Region Based Memetic Algorithm with LS chaining

Benjamin Lacroix
Universidad de Granada,
Department of Computer Science and
Artificial Intelligence
Email: benjamin@decsai.ugr.es

Daniel Molina
Universidad of Cádiz,
Department of Computer Science
and Engineering
Email: daniel.molina@uca.es

Francisco Herrera
Universidad de Granada,
Department of Computer Science and
Artificial Intelligence
Email: herrera@decsai.ugr.es

Abstract—Memetic algorithms with an appropriate trade-off between the exploration and exploitation can obtain very good results in continuous optimisation. That implies the evolutionary algorithm component should be focused in exploring the search space while the local search method exploits the achieved solutions. In a previous work, it was proposed a memetic algorithm, MA-LSCh-CMA, that was able to work with a local search method, CMA-ES, with a great exploitation factor, but without a mechanism to maintain diversity and avoid competition between the evolutionary algorithm and CMA-ES. In this work, we propose a variation of this algorithm, called RMA-LSCh-CMA, adding a niching strategy that divide the domain search in equal hypercubes. The experimental results obtained show that the new version is statistically better than the previous one and is very competitive in comparisons with the state-of-the-art algorithm IPOP-CMA-ES, obtaining equivalent results on the medium and higher dimensions, although slightly better in the higher dimension.

I. INTRODUCTION

One of the main issues when designing an evolutionary algorithm (EA) [1] for real-coded parameter optimisation problems is to propose a good exploration of the search space and, at the same time, to exploit the most promising region to obtain high quality solutions. For that purpose memetic algorithms (MA) arose [2], [3], which are a hybridisation between EA and local search (LS) algorithms, mixing in one model the exploration power of EA and the exploitative power of the LS.

MAs with an appropriate trade-off between the exploration and exploitation can obtain accurate solutions, improving the search [4], [5]. Therefore, the key point in the design of a MA is to avoid competition between the EA and the LS to obtain that good trade-off. This is usually made increasing the exploration factor of the EA, as the exploitation should be made more efficiently by the LS component.

In a previous work [6] it was proposed an MA, MA-LSCh-CMA. The originality of this model lies in its ability to apply various times the LS on the same solution. The final state of the LS parameters after each LS application becomes the initial point of a subsequent LS application over the same solution, creating a *LS chain*. This way, MA-LSCh-CMA adapts the *intensity* of the LS to a solution in function of its quality.

MA-LSCh-CMA obtains good results, but the separation between the effort of the EA and the LS in their specific tasks, respectively exploration and exploitation, is not ensured.

In this work, we propose to add a niching strategy to the MA-LSCh-CMA model, in order to avoid competition between the EA and the LS by ensuring a high diversity in the EA's population. Contrarily to most niching strategies where the niches are defined around the solutions of the population, the niches are predefined as divisions of the search space. The search space is divided into equal hypercubes each of which represent one exclusion region, not allowing more than one solution in each one. Also, the LS method is initialised to explore inside these regions. This way, there is no competition between the EA and the LS method.

This paper is structured as follows. In Section II, we give a brief explanation of the MA-LSCh-CMA algorithm. In Section III, we describe in detail the new proposal, remarking the differences with the previous model. In Section IV, the experimental framework is designed. In Section V, several comparisons are carried out to study how the next algorithm improves the previous one, and to show if the proposal is a competitive algorithm, comparing with other state-of-the-art algorithm. Finally, in Section VI, we present the conclusions and future works.

II. THE MA-LSCh-CMA ALGORITHM

This section briefly describes the general scheme of the MA-LSCh-CMA algorithm and its main components. More details can be seen in [6].

A. General scheme

MA-LSCh-CMA was designed with the idea that the LS should be applied with higher intensity on the most promising regions. By promising regions, we consider the areas/regions where the solutions are maintained the most time in the population for their good fitness.

