extensive experimental analysis are presented in Section IV. Finally the paper concludes in Section V with a discussion and some pointers for future work.

II. Motivation

In a recent work [6], we have exploited Differential Evolution's tendency to distribute the individuals of its population in the vicinity of the considered problem's minima [7], [17], and introduced a family of mutation strategies that produce a niching effect. This family of mutation strategies take advantage of the nearest neighbor interactions between the individuals of a population and by a simple modification can accurately locate and maintain many global minima, through the evolution stages of the algorithm.

Although, to produce stable niches through the evolution process, the aforementioned family incorporates into the mutation strategies the nearest neighbor of each individual. This information can be characterized as a global information, since it incorporates information calculated from the entire population and demands a high computational cost. One way to calculate the nearest neighbor interactions is to employ properly computational geometry methods to efficiently reduce the computational complexity of this procedure [29]. Nevertheless, its incorporation in an algorithm is not a straight forward procedure, since it demands great knowledge of computational geometry concepts and methodologies, as well as great programming skills.

Recently, it has been demonstrated that the simple Particle Swarm Optimization (PSO) algorithm with an index-based ring topology is able to induce stable niching behaviors [25]. The produced niching effect is based on PSO's ability to posses a local memory into its population (swarm). The main role of the index-based ring topology was to provide a mechanism to slow down information propagation in the population, as well as to allow different neighboring individuals to coexist through time. Furthermore, index-based neighborhoods are easily implemented without employing an additional overhead into the considered algorithm.

Motivated by these findings, we also incorporate an index-based neighborhood. To this end, we reduce the computational cost of the mutation strategies and still are able to efficiently locate and maintain multiple global optima of a multimodal function.

III. Differential Evolution mutation strategies with index-based neighborhoods

The DE algorithm [14] is a stochastic parallel direct search method, which implements concepts borrowed from the broad class of Evolutionary Algorithms (EAs). In detail, DE is a population–based stochastic algorithm that takes advantage of a population of NP potential solutions, *individuals*, to effectively probe the search space. In the first stage of evolution, DE randomly initializes the population in the D–dimensional optimization domain through a uniform probability distribution. Individuals evolve over consecutive iterations to explore the search space and locate the optima of the objective function. At each iteration, called *generation*, new vectors are created by the combination of a base vector and some randomly chosen vectors from the current population. This operation in our context can be referred to as *mutation*, while the outcoming vectors as *mutant individuals*. The selection of the base vector depends on the mutation operator. In turn, each mutant individual is then mixed with another vector – the *target* vector – through an operation called *recombination* or *crossover*. This operation yields the so–called *trial* vector. Finally, the trial vector undergoes the *selection* operation, according to which it is accepted as a member of the population of the next generation only if it yields a reduction in the value of the objective function f relative to that of the target vector. Otherwise, target vector is retained in the next generation. DE operators efficiently shuffle information among the individuals, enabling the search procedure to focus on the most promising regions of the optimization space. A more comprehensive description of the DE can be found in [14]–[17].

The recently proposed DE/nrand family is capable of handling multimodal landscapes by taking advantage from the dynamics of the classic DE's mutation strategies and the local information of the current population [6]. To evolve each individual it employs into the mutation schemes as a base vector its real nearest neighbor individual. Thus, a niching effect is produced by maintaining the individuals to the vicinity of the problem's optima, while they simultaneously explore the search space. The computational complexity of the DE/nrand family is determined by the computational burden of the nearest neighbor computations. In the worst case, the DE/nrand strategies exhibit quadratic complexity in the population size, $O(NP^2)$, at each generation, since they have to calculate the nearest neighbors of the whole population. A naive implementation of this procedure may cost the calculation of the population's affinity matrix, i.e. NP^2 Euclidean distances.

In this work, we propose not to compute the real nearest neighbors; instead we propose to find the real nearest neighbor of each individual in an index–based neighborhood. To this end, we employ an index–based neighborhood into the population and act as the neighborhood of each individual is the index neighborhood based on a predefined topology, e.g. a ring, a star, a Von Neumann, etc. As such, for each individual x_g^i, $i = 1, 2, \ldots, NP$, where g denotes the current generation, the mutant individual v_{g+1}^i can be generated according to one of the following proposed equations:

1) "**DE/inrand/1**"

$$v_{g+1}^i = x_g^{INN_i} + F(x_g^{r_1} - x_g^{r_2}), \qquad (1)$$

2) "**DE/inrand/2**"

$$v_{g+1}^i = x_g^{INN_i} + F(x_g^{r_1} - x_g^{r_2}) + F(x_g^{r_3} - x_g^{r_4}), \quad (2)$$

where $F > 0$ is a real parameter, called *mutation or scaling factor*, $x_g^{INN_i}$ is the nearest neighbor of the index–based neighborhood of the current individual x_g^i, and $r_1, r_2, r_3, r_4 \in \{1, 2, \ldots, NP\} \setminus \{i\}$ are random integers mutually different and not equal to the running index i. The remaining DE steps are the same as the classic DE/rand/1/bin scheme [14], i.e. we employ the binomial crossover operator and the simple elitist selection operator.

Here, we incorporate two index-based neighborhoods, a ring (R) and a Von Neumann (V) neighborhood [27], [28]. Thus, the mutation strategies are named DE/inrand/1R, DE/inrand/2R for the ring neighborhood and DE/inrand/1V, DE/inrand/2V for the Von Neumann. To this end, its complexity decreases radically to the calculation of $n_n \ll NP$ distances for each individual, where n_n is the number of the neighbors in the index-based neighborhood. Obviously, as the number of index neighbors increase, the complexity of the corresponding mutation strategy increases. We expect that the newly developed mutation strategies will have more exploratory characteristics, due to the randomness of the index–based neighborhood, while still posses the ability to locate and maintain the discovered global optima throughout the evolutionary process, due to the real nearest neighbor interactions in the index–based neighborhood.

