Improved Topological Niching for Real-Valued Global Optimization

Mike Preuss

Chair of Algorithm Engineering, Computational Intelligence Group,
Dept. of Computer Science, Technische Universität Dortmund, Germany
mike.preuss@tu-dortmund.de

Abstract. We show how nearest-better clustering, the core component of the NBC-CMA niching evolutionary algorithm, is improved by appyling a second heuristic rule. This leads to enhanced basin identification for higher dimensional (5D to 20D) optimization problems, where the NBC-CMA has previously shown only mediocre performance compared to other niching and global optimization algorithms. The new method is integrated into a niching algorithm (NEA2) and compared to NBC-CMA and BIPOP-CMA-ES via the BBOB benchmarking suite. It performs very well on problems that enable recognizing basins at all with reasonable effort (number of basins not too high, e.g. the Gallagher problems), as expected. Beyond that point, niching is obviously not applicable any more and random restarts as done by the CMA-ES are the method of choice.

1 Introduction

The idea to apply niching in *evolutionary optimization* (EC) is almost as old as EC itself, starting with Sharing and Crowding in the 1970s. The general scheme is to host multiple explicitly or implicitly separated populations in one optimization run and by driving these into the better regions of a multimodal problem landscape, provide more than one good solution at once. However, they can also be considered global optimizers as on complex landscapes, it is necessary to retrieve as many good local optima as possible to determine the global one (or at least a very good one). Premature convergence has never been completely ruled out and probably will never be, otherwise one would have to safely recognize where the global optimum is before actually approaching it. We believe that while it is possible to identify some basins of attraction before too many function evaluations are spent, trying to detect which of these would host the global optimum with some safety is impossible without actually applying several restricted (or local) searches.

Generally, niching relies on geometrical properties of the populations (as distances), which makes it less and less applicable if the number of search space dimensions rises. Most methods are experimentally tested in 2D and maybe 3 to 5D, rarely on 10 or more dimensions. [9] gives a good overview over the "classical" niching strategies and some of their newer variants, also taking different

C. Di Chio et al. (Eds.): EvoApplications 2012, LNCS 7248, pp. 386–395, 2012.

Clearing procedures into account. Today, there are still new works in the niching context every year, some of the most recent being [8], [10], and [5]. And we can assess some development from fixed niche radii towards either adaptive radii or even shapes [8], while the topological methods do completely without radii, with [10] and without [7] employing additional evaluations. An interesting related approach that follows more the parallelization ideas of island models [3] is the Particle Swarm CMA-ES [4]. It is especially designed for multi-funnel landscapes (arguably second-order multimodal problems) and shows good results on the CEC 2005 test set even for 10 to 50 dimensions.

As documented in [10], many niching methods perform quite well for the lower dimensional classical test problems as 6-hump camel back, but extensive testing on recent benchmarks as the BBOB 2009/2010 test set[1] is rarely seen. To our knowledge, the *nearest-better clustering cumulative matrix adaptation evolution strategy* (NBC-CMA-ES) [5] is the only niching algorithm which has been investigated on this benchmark yet. In this work, we enhance the NBC-CMA by two strong modifications:

a) We add a second heuristic rule to the basin identification mechanism that is known as nearest-better clustering that is especially tasked at higher dimensional search spaces.
b) We change the local search scheme of the NBC-CMA from a parallel one to a more sequential one that is more robust against basin identification mismatches.

The resulting algorithm is termed NEA2 and we experimentally investigate how each of the modifications increases performance either concerning basin identification or by comparing to the NBC-CMA on the BBOB problem collection. However, we start with giving a very rough general picture of the functioning of the NBC-CMA in order to enable understanding the changes.

2 NBC-CMA General Scheme

The algorithm starts with setting out a larger start population (the default size is 40D) as evenly as possible in the search space. This can be done by employing a space-filling method as *Latin Hypercube Sampling* (LHS) but in principle, a uniform random distribution may also be used, this just lowers the basin identification quality gradually. On this sample, the NBC (see next section) is run and returns a split into separate populations. These are set up as separate CMA-ES runs, respecting the default population size of the CMA-ES according to the search space dimensionality: $\lambda = 4 + \lfloor 3ln(D) \rfloor$ (CMA-ES default parameters are collected nicely here [2]). These separate CMA-ES instances are then run for one generation, the resulting individuals collected, and the basin identification mechanism applied again as described above, starting the loop again. Learned step-sizes and covariance-matrices are stored also in the individuals and after

[1] `http://coco.gforge.inria.fr/doku.php`

clustering and redistributing individuals into populations, each population uses the values found in the best individual (otherwise these values could not be adapted over time). The CMA-ES restart conditions are also applied, so that in case of stagnation the whole algorithm is started over.