The MA-LSCh-CMA is a steady state MA which alternatively applies a Steady-State Genetic Algorithm(SSGA) as EA [7], and a CMA-ES [8] as LS method with an I_{str}. This hybridisation model allow the same solution improve several times, creating *LS chain*. Also, it uses a mechanism to store with the solution the final state of the LS parameters after each LS application. In this way, the final state of a LS application over a solution will be used as the initial point of a subsequent LS application over the same solution, *continuing* the LS. The general algorithm can be seen in Algorithm 1.

Algorithm 1 Pseudocode of MA-LSCh-CMA
1: Generate the initial population
2: **while** not termination-condition **do**
3:     Perform the SSGA with n_{frec} evaluations
4:     Build the set S_{LS} of individuals which can be refined by LS
5:     Pick the best individual c_{LS} in S_{LS}
6:     **if** c_{LS} belongs to an existing LS chain **then**
7:         Initialise the LS operator with the LS state stored with c_{LS}
8:     **else**
9:         Initialise the LS operator with the default LS parameters
10:     **end if**
11:     Apply the LS algorithm to c_{LS} with I_{str} evaluations, giving c_{LS}^r
12:     Replace c_{LS} by c_{LS}^r
13:     Store the final LS state with c_{LS}^r
14: **end while**

To select the individual c_{LS} to which the LS will be applied, the following process is used (steps 3 and 4):

1) The set S_{LS} is build with the individuals of the population that:
   a) have never been improved by the LS.
   b) have been improved by the LS but with an improvement (in fitness) superior to δ_{LS}^{min}.
2) If $|S_{LS}| \neq 0$, the LS is applied on the best individual in S_{LS}. If S_{LS} is empty, the whole population is reinitialised except for the best individual which is maintained in the population.

With this mechanism, if SSGA obtains a next best solution, it should be improved by the LS in the following application of the LS method.

B. The EA

The SSGA applied was specifically designed to promote high population diversity levels by means of the combination of the $BLX-\alpha$ crossover operator [9] with a high value for its associated parameter ($\alpha = 0.5$) and the *negative assortative mating* strategy (NAM) [10]. Diversity is favoured as well by means of the BGA mutation operator. The replacement strategy used is *Replacement Worst, RW*. The combination *NAM-RW* produces a high selective pressure.

C. The LS

The continuous LS algorithm is CMA-ES [8]. This algorithm is the *state-of-the-art* in continuous optimisation. Thanks to the adaptability of its parameters, its convergence is very fast and obtains very good results. CMA-ES is an algorithm that uses a distribution function to obtain new solutions, and adapt the distribution around the best created solutions.

Its only parameters are the initial average of the distribution $\vec{m}$ and the initial σ. MA-LSCh-CMA sets the individual to optimise c_{LS} as $\vec{m}$, and as the initial σ value the half of the distance of c_{LS} to its nearest neighbour in the EA's population.

III. Region based MA-LSCh-CMA

This section present the basic concepts of the novel niching strategy that we included in the MA-LSCh-CMA.

Most EA for continuous optimisation are designed to offer a good trade-off between exploration and exploitation. However, when implemented in a MA framework, they are being used for their exploration ability as the exploitation part is performed more efficiently by the LS algorithm.

We propose here to add to the MA-LSCh-CMA a niching strategy to maintain the certain diversity in the EA's population. Through this niching strategy, we propose a clearer separation between the exploration effort done by the EA and the exploitation task of the LS method.

In Section III-A, we describe the proposed niching strategy. Including such niching strategy implied two major modifications in the MA-LSCh-CMA, the redefinition of the EA, explained in Section III-B and the initial parameters of the LS explained in Section III-C.