IV. Experimental Results

In this section, we evaluate the proposed approaches by comparing them with several Differential Evolution variants that handle multimodal optimization problems. To verify the effectiveness of the proposed approaches, we employ eight classic multimodal benchmark functions with different characteristics, such as multiple evenly and unevenly spaced global optima, multiple global optima in the presence of multiple local optima and deceptiveness [22]. A thorough description of the benchmark functions along with their main characteristics can be found in [6], [22].

To demonstrate the efficiency of the proposed approaches, we compare them against five methods that can handle multimodal landscapes, i.e. the two recently proposed DE/nrand/1 and DE/nrand/2 algorithms and three state-of-the-art Differential Evolution variants, namely the Crowding DE [18], the DELS [21] and the Species-based DE (Species DE) [19], [20]. Throughout this section, all the reported results are averaged over 100 independent simulations. For each simulation and each algorithm we have initialized the populations using a uniform random number distribution with the same random seeds. Moreover, all methods have been implemented with the default parameters settings, as have been proposed in the literature. Regarding the DE control parameters, the common settings of $F = 0.5$ and $CR = 0.9$ were used for all DE variants [15]–[17]. For each simulation, we have incorporated fixed size populations with $NP = 100$ individuals and a budget of $\max NFEs = 10^5$ function evaluations.

To verify the effectiveness and the ability of the proposed approaches to accurately locate the global minima, we first specify a *level of accuracy*, $\varepsilon \in (0, 1]$. The *level of accuracy* parameter depicts the tolerance level of a computed solution to be considered as a global optimum. In detail, if the Euclidean distance of a computed solution to a known global optimum is less than the pre-specified *level of accuracy* ε, then the solution is considered to be a global optimum. Moreover, since in the current benchmark functions the number as well as the location of optima to be found is known *a priori*, we can use it as a performance metric. Therefore, based on the aforementioned metric, to compare the performance of the implemented algorithms, we adopt the *peak ratio* and the *success rate* measures [18]. In detail, for a pre-specified budget of function evaluations ($\max NFEs$) and an *accuracy level* ε, the *peak ratio* (PR) measures the percentage of global optima (i.e. peaks) located over the total number of known global optima. Therefore, for a single simulation the *peak ratio* can be defined as: PR $= (\#\text{PF})/(\#\text{PT})$, where #PF is the number of founded peaks while #PT determines the number of total peaks. Notice that the reported PR values are average values over 100 independent simulations. In turn, the *success rate* (SR) measures the percentage of simulations in which *all* global optima have been successfully located.

In the experiment, we try to find out if the considered algorithms can accurately locate the global minima in different levels of accuracy. Hence, we consider seven different accuracy levels, namely $\varepsilon \in \{10^{-3}, 10^{-4}, \ldots, 10^{-9}\}$. This task turns to be very challenging, since when the accuracy level increases the accuracy of the computed global minima increases. It is expected that, as the accuracy levels increase, both the SR and the PR measures vary with respect to different levels of accuracy, while the algorithms tend to produce smaller values of both peak ratio and success rates as the accuracy levels increase. Table I demonstrates extensive experimental results of all algorithms over the considered benchmark suite.

In detail, Species-based DE exhibits poor performance in three of the four most challenging functions on this benchmark set (e.g. functions f_3, f_6, and f_7) with the considered parameter settings. In the remaining functions although it can successfully locate the desired number of minima in the first four accuracy levels, in the remaining levels (1.0e-06 – 1.0e-09) performs poorly. Species DE performs well only in the f_8 function. We speculate that Species DE needs a higher population size to accurately locate all global minima on the given budget of function evaluations (see also Section IV-C, where we study the effect of the population size on the algorithms). Crowding DE is one of the most promising