As the NBC returns a previously unknown number of basins, and taking into account that it is a heuristic and can fail (if so often by recognizing several basins where there is only one), it is necessary to limit the maximum number of niches pursued at the same time. Of the identified niches, only the best $nich_{max}$ ones are considered, where $nich_{max}$ has a default value of 20. This is connected to a major problem for the NBC-CMA: If too many basins have been erroneously detected, many evaluations are wasted because several populations follow the same peak and eventually progress to the same optimum. This downgrades the performance enormously on very simple functions (e.g. the sphere function). Thus, we clearly have a strong motivation to increase the reliability of NBC, the basin identification method.

3 Nearest Better Clustering with Rule2

This basin identification method has been introduced in [6]. It works by connecting every search point in the population to the nearest one that is better and cutting the connections that are longer than $2\times$ the average connection. The remaining connections determine the found clusters by computing the weakly connected components. The reasoning behind cutting the longest connection is that they are very likely to reach out into another basin; these points seem to be locally optimal, at least considering the given sample. The scheme has huge advantages compared to other clustering methods as no additional evaluations beside the initial scan are needed, and neither shape nor size of the basins is predefined but recognized from the sample. It works very well for a reasonably large populations in two or three dimensions, but increasingly fails if the number of dimensions increases.

Therefore, we now add a second additional cutting rule: For all search points that have at least 3 incoming connections (it is the nearest better point for at least 3 others), we divide the length of its own nearest-better connection by the median of its incoming connections. If this is larger than a precomputed correction factor b, the outgoing connection is cut (and we have one additional cluster). Both rules are applied in parallel, that is, the edges to cut due to rule 2 must be computed before actually cutting due to rule 1. Edges cannot be cut more than once, so if both rules apply, this is not specially treated. Algorithm 1 presents the updated NBC method containing both rules. As the basin identification capability of rule 1 in 2D seems to be sufficient, rule 2 is only applied in at least 3 dimensions.

The motivation for rule 2 was that in a sufficiently large samples (at least around $40 \times D$), often points with several incoming connections are found whose outgoing edge is not cut with rule 1 because it is longer than all the incoming ones but not one of the longest of the the whole sample. However, determination

Algorithm 1. Nearest-better clustering (NBC) with rule2

```
1 compute all search points mutual distances;
2 create an empty graph with num(search points) nodes;
  // make spanning tree:
3 forall the search points do
4     find nearest search point that is better; create edge to it;
  // cut spanning tree into clusters:
5 RULE1: delete edges of length > φ · mean(lengths of all edges);
6 RULE2: forall the search points with at least 3 incoming and 1 outgoing edge
  do
7     if length(outgoing edge)/median(length(incoming edges)) > b then
8         cut outgoing edge;
  // find clusters:
9 find connected components;
```

of the correction factor b for rule 2 is not as trivial as setting the cutting factor to 2 for rule 1. It is experimentally derived and presumed to depend on D and the sample size S. As we want to recognize only one cluster on unimodal problems and ideally two or more on multimodal problems, we employ two extreme test functions, namely the sphere (2) and a deceptive function (1) with 2^D optima, located in the corners of the hypercube, restricting the search space to $[0,1]^D$.

$$dec(\boldsymbol{x}) = \sum_{i=1}^{D} 1 - 2 * abs(0.5 - x_i) \tag{1}$$

$$sphere(\boldsymbol{x}) = \sum_{i=1}^{D} (x_i - 0.5)^2 \tag{2}$$

Figure 1 shows the recognized number of clusters on our two problems while varying b for up to $20D$ and sample sizes of 300. This already provides a good idea how to set b in order to prevent obtaining more than 1.1 clusters on average for the sphere, and it also shows that with these values of b, we can still expect to obtain at least 2 clusters on average on the deceptive function (white areas). However, the corridor is shrinking for higher dimensions. By applying a simple interval search method for finding the right b for every combination of (S, D), using at least 100 repeats for every measurement, we obtain a sequence of points we can use for a linear regression over $log10(S)$. This makes sense as the figure already shows a near linear structure (in logarithmic scaling). The resulting formula is given in (3).