A. The region based MA

Contrarily to most niching strategies where the niches are defined by the area surrounding solutions of the population, we propose here a strategy in which the niches are predefined as divisions of the search space, divided into hypercubes of equal size called here regions. This definition of a niche is illustrated in Figure 1. Each dimension is divided into ND divisions creating a grid of equal hypercubes, that represent exclusive regions (niches) which can contain only one solution.
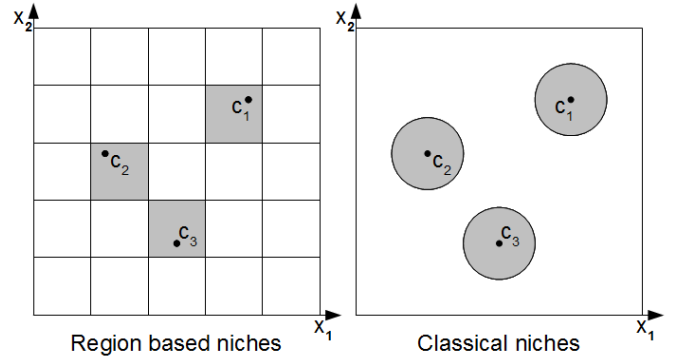

Fig. 1. Different niching strategy

B. The SSGA in a region-based MA

One of the key issues in niching strategies is to decide what to do with a solution generated in the exclusion area of an other solution. The modifications to the SSGA are described in Algorithm 2. It consists in not allowing the generation of a solution by the SSGA in a region that is already occupied by another solution in the population if this solution is optimised. By optimised, we refer to the fact that the last LS applied to this solution has not brought enough improvements (upper

than δ_{LS}^{min}). Then, if a solution is optimised, we consider its neighbourhood (and by consequence the region it lies in) has sufficiently been explored. On the other hand, if the solution is not optimised, the EA can replace it with a solution with a better fitness in that region. That way, we avoid unnecessary LS evaluations within the region to get a higher quality solutions in it. In the latter case, instead of replacing the worst individual in the population, the new solution will replace (if it is better) the solution lying in the same region. This way, we ensure that the population will not hold two solutions in the same region.

Algorithm 2 Pseudo-code for the region-based SSGA

1: Randomly generate the population
2: **while** not termination-condition **do**
3:     Select two parents in the population
4:     Create an offspring c_n using crossover and mutation
5:     **if** c_n falls in a region containing an individual c_o **then**
6:         **if** c_o is considered optimised **then**
7:             Mutate c_n using the BGA mutation [11] and go back to 5
8:         **end if**
9:     **end if**
10:    **if** c_n falls in a region containing an individual c_o **then**
11:        Replace c_o by c_n if $f(c_o) > f(c_n)$
12:    **else**
13:        Replace the worst individual c_{worst} in the population if $f(c_{worst}) > f(c_n)$
14:    **end if**
15: **end while**

C. The LS in a region-based MA

In order to put the emphasis on dedicating the exploration task to the EA and the exploitation one to the LS, we have also modified the strategy for initialising the parameters of the LS. In the MA-LSCh-CMA, the initial step of the CMA-ES is set between the area limited by its neighbouring solutions. Here the CMA-ES initial step is set according to the size of the region. We want to ensure that the close surrounding of a solution are properly explored by the LS as this task will not be done by the EA. The initial standard deviation is set to half the size of the region. Apart from this modification, in order to allow a proper refinement of the solution, the LS is not influenced by the divisions of the search space. However, if at the end of the LS application, the new solution is in a region occupied, the best one is kept and the other one is replaced by a randomly generated solution.

IV. EXPERIMENTAL FRAMEWORK

For the experimental sections, we have used the benchmark proposed in the Special Session on Real Parameter Optimisation organised in the 2005 IEEE Congress on Evolutionary Computation (CEC'2005). The complete description of the functions can be seen in [12]. Note that every functions have

been shifted to ensure that the global optimum is not in the center of search space.

In order to be able to compare our results with other algorithms involved in the competition, we followed the requirements descibed in [12] :

- Each algorithm is run 25 times for each test function, and the average of error of the best individual of the population is computed. The *function error* value for a solution x is defined as $(f(x) - f(x^*))$, where x^* is the global optimum of the function.
- The study has been made with dimensions $D = 10$, $D = 30$, and $D = 50$.
- The maximum number of fitness evaluations for each run is $10{,}000 \cdot D$, where D is the dimension of the problem.
- Each run stops either when the error obtained is less than 10^{-8}, or when the maximal number of evaluations is achieved.