TABLE I
SUCCESS RATIO AND PEAK RATIO MEASURES FOR THE MULTIMODAL FUNCTIONS f_1–f_8

Function f_1	Crowding DE		DELS		Species DE		DE/nrand/1		DE/nrand/2		DE/inrand/1R		DE/inrand/2R		DE/inrand/1V		DE/inrand/2V	
Acc. level ε	SR	PR	SR	PR	SR	PR	SR	PR	SR	PR	SR	PR	SR	PR	SR	PR	SR	PR
1.0e-03	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	0.950	0.983	0.940	0.980	0.910	0.970	0.840	0.947
1.0e-04	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	0.990	0.997	0.900	0.967	0.920	0.973	0.930	0.977	0.850	0.950
1.0e-05	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	0.950	0.983	0.920	0.973	0.870	0.957	0.880	0.960
1.0e-06	0.910	0.970	1.000	1.000	0.860	0.953	1.000	1.000	1.000	1.000	0.940	0.980	0.970	0.990	0.910	0.970	0.840	0.947
1.0e-07	0.110	0.400	1.000	1.000	0.210	0.587	1.000	1.000	0.990	0.997	0.940	0.980	0.910	0.970	0.910	0.970	0.890	0.963
1.0e-08	0.000	0.077	1.000	1.000	0.010	0.193	1.000	1.000	0.990	0.997	0.940	0.980	0.910	0.970	0.880	0.960	0.840	0.947
1.0e-09	0.000	0.003	1.000	1.000	0.000	0.040	1.000	1.000	1.000	1.000	0.960	0.987	0.910	0.970	0.840	0.947	0.800	0.933
Function f_2	Crowding DE		DELS		Species DE		DE/nrand/1		DE/nrand/2		DE/inrand/1R		DE/inrand/2R		DE/inrand/1V		DE/inrand/2V	
Acc. level ε	SR	PR	SR	PR	SR	PR	SR	PR	SR	PR	SR	PR	SR	PR	SR	PR	SR	PR
1.0e-03	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	0.980	0.995
1.0e-04	1.000	1.000	1.000	1.000	0.960	0.990	1.000	1.000	1.000	1.000	1.000	1.000	0.980	0.995	0.990	0.998	0.980	0.995
1.0e-05	1.000	1.000	1.000	1.000	0.410	0.835	1.000	1.000	1.000	1.000	1.000	1.000	0.990	0.998	1.000	1.000	0.950	0.988
1.0e-06	1.000	1.000	1.000	1.000	0.030	0.482	1.000	1.000	1.000	1.000	1.000	1.000	0.960	0.990	0.990	0.998	0.950	0.988
1.0e-07	0.920	0.980	1.000	1.000	0.000	0.132	1.000	1.000	1.000	1.000	1.000	1.000	0.930	0.983	1.000	1.000	0.890	0.973
1.0e-08	0.170	0.558	1.000	1.000	0.000	0.050	1.000	1.000	1.000	1.000	1.000	1.000	0.930	0.980	0.990	0.998	0.900	0.975
1.0e-09	0.000	0.050	1.000	1.000	0.000	0.005	1.000	1.000	1.000	1.000	0.990	0.998	0.910	0.975	0.960	0.990	0.800	0.945
Function f_3	Crowding DE		DELS		Species DE		DE/nrand/1		DE/nrand/2		DE/inrand/1R		DE/inrand/2R		DE/inrand/1V		DE/inrand/2V	
Acc. level ε	SR	PR	SR	PR	SR	PR	SR	PR	SR	PR	SR	PR	SR	PR	SR	PR	SR	PR
1.0e-03	0.000	0.066	0.050	0.779	0.000	0.398	0.000	0.723	0.940	0.996	0.280	0.877	0.750	0.984	0.030	0.636	0.690	0.980
1.0e-04	0.000	0.006	0.000	0.186	0.000	0.104	0.010	0.741	0.880	0.993	0.340	0.896	0.690	0.976	0.000	0.607	0.680	0.980
1.0e-05	0.000	0.000	0.000	0.017	0.000	0.018	0.000	0.726	0.940	0.997	0.280	0.865	0.720	0.983	0.020	0.622	0.640	0.971
1.0e-06	0.000	0.001	0.000	0.004	0.000	0.004	0.000	0.726	0.980	0.999	0.130	0.842	0.740	0.984	0.010	0.601	0.590	0.963
1.0e-07	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.726	0.970	0.998	0.240	0.870	0.730	0.983	0.010	0.617	0.690	0.973
1.0e-08	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.724	1.000	1.000	0.310	0.872	0.520	0.958	0.010	0.592	0.720	0.981
1.0e-09	0.000	0.000	0.000	0.000	0.000	0.000	0.010	0.724	0.910	0.995	0.270	0.873	0.260	0.841	0.000	0.614	0.610	0.962
Function f_4	Crowding DE		DELS		Species DE		DE/nrand/1		DE/nrand/2		DE/inrand/1R		DE/inrand/2R		DE/inrand/1V		DE/inrand/2V	
Acc. level ε	SR	PR	SR	PR	SR	PR	SR	PR	SR	PR	SR	PR	SR	PR	SR	PR	SR	PR
1.0e-03	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
1.0e-04	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
1.0e-05	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
1.0e-06	1.000	1.000	1.000	1.000	0.980	0.990	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
1.0e-07	1.000	1.000	1.000	1.000	0.620	0.790	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
1.0e-08	1.000	1.000	1.000	1.000	0.140	0.385	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
1.0e-09	1.000	1.000	1.000	1.000	0.000	0.125	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
Function f_5	Crowding DE		DELS		Species DE		DE/nrand/1		DE/nrand/2		DE/inrand/1R		DE/inrand/2R		DE/inrand/1V		DE/inrand/2V	
Acc. level ε	SR	PR	SR	PR	SR	PR	SR	PR	SR	PR	SR	PR	SR	PR	SR	PR	SR	PR
1.0e-03	0.000	0.723	0.000	0.513	0.000	0.372	0.000	0.398	0.000	0.393	0.000	0.362	0.000	0.303	0.000	0.337	0.000	0.273
1.0e-04	0.000	0.715	0.000	0.495	0.000	0.364	0.000	0.393	0.000	0.387	0.000	0.351	0.000	0.311	0.000	0.338	0.000	0.286
1.0e-05	0.000	0.609	0.000	0.475	0.000	0.340	0.000	0.387	0.000	0.383	0.000	0.349	0.000	0.300	0.000	0.328	0.000	0.279
1.0e-06	0.000	0.352	0.000	0.445	0.000	0.193	0.000	0.378	0.000	0.351	0.000	0.340	0.000	0.289	0.000	0.336	0.000	0.276
1.0e-07	0.000	0.126	0.000	0.406	0.000	0.057	0.000	0.366	0.000	0.291	0.000	0.335	0.000	0.273	0.000	0.317	0.000	0.258
1.0e-08	0.000	0.021	0.000	0.363	0.000	0.012	0.000	0.352	0.000	0.201	0.000	0.324	0.000	0.248	0.000	0.306	0.000	0.241
1.0e-09	0.000	0.004	0.000	0.282	0.000	0.001	0.000	0.340	0.000	0.095	0.000	0.322	0.000	0.205	0.000	0.298	0.000	0.211
Function f_6	Crowding DE		DELS		Species DE		DE/nrand/1		DE/nrand/2		DE/inrand/1R		DE/inrand/2R		DE/inrand/1V		DE/inrand/2V	
Acc. level ε	SR	PR	SR	PR	SR	PR	SR	PR	SR	PR	SR	PR	SR	PR	SR	PR	SR	PR
1.0e-03	1.000	1.000	0.620	0.982	0.000	0.527	0.640	0.982	0.670	0.984	0.280	0.950	0.430	0.968	0.220	0.939	0.300	0.956
1.0e-04	1.000	1.000	0.620	0.983	0.000	0.336	0.640	0.983	0.610	0.982	0.290	0.952	0.330	0.963	0.300	0.948	0.360	0.963
1.0e-05	1.000	1.000	0.490	0.973	0.000	0.093	0.630	0.983	0.570	0.980	0.220	0.949	0.440	0.968	0.260	0.942	0.460	0.969
1.0e-06	1.000	1.000	0.010	0.762	0.000	0.016	0.660	0.984	0.630	0.982	0.320	0.951	0.360	0.957	0.290	0.948	0.480	0.971
1.0e-07	1.000	1.000	0.000	0.192	0.000	0.001	0.690	0.984	0.570	0.980	0.300	0.957	0.360	0.962	0.270	0.947	0.360	0.965
1.0e-08	1.000	1.000	0.000	0.025	0.000	0.000	0.670	0.986	0.610	0.981	0.260	0.946	0.440	0.966	0.250	0.941	0.500	0.971
1.0e-09	1.000	1.000	0.000	0.002	0.000	0.000	0.540	0.978	0.610	0.978	0.290	0.955	0.420	0.964	0.290	0.943	0.410	0.963
Function f_7	Crowding DE		DELS		Species DE		DE/nrand/1		DE/nrand/2		DE/inrand/1R		DE/inrand/2R		DE/inrand/1V		DE/inrand/2V	
Acc. level ε	SR	PR	SR	PR	SR	PR	SR	PR	SR	PR	SR	PR	SR	PR	SR	PR	SR	PR
1.0e-03	1.000	1.000	0.270	0.951	0.000	0.528	0.000	0.825	0.020	0.869	0.000	0.738	0.000	0.760	0.000	0.705	0.000	0.742
1.0e-04	1.000	1.000	0.160	0.944	0.000	0.364	0.000	0.804	0.010	0.837	0.000	0.728	0.000	0.733	0.000	0.681	0.000	0.723
1.0e-05	0.010	0.865	0.010	0.880	0.000	0.105	0.000	0.783	0.000	0.573	0.000	0.708	0.000	0.627	0.000	0.670	0.000	0.630
1.0e-06	0.000	0.244	0.000	0.510	0.000	0.019	0.000	0.748	0.000	0.203	0.000	0.712	0.000	0.324	0.000	0.650	0.000	0.382
1.0e-07	0.000	0.029	0.000	0.109	0.000	0.002	0.000	0.697	0.000	0.023	0.000	0.676	0.000	0.106	0.000	0.644	0.000	0.127
1.0e-08	0.000	0.003	0.000	0.013	0.000	0.000	0.000	0.648	0.000	0.006	0.000	0.647	0.000	0.018	0.000	0.612	0.000	0.026
1.0e-09	0.000	0.000	0.000	0.002	0.000	0.000	0.000	0.570	0.000	0.000	0.000	0.600	0.000	0.002	0.000	0.568	0.000	0.005
Function f_8	Crowding DE		DELS		Species DE		DE/nrand/1		DE/nrand/2		DE/inrand/1R		DE/inrand/2R		DE/inrand/1V		DE/inrand/2V	
Acc. level ε	SR	PR	SR	PR	SR	PR	SR	PR	SR	PR	SR	PR	SR	PR	SR	PR	SR	PR
1.0e-03	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	0.990	0.998	1.000	1.000	0.990	0.998	0.990	0.995
1.0e-04	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
1.0e-05	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	0.980	0.995
1.0e-06	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	0.990	0.998	0.990	0.995
1.0e-07	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	0.990	0.998	1.000	1.000
1.0e-08	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	0.990	0.998	1.000	1.000	1.000	1.000	1.000	1.000
1.0e-09	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	0.990	0.998	1.000	1.000