$$\begin{aligned} b(S,D) = (&-4.69 * 10^{-4} * D^2 + 0.0263 * D + 3.66/D - 0.457) * log10(S) \\ &+7.51e-4 * D^2 - 0.0421 * D - 2.26/D + 1.83 \end{aligned} \tag{3}$$

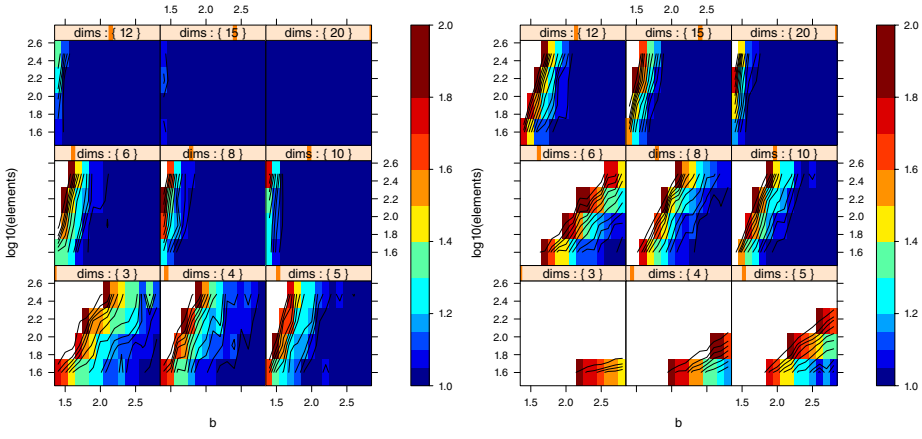

Fig. 1. Number of clusters found over up to 300 elements in dimensions 3 to 20, by applying different values for correction factor b. Left: Sphere (we must not obtain more than 1 cluster here), right: Deceptive test function with 2^D basins. b has been chosen to result in at most 1.1 clusters on the sphere. The gap to the corresponding value for the deceptive function shows that still at least 2 clusters can be found reliably here up to around $20D$.

3.1 Experiment: Effectiveness of Rule 2

In order to assess if rule 2 helps to identify more (meaningful) clusters than before especially in higher dimensions we compare the average number of obtained clusters and the pairwise accuracy of all points on the sphere and the deceptive problem with and without applying rule 2. The *pairwise accuracy* (*pa*) counts the fraction of the points belonging to the same basin of attraction of all pairs of points (without ordering) that have been put into the same cluster by the clustering method. Note that the *pa* measure gets very small rapidly if the number of clusters is much lower than the one of the existing basins (some clusters cover several basins), but it may still be a good first orientation concerning the quality of a clustering.

Setup. We run the clustering with and without rule 2 in 3,4,5,6,8,10,12,15,20 dimensions, on the sphere and the deceptive function, respectively, with 50 repeats. The sample sizes are always $40 \times D$.

Task. To assess a successful improvement (due to rule 2), we demand that a) the average number of obtained clusters on the deceptive function is significantly higher (Wilcoxon rank-sum test at 5%) than the number of clusters with rule 1 alone, b) the cluster numbers on the sphere function do not surpass 1.1 on average, and b) that the pairwise accuracy for the case with rule 2 is significantly (same test) better than the one without at least up to 10 dimensions

(10 dimensions means 1024 optima, as the *pa* has quadratic nature, the values will become nearly 0 quickly if D increases).

Results/Visualization. The mean numbers of clusters and pairwise accuracies on the deceptive function are given on the right side of fig. 2 (the left side gives an example in 3D), on the sphere function the method always returns 1 without rule 2, the mean cluster numbers with rule 2 are: $1.06, 1, 1.1, 1.08, 1.1, 1.08, 1.16, 1.14, 1.18$ in dimensions $3, 4, 5, 6, 8, 10, 12, 15, 20$. The difference in cluster numbers on the deceptive function is significant for all dimensions, and for the accuracies up to dimension 15.

Observations. Cluster numbers recognized with rule 2 increase up to around 6 for ten dimensions, then fall again. Without rule 2, values are practically 1 for $D > 3$. The accuracy values for rule 2 are always around double the size than without.

Discussion. It is clear that the deceptive function resembles a more or less ideal case, real problems may be much more difficult. Nevertheless, it is important that the improved clustering method at least works well under these conditions. It clearly does so, as seen from the figures, for $D > 3$ the difference is enormous. Required statistical significance has also been achieved, so that we can expect to improve also the performance of a niching method built on the improved basin identification method we deliver. However, the *pa* measure is a bit difficult to handle as its values get very small with increasing number of basins. One may think of an alternative.