For the comparisons we use non-parametric tests [13], because [12] shows that for this benchmarks parametric test are not possible. In particular, we have considered two tests: Wilcoxon's test, to compare two algorithms, and Iman and Davenport's test and the *post-hoc* Holm's test to compare more algorithms.

V. EXPERIMENTAL RESULTS

We have carried out the experiments of RMA-LSCh-CMA using the parameters' values proposed by the authors of MA-LSCh-CMA [6], for a reliable comparison: Popsize is 80, $N_{NAM} = 3$, $P_{mutation} = 0.125$, $I_{str} = 500$, and $Ratio_{LS} = 0.5$. The same way, the CMA-ES population is $popsize_{cmaes} = 4 + 3\log(D)$, where D is the dimension, and the offspring size is set to $popsize_{cmaes}/2$

This section is thus composed as follows. In Section V-A, we study the influence of the number of divisions on the search. In Section V-B, we demonstrate through a few examples that the diversity is better maintained with the regionalisation of the search space. In Section V-C, we compare the results of our model against the original one, MA-LSCh-CMA, and, in Section V-D, to the state-of-the-art, IPOP-CMA-ES.

A. Study of the number of divisions

When implementing a niching strategy, the most critical parameter is the size of the niches. Here, the size of the niches are defined by the number of divisions per dimension ND. We test in this section three values of ND: 10, 50 and 100. The detailed results can be found in the Appendix.

Figure 2 shows the average rankings obtained by the RMA-LSCh-CMA instances with different ND values on the 25 test functions with dimensions $D = 10$, 30, and 50. The mean rankings correspond to the average of the ranking of each algorithm on each function. We can note that the influence on the number of divisions depends on the dimension. Indeed, for smaller dimensions, a smaller number of divisions obtains better results while for higher dimensions, a higher number of divisions performs better on higher dimensions.
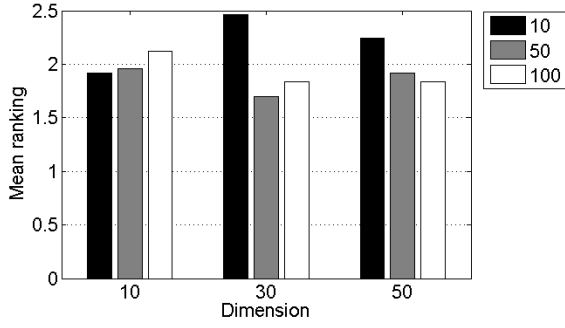
Fig. 2. Mean rankings obtained by RMA-LSCh-CMA with different number of divisions over every functions of the CEC'2005 benchmark. The lower columns corresponds to the best algorithms.

TABLE I
IMAN-DAVENPORT TEST FOR SIGNIFICANT DIFFERENCE BETWEEN THE INSTANCES OF R-MA-LSCH-CMA WITH $ND = 10$, $ND = 50$ AND $ND = 100$

Dimension	p-value	Significant differences?
10	0.763	No
30	0.014	Yes
50	0.333	No

We first applied the Iman-Davenport's test to the results of the three instances of the model to assess any significant differences. Table I shows that there are no significant differences in dimension 10 and 50. In dimension 30, we can observe significant differences, thus we apply the Holm's test using the algorithm with best fitness, ND=50, as the *control algorithm*. Table II show the results. It can be observed that ND=50 gives significantly better results than with ND=10 and they are fairly equivalent with ND=100. Thus, for the following experiments, we set the number of divisions to 50 (ND=50).