approaches tested in this study, since it exhibits a very good performance (e.g. functions f_4, f_6, and f_8). However, when the accuracy level increases it can not maintain the population on many global minima (e.g. functions f_1–f_3, f_5, and f_7). Hence, as depicted in Table I, for the aforementioned functions, both SR and PR measures decrease as the accuracy level increase.

Similarly, the DELS method, although it performs very well in functions f_1, f_2, f_4 and f_8, in the most challenging functions with many global minima (e.g. functions f_3, f_5, f_6, and f_7) its performance slowly degrades as the accuracy level increases.

As expected DE/nrand/1 and DE/nrand/2 algorithms, exhibit very good performance, in terms of SR and PR measures, for the majority of the considered functions (e.g. functions $f_1 - f_4, f_6, f_7$, and f_8). Both strategies, exhibit either equal or better performance when compared against the aforementioned methods, especially when the accuracy level increases (e.g. functions $f_1 - f_4$, and f_8). In detail, in most challenging problems (e.g. functions f_3, f_5, f_6, and f_7), DE/nrand/1 locates either all or a high number of global minima, irrespectively of the accuracy level. In the functions where Crowding DE and DELS perform poorly, DE/nrand/1 performs better in terms of PR and SR, e.g. in f_3, and the high accuracy levels of f_5 and f_7. Similarly, in most of the functions DE/nrand/2 performs equally well, while in the challenging f_3 function exhibits the best performance in comparison with all the other methods.

Comparing DE/nrand against the proposed algorithms, we can generally observe that the proposed approaches exhibit either a comparative or an equally well performance in most of the tested functions. Only in a few cases, the proposed approaches exhibits a superior performance against the DE/nrand algorithms. With much less computational cost, algorithms with a similar or slightly better performance can be developed. In detail, the index-based neighborhood approaches follow the performance of their corresponding DE/nrand algorithm, i.e. the DE/inrand/1R and DE/inrand/1V follow the performance of DE/nrand/1, and similarly DE/inrand/2R and DE/inrand/2V follow the performance of DE/nrand/2.

All proposed approaches with either the ring or the Von Neumann index-based neighborhood exhibit very good performance in the f_2, f_4 and f_8 functions, with the latter neighborhood to perform slightly worse for some accuracy levels, e.g. in the f_2 and f_8 functions. Regarding the f_1 function, the proposed algorithms exhibit lower PR and SR values than the DE/nrand/1 and DE/nrand/2 algorithms. In the most challenging problems, the proposed approaches follow the performance of the DE/nrand family with either equal or slightly worse performance in terms of SR and PR values. The proposed approaches exhibit either a superior or an equally well performance in the f_3 function. In this case the DE/inrand/1R exhibits higher SR and PR values against the DE/nrand/1 algorithm, irrespectively of the accuracy level. In f_5 and f_7, all proposed algorithms perform similarly with the corresponding DE/nrand algorithms, while in the f_6 function the index-based neighborhood approaches exhibit lower PR and SR values, with the same trends as the corresponding DE/nrand algorithms. Between the index-based neighborhoods, the ring topology seems to exhibit better performance than the Von Neumann topology. However, a further investigation on the index-based topologies should be made to draw any safe conclusions.