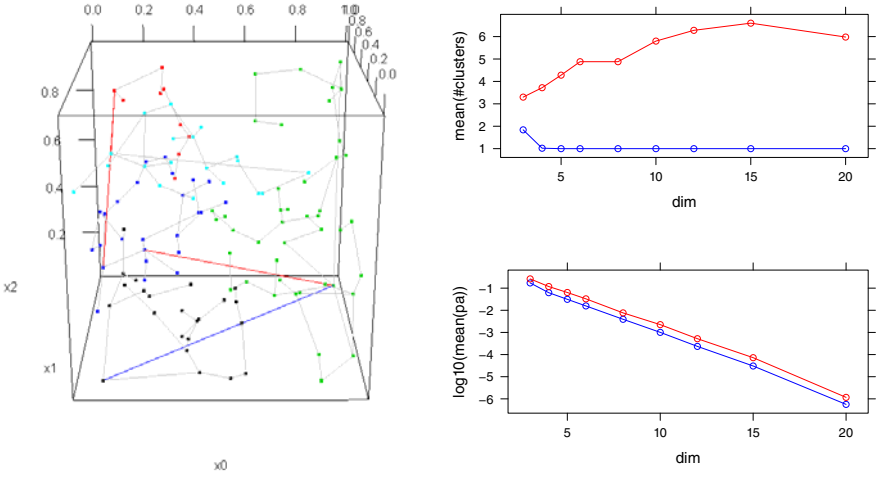

Fig. 2. Left: Example clustering on the 3D deceptive test function, the 120 points are colored per cluster, gray lines are the nearest better edges, blue lines represent the edges cut due to rule 1, red lines the ones cut due to rule 2. Right: Average number of recognized clusters (top) and pairwise accuracies (bottom) for $40 \times D$ points, with rule 2 (red) and without (blue). Note the logarithmic scaling in the lower figure, Absolute values for red are constantly about factor 2 higher.

4 NEA2: Parallel Niching and Sequential Local Optimization

The NBC-CMA as suggested in [5] had its difficulties with performing adequately on unimodal functions. The reason is that as the NBC basin identification mechanism is a heuristic, it is prone to erroneously building more than one cluster where this is unnecessary. The new variant of the algorithm we suggest here, NEA2, overcomes this problem by switching from a BFS-like to a DFS-like search in which the clusters are treated sequentially sorted according to their best members (best first). Should the problem be less multimodal then detected, (e.g. unimodal), NEA2 would perform very similar to the CMA-ES (without heuristic population enlargement as e.g. implemented by the IPOP- [1] and BIPOP-CMA) as every start point leads to the same optimum.

Another advantage of a sequential method is that it is much easier to detect if we approach an already found local optimum again because we already know the local optimum approximations resulting from the previous "local searches". However, this is currently not exploited. We have to admit that changing from parallel to sequential searches is unavoidable because many more clusters are now found also in higher dimensions, where NBC-CMA detected only one. Summarizing, the whole NEA2 method works as given in algorithm 2.

Algorithm 2. NEA2 (with updated NBC)

```
1 distribute an evenly spread sample over the search space;
2 apply the NBC to separate the sample into populations according to detected
  basins;
3 forall the populations do
4 |  run a CMA-ES until restart criterion is hit;
  // start all over:
5 if !termination then
6 |  goto step 1
```

5 Experimental Comparison

It is clear that the BBOB test set of 24 functions does not resemble the ideal test bed for niching methods as one would never employ one if it is highly likely that the treated problem is unimodal. However, we compare the new variant NEA2 to its ancestor NBC-CMA and also to the BIPOP-CMA-ES (winner of the BBOB 2009 competition) on this benchmark suite because a) there is a considerable amount of data and knowledge on the performance of different algorithm types generated during the last two BBOB competitions, and b) there are at least

some multimodal functions without strong global structure (which would be the setting niching algorithms are targetted at). These functions are the one in the last group, f20 (Schwefel), f21 (Gallagher 1), f22 (Gallagher 2), f23 (Katsuuras), and f24 (Lunacek). While f20 can be considered a deceptive problem, f21/f22 are moderately multimodal, f23/f24 are very highly multimodal and f24 additionally possesses a funnel structure.