TABLE II
COMPARISON USING HOLM'S TEST WITH $\alpha = 0.05$ OF THE INSTANCE WHERE $ND = 50$ AGAINST THE OTHER INSTANCES

i	ND	$z = (R_0 - R_i)/SE$	p	α/i	Significant difference?
2	10	2.687	0.007	0.025	Yes
1	100	0.495	0.621	0.05	No

B. Diversity study

One objective of this new model is to maintain a certain diversity in the population. In this section, we analyse the evolution of the diversity along the search in each instances of R-MA-LSCh-CMA and plotted it against the diversity observed in the MA-LSCh-CMA. We chose the *distance-to-average-point* measure as described in [14]. Figure 3 represents the evolution of the diversity of each EA's population.

Two important points are to be noted here: first, the number of divisions influences the diversity. Indeed, the smaller the number of divisions, the higher is the diversity in the population remains along the search. Also, RMA-LSCh-CMA

maintains a higher diversity in the EA's population than MA-LSCh-CMA.

C. Comparison with the MA-LSCh-CMA

The original purpose of this work was to improve the promising results of the MA-LSCh-CMA. We analyse in the improvements brought by the proposed niching strategy to this algorithm.

TABLE III
WILCOXON SIGNED RANK TEST RESULTS OF RMA-LSCH-CMA VS MA-LSCH-CMA

| | $R+$ | $R-$ | |
Dim	RMA-LSCh-CMA	MA-LSCh-CMA	$p - value$
10	209.5	92	0.100
30	199.5	102	0.177
50	237.5	87.5	0.043

Table III shows the Wilcoxon signed rank obtained when comparing both algorithms. We can see that the new model obtains better results in every dimensions and is statistically better in dimension 10 (with $\alpha = 0.1$) and 50 (with $\alpha = 0.05$). The significant improvement in dimension 50 is particularly interesting for being the most difficult optimisation cases.

D. Comparison with the IPOP-CMA-ES

In this section, we compare the efficiency of our algorithm with IPOP-CMA-ES [15]. IPOP-CMA-ES is the winner of the CEC2005 Real-Parameter.

Table IV shows the results of the comparison with IPOP-CMA-ES applying the Wilcoxon's test. We can see that although the IPOP-CMA-ES is significantly better in dimension 10, we do not obtain statistical differences in dimension 30 and 50.

VI. CONCLUSION

This paper presents a MA, RMA-LSCh-CMA, in which we demonstrated the importance of separating the effort of the global search from the refinement of the solution. To avoid the competition between the EA and the LS, we have decided to divide the search space into predefined regions ensuring each region only contains one solution. This division on the search space led to two major modifications in the MA-LSCh-CMA used as basis. The first one is to ensure that only one solution of the EA's population can be present in a region. This maintains a certain diversity in the population to be maintained and that the close neighbourhood of a solution will not be explored by the EA. The second modification is