Based on the aforementioned observations, we believe that the proposed index-based neighborhood approaches exhibit a great potential to accurately locate many global minima with a low computational cost, and thus successfully tackle multimodal functions.

A. Convergence speed

In this section, we exhibit experimental results for all considered algorithms in terms of convergence speed. To measure the convergence speed of an algorithm at a prespecified level of accuracy, we obtain the required number of function evaluations in which it can accurately locate all global minima of the problem at hand. Specifically, for the accuracy level of $\varepsilon = 10^{-4}$, for each problem and each algorithm, we have conducted 100 independent simulations and the aforementioned parameter setup. Table II demonstrates the mean number of function evaluations (*Mean*), its standard deviation (*St.D.*), the SR measure, and the mean value of the PR measure over the f_1, f_2, f_4, and f_8 functions. Furthermore, to evaluate the statistical significance of the observed function evaluations differences, for each algorithm we conduct four two-sided Wilcoxon rank sum test between the corresponding algorithm and the proposed index-based approaches in the following order: DE/inrand/1R, DE/inrand/2R, DE/inrand/1V, and DE/inrand/2V. The null hypothesis in each test is that the samples compared are independent samples from identical continuous distributions with equal medians. Thereby, we mark with "+" the cases when the null hypothesis is rejected at the 5% significance level and the corresponding proposed approach exhibits superior performance. The "–" mark indicates that the null hypothesis is rejected at the same level of significance and the proposed approach exhibits inferior performance, while the "=" mark indicates that the performance difference is not statistically significant.

It can be clearly observed that in the majority of the tested cases the proposed index-based neighborhood algorithms, with either ring or Von Neumann neighborhoods, produce competitive or better performance in terms of lower mean values of function evaluations. The most promising approach seems to be the DE/inrand/1V algorithm, since it exhibits either the best or an equally well performance in comparison with the other algorithms, while in the second place comes the DE/inrand/1R algorithm. Comparing the proposed algorithms with the Crowding DE, the DELS and the Species DE algorithms, it can be clearly observed that all the proposed approaches exhibit statistical significant performance gains for all considered test cases. Furthermore, the index-based neighborhood algorithms perform better or equally well to the DE/nrand/1 and DE/nrand/2 algorithms. Both DE/inrand/1R and DE/inrand/1V enhance the convergence speed of the DE/nrand/1 algorithm, while a similar behavior is observed for the DE/inrand/2R and DE/inrand/2V algorithms against the DE/nrand/2. Between the index-based neighborhood algorithms the DE/inrand/1R algorithm produce the best performance in terms of Success Rate and a quite well performance in terms of mean values of function evaluations.

Furthermore, Figure 1, illustrates the behavior of the considered methods during the simulations. Specifically, we perform

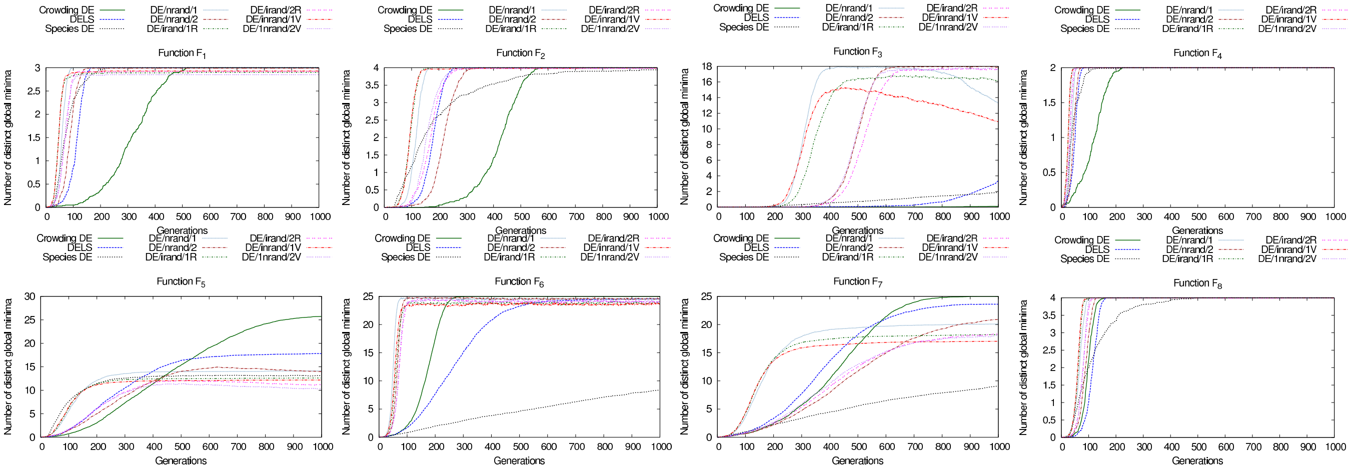

Fig. 1. Mean number of global minima found during 100 independent simulations of all methods over f_1–f_8 ($\varepsilon = 10^{-4}$)

TABLE II
CONVERGENCE SPEED (ACCURACY LEVEL $\varepsilon = 10^{-4}$)