Setup. NEA2 is run with a maximum of $300,000$ function evaluations as was done with NBC-CMA in [5]. This may not enough for a full picture, but enables a first comparison. NEA2 employs an initial sample of $40 \times D$ that is spread over the search space by means of an LHS (Latin Hypercube Sampling). All CMA-ES specific parameters (used inside the NEA2) are set to their defaults, the initial step size is 1.5. NBC-CMA had used an initial sample of 100, a fixed population size of $\mu = 5, \lambda = 10$, and a maximum of 20 concurrent populations. Runs are immediately stopped if the BBOB frameworks signals hitting the global optimum. We run over the dimensions $2, 3, 5, 10, 20$, all 15 instances provided by the BBOB set.

Task. We require that we see improvement of NEA2 in comparison to the NBC-CMA in many cases, and few performance losses (over the 5 functions and 5 different dimensions. This is not a very formal criterion, but the benchmark set is a bit too small to do final decisions anyway.

Results/Visualization. Performance pictures as generated by the BBOB tools are provided in figure 3. Counting the number of improvements (over problems and dimensions, together 25) results in 11 improvements, 3 losses, and 11 cases of equal (or no) performance. The log files indicate that the number of basins identified is usually much higher for the NEA2 than for the NBC-CMA, especially in $5D$ and up. Note that small differences should not be over-interpreted as we have only 15 repeats on different instances, so a considerable amount of noise in the result can be expected.

Observations. On the Schwefel problem, not much difference between NBC-CMA and NEA2 is visbible, even on the Gallagher problems, the difference is small. However, on f23 and f24, NEA2 is clearly better than NBC-CMA (although only in small dimensions).

Discussion. The very similar behavior of NBC-CMA and NEA2 (while still being better than the BIPOP-CMA-ES) on the Gallagher problems is a bit disappointing. Obviously, a better clustering did not help to speed up search here. However, on f24 and especially f23, there is an unexpected clear difference. As these functions are highly multimodal, better clustering should be of limited importance. We are currently not able to give a good explanation for this, but it seems clear that the basin identification plays some role because in $D > 3$, especially on f23, it obviously gets unproportionally more difficult to obtain the global optimum for NEA2.