TABLE IV
WILCOXON SIGNED RANK TEST RESULTS OF RMA-LSCH-CMA VS IPOP-CMA-ES

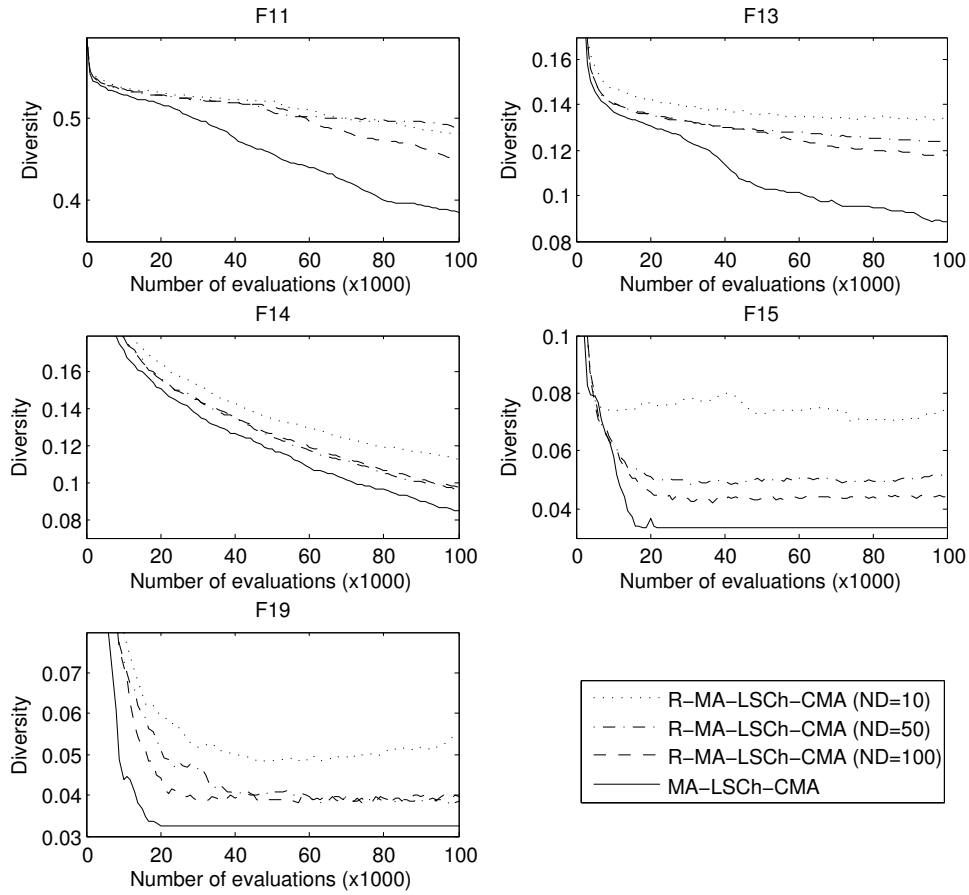| | $R+$ | $R-$ | |
Dim	RMA-LSCh-CMA	MA-LSCh-CMA	$p - value$
10	255.5	59.5	0.011
30	172	153	0.798
50	128	172	0.529

Fig. 3. Evolution of the diversity in the population of the EA of the RMA-LSCh-CMA and MA-LSCh-CMA for different functions

the initialisation of the LS. It is now initialised according to the size the regions to ensure that the region the solution to which the LS is applied is properly explored.

We have experimented RMA-LSCh-CMA using the CEC'2005 benchmark. The results obtained are significantly better than obtained by MA-LSCh-CMA. We have also proven that the diversity in the EA's population is maintained a higher level. We have observed that the number of divisions influenced the quality of the results according to the dimension. Finally, we have compared the results with the IPOP-CMA-ES obtaining in dimension 30 and 50 statistically equivalent results, although slightly better in dimension 50.

To summarize, RMA-LSCh-CMA improves in fitness and in diversity the original MA-LSCh-CMA, and becomes a very competitive algorithm.

As future work, considering the good results obtained in dimension 50, we are going to study its application to higher dimensions problems. Also, because there is a relation between the number of divisions and the dimension, we will try to design a model with an adaptable number of divisions.

VII. ACKNOWLEDGEMENT

REFERENCES

[1] T. Bck, D. B. Fogel, and Z. Michalewicz, Eds., *Handbook of Evolutionary Computation*. Bristol, UK: IOP Publishing Ltd., 1997.
[2] N. Krasnogor and J. Smith, "A tutorial for competent memetic algorithms: model, taxonomy, and design issues," *IEEE Transactions on Evolutionary Computation*, vol. 9, no. 5, pp. 474–488, oct. 2005.
[3] P. A. Moscato, *Memetic algorithms: a short introduction*. McGraw-Hill, London, 1999, pp. 219–234.
[4] L. Davis, *Handbook of Genetic Algorithms*. Van Nostrand Reinhold, New York, 1991.
[5] D. E. Goldberg and S. Voessner, "Optimizing global-local search hybrids," in *Proceedings of the Genetic and Evolutionary Computation Conference*, W. Banzhaf, J. Daida, A. E. Eiben, M. H. Garzon, V. Honavar, M. Jakiela, and R. E. Smith, Eds., vol. 1, no. 99001. Morgan Kaufmann, 1999, pp. 220–228.
[6] D. Molina, M. Lozano, C. García-Martínez, and F. Herrera, "Memetic Algorithms for Continuous Optimisation Based on Local Search Chains," *Evolutionary Computation*, vol. 18, no. 1, pp. 27–63, Jan. 2010.
[7] D. Whitley, "The GENITOR Algorithm and Selection Pressure: Why Rank-Based Allocation of Reproductive Trials is Best," *Proceedings of the Third International Conference on Genetic Algorithms*, pp. 116–121, 1989.
[8] N. Hansen, S. D. Müller, and P. Koumoutsakos, "Reducing the Time Complexity of the Derandomized Evolution Strategy with Covariance Matrix Adaptation (CMA-ES)," *Evolutionary Computation*, vol. 1, no. 11, pp. 1–18, 2003.