Function f_1	*Mean*	*St.D.*	SR	Mean PR	St. Sig.
Crowding DE	39503.000	6215.320	1.000	1.000	(+/+/+/+)
DELS	13264.000	1544.790	1.000	1.000	(+/+/+/+)
Species DE	12098.500	4111.740	1.000	1.000	(+/+/+/+)
DE/nrand/1	7210.000	1057.780	1.000	1.000	(+/–/+/–)
DE/nrand/2	10991.900	1694.270	0.990	0.997	(+/+/+/+)
DE/inrand/1R	7137.780	5447.700	0.900	0.967	(=/–/+/–)
DE/inrand/2R	8910.870	1147.710	0.920	0.973	(+/=/+/+)
DE/inrand/1V	**6496.770**	2586.140	0.930	0.977	(–/–/=/–)
DE/inrand/2V	9090.590	1852.580	0.850	0.950	(+/–/+/=)
Function f_2	***Mean***	***St.D.***	**SR**	**Mean PR**	**St. Sig.**
Crowding DE	49245.000	4033.680	1.000	1.000	(+/+/+/+)
DELS	21499.000	2040.330	1.000	1.000	(+/+/+/+)
Species DE	37107.200	19760.100	0.960	0.990	(+/+/+/+)
DE/nrand/1	13806.000	1437.790	1.000	1.000	(+/–/+/–)
DE/nrand/2	26079.000	3029.230	1.000	1.000	(+/–/+/+)
DE/inrand/1R	**11730.000**	1798.510	1.000	1.000	(=/–/+/–)
DE/inrand/2R	20539.800	3179.180	0.980	0.995	(+/=/+/+)
DE/inrand/1V	12489.900	6836.550	0.990	0.998	(–/–/=/–)
DE/inrand/2V	19965.300	3101.480	0.980	0.995	(+/–/+/=)
Function f_4	***Mean***	***St.D.***	**SR**	**Mean PR**	**St. Sig.**
Crowding DE	14390.000	3602.620	1.000	1.000	(+/+/+/+)
DELS	5779.000	1063.380	1.000	1.000	(+/+/+/+)
Species DE	6219.280	2854.200	1.000	1.000	(+/+/+/+)
DE/nrand/1	4259.000	687.713	1.000	1.000	(+/–/+/–)
DE/nrand/2	5198.000	906.428	1.000	1.000	(+/–/+/+)
DE/inrand/1R	3010.000	713.294	1.000	1.000	(=/–/+/–)
DE/inrand/2R	3776.000	801.655	1.000	1.000	(+/=/+/+)
DE/inrand/1V	**2995.000**	674.406	1.000	1.000	(–/–/=/–)
DE/inrand/2V	3558.000	624.933	1.000	1.000	(+/–/+/=)
Function f_8	***Mean***	***St.D.***	**SR**	**Mean PR**	**St. Sig.**
Crowding DE	11729.000	1554.090	1.000	1.000	(+/+/+/+)
DELS	13737.000	1257.520	1.000	1.000	(+/+/+/+)
Species DE	22453.900	9632.750	1.000	1.000	(+/+/+/+)
DE/nrand/1	8211.000	1058.200	1.000	1.000	(+/–/+/–)
DE/nrand/2	10777.000	1178.500	1.000	1.000	(+/+/+/+)
DE/inrand/1R	7535.000	826.013	1.000	1.000	(=/–/+/–)
DE/inrand/2R	9588.000	967.062	1.000	1.000	(+/=/+/+)
DE/inrand/1V	**7323.000**	782.234	1.000	1.000	(–/–/=/–)
DE/inrand/2V	9300.000	979.487	1.000	1.000	(+/–/+/=)

100 independent simulations for the accuracy level $\varepsilon = 10^{-4}$ and record the number of global minima the method locates throughout the simulation. As expected, DE/nrand/1 and DE/nrand/2 algorithms efficiently and accurately locate a high number of global minima and maintain them until the end of the simulation for the majority of the tested cases. Similarly, the proposed approaches follow the same behavior in most of the cases, while in some exhibit better performance in terms of convergence speed (e.g. in functions f_2, f_4 and f_8). Nevertheless, there are a few cases where their performance deteriorates. Although they exhibit a good niching effect and succeed to locate several minima, they can not maintain them until the end of the simulation, such as DE/inrand/1V in the f_3 function and DE/inrand/2R, DE/inrand/2V in the f_5 function. Moreover, DELS and Crowding DE exhibit similar behavior, but they tend to locate the global minima slowly. On the other hand, Species DE performs poorly in three of the most challenging functions (f_3, f_6 and f_7), while in functions f_2 and f_8 locates the required number of global optima quite slowly. It can accurately locate and maintain all global optima with a good convergence speed only in the f_1 and f_4 functions. The aforementioned behavior of the considered algorithms tends to be more visible as the accuracy level increases and can be captured by the PR measure demonstrated in Table I.

B. Population size influence

In this section, we study the influence of the population size on the performance of the considered algorithms. Thus, for several population sizes, we measure their performance by calculating the number of global minima found at the accuracy level $\varepsilon = 10^{-4}$. The test suite comprises of the functions having many global minima (e.g. functions $f_3, f_5, f_6,$ and f_7). Figure 2 illustrates the performance of the considered algorithms, as the population size increases from 40 to 300 individuals.

As expected, all algorithms exhibit a problem dependent behavior. In general, there exist a range of population size values where most of the algorithms can locate a high number of global minima. More specifically, in function f_6 as the population size increase, almost all algorithms can locate more global minima. In functions f_5 and f_7, as the population