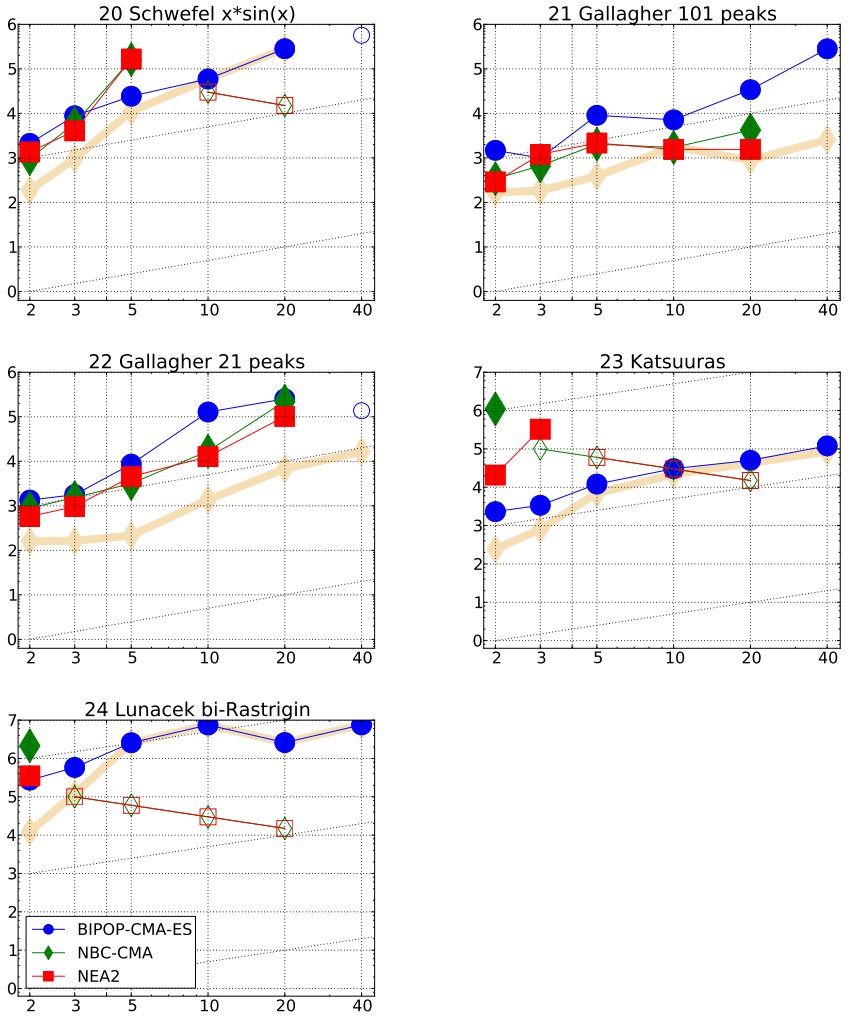

Fig. 3. Performance comparison of NEA2, NBC-CMA and BIPOP-CMA-ES on the 5 multimodal problems without strong global structure the BBOB test set provides. X-axis: problem dimension, y-axis: evaluations in log10-scale. Note that NEA2 and NBC-CMA have been allowed a maximum of 3×10^5 evaluations only.

6 Summary and Conclusions

We have shown that by adding a second heuristic rule to the nearest-better clustering algorithm (NBC), its performance in basin identification is greatly improved. However, more data on multimodal test functions with known basins or at least local optima is needed to see how large this improvement is on functions that have not been taken into account while designing the heuristic.

As found in sec. 3.1, it may also make sense to think of another accuracy measure to rate different clusterings when the true basins are known.

Concerning the comparison of the proposed NEA2 algorithm to the NBC-CMA, we can attest slight improvements, and at the same time we obtain many more clusters to start with, which could be interesting for deriving some knowledge about the treated problem quickly (e.g. its degree of multimodality). However, on the BBOB set it is currently not possible to check the accuracy of the obtained clustering easily. This deserves some further testing, also employing different problem generators which provide more support in this respect (but unfortunately much less in others as automated visualization).

References

1. Auger, A., Hansen, N.: A restart CMA evolution strategy with increasing population size. In: Proceedings of the IEEE Congress on Evolutionary Computation, CEC 2005, Edinburgh, UK, September 2-4, pp. 1769–1776. IEEE Press (2005)
2. Hansen, N.: The CMA Evolution Strategy: A Tutorial, `http://www.lri.fr/~hansen/cmatutorial.pdf` (version of June 28, 2011)
3. Martin, W.N., Lienig, J., Cohoon, J.P.: Island (migration) models: evolutionary algorithms based on punctuated equilibria. In: Handbook of Evolutionary Computation, pp. pp. C6.3:1–C6.3:16. Institute of Physics Publishing, Bristol (1997)
4. Müller, C.L., Baumgartner, B., Sbalzarini, I.F.: Particle swarm CMA evolution strategy for the optimization of multi-funnel landscapes. In: Proceedings of the Eleventh Congress on Evolutionary Computation, CEC 2009, pp. 2685–2692. IEEE Press (2009), `http://dl.acm.org/citation.cfm?id=1689599.1689956`
5. Preuss, M.: Niching the CMA-ES via nearest-better clustering. In: Proceedings of the 12th Annual Conference Companion on Genetic and Evolutionary Computation, GECCO 2010, pp. 1711–1718. ACM (2010)
6. Preuss, M., Schönemann, L., Emmerich, M.: Counteracting genetic drift and disruptive recombination in $(\mu + /, \lambda)$-EA on multimodal fitness landscapes. In: Beyer, H.G., et al. (eds.) Proc. Genetic and Evolutionary Computation Conf. (GECCO 2005), Washington D.C, vol. 1, pp. 865–872. ACM Press, New York (2005)
7. Preuss, M., Stoean, C., Stoean, R.: Niching foundations: basin identification on fixed-property generated landscapes. In: Proceedings of the 13th Annual Conference on Genetic and Evolutionary Computation, GECCO 2011, pp. 837–844. ACM (2011)
8. Shir, O.M., Emmerich, M., Bäck, T.: Adaptive Niche Radii and Niche Shapes Approaches for Niching with the CMA-ES. Evolutionary Computation 18(1), 97–126 (2010)
9. Singh, G., Deb, K.: Comparison of multi-modal optimization algorithms based on evolutionary algorithms. In: Proceedings of the 8th Annual Conference on Genetic and Evolutionary Computation, GECCO 2006, pp. 1305–1312. ACM (2006)
10. Stoean, C., Preuss, M., Stoean, R., Dumitrescu, D.: Multimodal optimization by means of a topological species conservation algorithm. IEEE Transactions on Evolutionary Computation 14(6), 842–864 (2010)