[9] L. Eshelman and J. D. Schaffer, "Real-coded genetic algorithms in genetic algorithms by preventing incest," *Foundation of Genetic Algorithms 2*, pp. 187–202, 1993.

[10] C. F. Agostinho, C. Fernandes, and A. Rosa, "A Study on Non-random Mating and Varying Population Size in Genetic Algorithms Using a Royal Road Function," in *Proceedings of the 2001 IEEE Congress on Evolutionary Computation*. IEEE Press, 2001, pp. 60–66.

[11] H. Mülenbein and D. Schlierkamp-Voosen, "Predictive models for the breeding genetic algorithm in continuous parameter optimization," *Evolutionary Computation*, vol. 1, pp. 25–49, 1993.

[12] P. N. Suganthan, N. Hansen, J. J. Liang, K. Deb, Y. P. Chen, A. Auger, and S. Tiwari, "Problem Definitions and Evaluation Criteria for the CEC 2005 Special Session on Real-Parameter Optimization," Tech. Rep., 2005.

[13] J. Derrac, S. García, D. Molina, and F. Herrera, "A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms," *Swarm and Evolutionary Computation*, vol. 1, no. 1, pp. 3–18, 2011.

[14] R. K. Ursem, "Diversity-Guided Evolutionary Algorithms," in *Proceedings of the Congress on Evolutionary Computation*. IEEE Press, 2002, pp. 1633–1640.

[15] A. Auger and N. Hansen, "A Restart {CMA} Evolution Strategy with Increasing Population Size," in *2005 IEEE Congress on Evolutionary Computation*, 2005, pp. 1769–1776.

TABLE V

RESULTS IN DIMENSION 10 OF THE RMA-LSCH-CMA WITH VARIOUS VALUES OF ND

F/ND	10	50	100
F1	**1.00E-008**	**1.00E-008**	**1.00E-008**
F2	**1.00E-008**	**1.00E-008**	**1.00E-008**
F3	**1.00E-008**	**1.00E-008**	**1.00E-008**
F4	**1.00E-008**	**1.00E-008**	**1.00E-008**
F5	**1.00E-008**	**1.00E-008**	**1.00E-008**
F6	**1.00E-008**	5.68E-003	1.68E-003
F7	**1.00E-008**	**1.00E-008**	**1.00E-008**
F8	**2.04E+001**	**2.04E+001**	**2.04E+001**
F9	8.15E-001	7.96E-002	**1.00E-008**
F10	4.18E+000	2.35E+000	**1.83E+000**
F11	**3.32E-001**	1.29E+000	1.64E+000
F12	1.47E+002	**1.22E+002**	2.19E+002
F13	6.29E-001	5.69E-001	**4.78E-001**
F14	2.84E+000	2.52E+000	**2.15E+000**
F15	**2.13E+002**	2.67E+002	2.72E+002
F16	**8.43E+001**	9.09E+001	9.02E+001
F17	9.72E+001	9.34E+001	**9.28E+001**
F18	**7.79E+002**	8.47E+002	8.57E+002
F19	**7.63E+002**	8.03E+002	8.60E+002
F20	**7.51E+002**	8.21E+002	8.36E+002
F21	**7.47E+002**	7.70E+002	7.70E+002
F22	7.42E+002	7.35E+002	**7.30E+002**
F23	**9.31E+002**	9.47E+002	9.35E+002
F24	2.36E+002	**2.12E+002**	2.76E+002
F25	4.10E+002	**4.06E+002**	4.40E+002