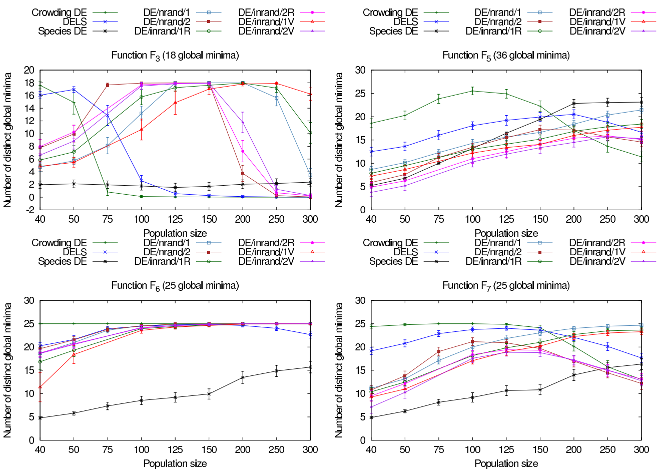

Fig. 2. Population size effect on the f_3, f_5, f_6 and f_7 functions

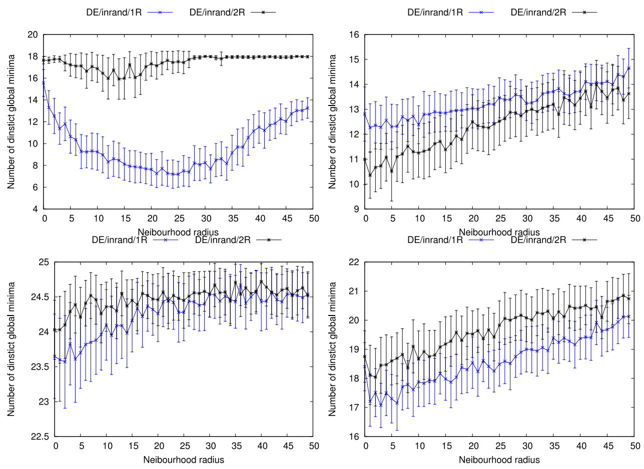

Fig. 3. Neighbourhood radius effect on the f_3, f_5, f_6 and f_7 functions

increases DE/nrand/1, DE/inrand/1R, DE/inrand/1V, manage to locate a higher number of global minima. The remaining algorithms can locate and maintain a high number of global optima with population sizes ranging from 75 to 150, while after that as the population size increases their performance deteriorates. Function f_3 is a special case for all algorithms, in which DE/nrand/2, DE/inrand/2R and DE/inrand/2V exhibit their best performance with a population size between 100 and 150 individuals, and afterwards their performance rapidly decrease. In turn, the performance of DE/nrand/1, DE/inrand/1R, and DE/inrand/1V generally increases as the population size increases. On the contrary, both Crowding DE and DELS perform poorly in the f_3 function, irrespectively of the population size.

In the case of the Species DE, the observed performance validates our assumption that it needs a higher population size to accurately locate all global minima on the given budget of function evaluations. Except for the f_3 function, where it performs poorly irrespectively of the population size, Species DE performance increases as the population size increases. Its best performance is demonstrated in f_5 function for the larger population sizes values, e.g. for population sizes ranging from 200 to 300.

C. Investigating the ring index-based neighborhood topology

The aforementioned observations suggest that a further investigation should be made on the index-based neighborhood approaches. Here, we conduct a brief analysis of the ring topology and especially how the ring radius influences the performance of the proposed approach. Specifically, for the accuracy level of $\varepsilon = 10^{-4}$, we measure the performance of the DE/inrand/1R and DE/inrand/2R by calculating the number of global minima found on the most challenging functions considered in this work (e.g. functions f_3, f_5, f_6, and f_7). For each problem and each algorithm we have conducted 100 independent simulations for neighborhood radius $n_r \in \{1, 2, \ldots, NP/2\}$. Noticed that the $n_r = NP/2$ case is the same as applying the DE/nrand/1 and DE/nrand/2 algorithms. To this end, Figure 3 illustrates mean values of the performance of the DE/inrand/1R and DE/inrand/2R algorithms calculated over the different neighborhood radius values. Error bars around the mean depict the standard deviation of the measured performance.

It can be observed that the performance of the applied algorithms is problem dependent. In the f_3 function, the DE/inrand/1R exhibits very good performance for the majority of the neighborhood radius values, while it performs best for either very small or very large radius values. On the contrary, DE/inrand/2R performs best only when $n_f = 1$, while for the remaining radius values its performance deteriorates. In the $f_5 - f_7$ functions the performance of both DE/inrand/1R and DE/inrand/2R increases as the neighborhood radius increase. This trend is strongly highlighted in the f_5 and f_7 functions, where the best performance for both algorithms is when the neighborhood radius reaches the $NP/2$ value (DE/nrand/1 and DE/nrand/2 algorithms). Finally, in the f_5 function, all radius values exhibit great performance, while both considered algorithms exhibit their best performance for radius values greater than 30. Nevertheless, in most cases the performance difference between the different neighborhood radius values is small. Thus, the selection of the smallest neighborhood radius $n_f = 1$ is highly recommended to minimize the computational cost of the algorithm.

In the majority of the algorithms that handle multimodal functions, the computational cost depends on the utilized niching methodology. In this study, we produce acceptable performance with the index-based neighborhood, with the potential to improve its performance with a computational cost overhead, i.e. if we utilize a high neighborhood radius value that reach the DE/nrand/1,2 algorithms. To this end, in a future work we intend to study the utilization of varying neighborhood topologies that may produce either a similar or a higher performance with the ability to maintain a minimal computational cost overhead.

V. Conclusions

In this work, we modify a recently proposed DE family of mutation strategies, which can handle multimodal functions, namely the DE/nrand family [6]. To induce the classical DE mutation strategies to exhibit a niching effect, DE/nrand family incorporates information regarding the real nearest neighborhood of each potential solution. Nevertheless, their calculation is computationally expensive. Thus, to alleviate this problem, instead of computing the real nearest neighbor computations, we incorporate an index-based neighborhood into the mutation strategies in an attempt to radically reduce their computational cost and simultaneously locate and maintain multiple global optima.

Experimental results on eight well known multimodal functions with different characteristics and comparisons against five state-of-the-art DE algorithms, demonstrate that the proposed mutation strategies are competitive and promising. Specifically, they can locate many global optima and maintain them through the evolution process, with a substantial decrease of their computational cost. In comparison with their ancestor mutation strategies, their performance is either equally well or slightly worse. In terms of convergence speed, they can accurately locate all global minima in less function evaluations than the other algorithms. Finally, experiments regarding the effect of the population size show that, in most of the cases, the behavior of the proposed algorithms is robust.

In a future work, an extensive study on the impact of the index–based neighborhoods on more challenging multimodal function families, with higher dimensions and different characteristics is scheduled. In addition, it would be very interesting to study other methodologies that could help reduce the computational cost of the DE/nrand family and simultaneously maintain their niching capabilities.

Acknowledge

The authors would like to thank the European Union (European Social Fund ESF) and Greek national funds through the Operational Program "Education and Lifelong Learning" of the National Strategic Reference Framework (NSRF) - Research Funding Program: "Heracleitus II. Investing in knowledge society through the European Social Fund." for financially supporting this work.

References

[1] T. Bäck, D. B. Fogel, and Z. Michalewicz, Eds., *Handbook of Evolutionary Computation*. Oxford University Press, 1997.

[2] K. A. De Jong, "An analysis of the behavior of a class of genetic adaptive systems." Ph.D. dissertation, University of Michigan, USA, 1975.

[3] S. W. Mahfoud, "Niching methods for genetic algorithms," Ph.D. dissertation, University of Illinois at Urbana-Champaign, Champaign, IL, USA, 1995, uMI Order No. GAX95-43663.

[4] D. E. Goldberg and J. Richardson, "Genetic algorithms with sharing for multimodal function optimization," in *Proceedings of the Second International Conference on Genetic Algorithms on Genetic algorithms and their application*, Hillsdale, NJ, USA, 1987, pp. 41–49.

[5] A. Pétrowski, "A clearing procedure as a niching method for genetic algorithms," in *IEEE International Conference on Evolutionary Computation*, 1996, pp. 798–803.

[6] M. G. Epitropakis, V. P. Plagianakos, and M. N. Vrahatis, "Finding multiple global optima exploiting differential evolution's niching capability," in *2011 IEEE Symposium on Differential Evolution (SDE)*, April 2011, pp. 1–8.

[7] D. K. Tasoulis, V. P. Plagianakos, and M. N. Vrahatis, "Clustering in evolutionary algorithms to efficiently compute simultaneously local and global minima," in *IEEE Congress on Evolutionary Computation*, vol. 2, 2005, pp. 1847–1854.

[8] K. E. Parsopoulos, V. P. Plagianakos, G. D. Magoulas, and M. N. Vrahatis, "Objective function "stretching" to alleviate convergence to local minima," *Nonlinear Analysis*, vol. 47, no. 5, pp. 3419–3424, 2001.

[9] K. E. Parsopoulos and M. N. Vrahatis, "On the computation of all global minimizers through particle swarm optimization," *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 3, pp. 211–224, 2004.

[10] D. Zaharie, "Extensions of differential evolution algorithms for multimodal optimization," in *6th International Symposium of Symbolic and Numeric Algorithms for Scientific Computing*, 2004, pp. 523–534.

[11] G. R. Harik, "Finding multimodal solutions using restricted tournament selection," in *Proceedings of the 6th International Conference on Genetic Algorithms*, San Francisco, CA, USA, 1995, pp. 24–31.

[12] B.-Y. Qu and P. Suganthan, "Novel multimodal problems and differential evolution with ensemble of restricted tournament selection," in *IEEE Congress on Evolutionary Computation*, 2010, pp. 1–7.

[13] J. Li, M. E. Balazs, G. T. Parks, and P. J. Clarkson, "A species conserving genetic algorithm for multimodal function optimization," *Evolutionary Computation*, vol. 10, no. 3, pp. 207–234, 2002.

[14] R. Storn and K. Price, "Differential evolution – a simple and efficient adaptive scheme for global optimization over continuous spaces," *Journal of Global Optimization*, vol. 11, pp. 341–359, 1997.

[15] K. Price, R. M. Storn, and J. A. Lampinen, *Differential Evolution: A Practical Approach to Global Optimization (Natural Computing Series)*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2005.

[16] S. Das and P. N. Suganthan, "Differential evolution: A survey of the state-of-the-art," *IEEE Transactions on Evolutionary Computation*, vol. 15, no. 1, pp. 4–31, 2010.

[17] M. G. Epitropakis, D. K. Tasoulis, N. G. Pavlidis, V. P. Plagianakos, and M. N. Vrahatis, "Enhancing differential evolution utilizing proximity-based mutation operators," *IEEE Transactions on Evolutionary Computation*, vol. 15, no. 1, pp. 99–119, 2011.

[18] R. Thomsen, "Multimodal optimization using crowding-based differential evolution," in *IEEE Congress on Evolutionary Computation*, vol. 2, 2004, pp. 1382–1389.

[19] X. Li, "Efficient differential evolution using speciation for multimodal function optimization," in *Proceedings of the 2005 conference on genetic and evolutionary computation, GECCO 2005*, 2005, pp. 873–880.

[20] J. Li, X. Li, and A. Wood, "Species based evolutionary algorithms for multimodal optimization: A brief review," in *2010 IEEE Congress on Evolutionary Computation (CEC)*. IEEE, July 2010, pp. 1–8.

[21] J. I. Rönkkönen and J. Lampinen, "An extended mutation concept for the local selection based differential evolution algorithm," in *Proceedings of the 9th annual conference on genetic and evolutionary computation*. London, England: ACM, 2007, pp. 689–696.

[22] J. Rönkkönen, X. Li, V. Kyrki, and J. Lampinen, "A framework for generating tunable test functions for multimodal optimization," *Soft Computing - A Fusion of Foundations, Methodologies and Applications*, pp. 1–18, 2010.

[23] R. Brits, A. P. Engelbrecht, and F. V. D. Bergh, "A niching particle swarm optimizer," in *Proceedings of the Conference on Simulated Evolution And Learning*, 2002, pp. 692–696.

[24] K. Deb and A. Saha, "Finding multiple solutions for multimodal optimization problems using a multi-objective evolutionary approach," in *Proceedings of the 12th annual conference on genetic and evolutionary computation*. Portland, Oregon, USA: ACM, 2010, pp. 447–454.

[25] X. Li, "Niching without niching parameters: Particle swarm optimization using a ring topology," *IEEE Transactions on Evolutionary Computation*, vol. 14, no. 1, pp. 150–169, 2010.

[26] E. L. Yu and P. N. Suganthan, "Ensemble of niching algorithms," *Information Sciences*, vol. 180, pp. 2815–2833, August 2010.

[27] J. Kennedy, "Small worlds and mega-minds: effects of neighborhood topology on particle swarm performance," in *Proceedings of the 1999 Congress on Evolutionary Computation, 1999. CEC '99*, vol. 3, 1999, pp. 3 vol. (xxxvii+2348).

[28] J. Kennedy and R. Mendes, "Population structure and particle swarm performance," in *Proceedings of the 2002 Congress on Evolutionary Computation, 2002. CEC '02*, vol. 2, 2002, pp. 1671–1676.

[29] M. N. Vrahatis, B. Boutsinas, P. Alevizos, and G. Pavlides, "The new k-windows algorithm for improving the k-means clustering algorithm," *Journal of Complexity*, vol. 18, pp. 375–391, March 2002.