TABLE VI

RESULTS IN DIMENSION 30 OF THE RMA-LSCH-CMA WITH VARIOUS VALUES OF ND

F/ND	10	50	100
F1	**1.00E-008**	**1.00E-008**	**1.00E-008**
F2	**1.00E-008**	**1.00E-008**	**1.00E-008**
F3	**1.00E-008**	1.06E-008	**1.00E-008**
F4	2.43E+001	4.02E-001	**2.98E-001**
F5	9.27E+001	3.77E+001	**5.70E+000**
F6	2.83E+001	**1.49E+001**	1.57E+001
F7	6.90E-004	**1.00E-008**	**1.00E-008**
F8	2.10E+001	**2.09E+001**	2.09E+001
F9	6.70E+000	2.73E-002	**5.69E-004**
F10	2.46E+001	1.79E+001	**1.71E+001**
F11	**4.04E+000**	1.24E+001	1.60E+001
F12	1.88E+003	**1.64E+003**	2.26E+003
F13	3.46E+000	2.50E+000	**2.17E+000**
F14	1.28E+001	**1.26E+001**	1.27E+001
F15	3.32E+002	3.15E+002	**3.14E+002**
F16	9.69E+001	8.57E+001	**7.55E+001**
F17	9.05E+001	**7.18E+001**	7.36E+001
F18	9.07E+002	**9.02E+002**	**9.02E+002**
F19	9.03E+002	**9.02E+002**	**9.02E+002**
F20	**9.03E+002**	9.06E+002	9.06E+002
F21	**5.00E+002**	**5.00E+002**	**5.00E+002**
F22	8.95E+002	**8.67E+002**	8.76E+002
F23	5.50E+002	**5.34E+002**	5.57E+002
F24	**2.00E+002**	**2.00E+002**	**2.00E+002**
F25	**2.10E+002**	2.11E+002	2.13E+002

TABLE VII

RESULTS IN DIMENSION 50 OF THE RMA-LSCH-CMA WITH VARIOUS VALUES OF ND

F/ND	10	50	100
F1	**1.00E-008**	**1.00E-008**	**1.00E-008**
F2	**1.00E-008**	1.01E-008	1.03E-008
F3	**1.00E-008**	1.05E-008	**1.00E-008**
F4	3.82E+003	**1.06E+003**	1.83E+003
F5	2.22E+003	1.82E+003	**1.70E+003**
F6	1.87E+001	**9.48E+000**	3.79E+001
F7	**1.00E-008**	1.08E-003	**1.00E-008**
F8	**2.11E+001**	**2.11E+001**	**2.11E+001**
F9	5.69E-001	2.19E-002	**1.00E-003**
F10	6.60E+001	**3.63E+001**	3.81E+001
F11	**1.11E+001**	2.61E+001	3.27E+001
F12	1.47E+004	**1.03E+004**	1.18E+004
F13	6.00E+000	4.52E+000	**4.06E+000**
F14	2.26E+001	2.23E+001	**2.21E+001**
F15	3.13E+002	3.57E+002	**3.01E+002**
F16	5.89E+001	7.12E+001	**5.33E+001**
F17	9.59E+001	8.47E+001	**5.92E+001**
F18	**8.72E+002**	8.97E+002	9.21E+002
F19	**8.21E+002**	9.21E+002	9.20E+002
F20	8.97E+002	**8.93E+002**	9.21E+002
F21	5.24E+002	5.12E+002	**5.00E+002**
F22	9.41E+002	9.35E+002	**9.12E+002**
F23	5.67E+002	**5.39E+002**	5.53E+002
F24	**2.00E+002**	**2.00E+002**	**2.00E+002**
F25	**2.14E+002**	2.15E+002	2.19E+002