Approximating the Number of Attraction Basins of a Function by Means of Clustering and Evolutionary Algorithms

Ruxandra STOEAN[1], Mike PREUSS[2] and Cătălin STOEAN[1]

[1]Faculty of Mathematics and Computer Science,
Department of Computer Science,
University of Craiova, Romania
[2]University of Dortmund, Germany
ruxandra.stoean,catalin.stoean@inf.ucv.ro
mike.preuss@cs.uni-dortmund.de

Abstract. A novel pre-multimodal evolutionary optimization inspection of the fitness landscape for an objective function would generally be highly needed and is therefore proposed in present paper. An evolutionary population of samples in the domain of a given problem is generated and clustered by means of two appropriately chosen approaches. The aim is to direct a set of samples towards the basins of attraction within the landscape of a function and subsequently group them around some approximate optima. What results from the suggested evolutionary-clustering tool is an estimation of the number of potential solutions for the problem and consequently an informed choice for parameter setting in subpopulation demarcation within a multimodal technique that would tackle the considered objective.
Keywords: evolutionary algorithms, basin number determination, clustering
Math. Subject Classification 2000: 68T20; 91C20; 92D15

1 Introduction

Knowledge on the fitness landscape of a problem would be highly helpful when attempting to address optimization by means of evolutionary algorithms (EAs). The information on how multimodal the objective function is facilitates the setting of appropriate values for parameters that control the differentiation of individuals into subpopulations in order to be positioned in the probable basins of attraction.

Due to inherent accuracy and simplicity, niching techniques [4] generally represent the most common option for tackling possibly multimodal landscapes. However, especially within such approaches, the central piece around which separation into distinct species revolves is the radius threshold. Its assessment must be carefully undertaken, since subpopulation formation in view of tracking all existing optima heavily depends on it. An efficient means to estimate the value for the radius relies on a famous formula that needs nonetheless to identify the number of optima of the underlying objective function [1].

Analysis of the fitness landscape with the purpose to estimate the number of potential solutions for combinatorial optimization problems has already been proposed [3] and [8]. Conversely, as concerns the continuous domain and to the best of our knowledge, there has been no attempt to approximate the amount of optima in the fitness landscape. It is in this respect that we propose a straightforward hybridized evolutionary-clustering technique for the early detection of the information on the number of existing basins of attraction. The approach moreover strives to achieve a small expenditure of fitness evaluations, since, aiming to be employed at the beginning of a multimodal evolutionary technique, it must compulsory exhibit a low budget behavior.

A canonical EA [2] initially offers a directed population of individuals and clustering is subsequently applied to the obtained potential solutions. Finally, unification of adjacent clusters is performed by exploiting landscape topology. Apart from an amount of clusters that gives the expected number of optima of the problem, some rough values for the optima are outlined by the resulting prototypes.

The paper is constituted in the following structure. Section 2 puts forward the two clustering techniques selected for this experiment: The state-of-the-art Jarvis-Patrick and the more recent, effective Nearest-Better grouping. The motivation for their preference and the underlying working principles of these methods are outlined and explained. Section 3 puts forward the suggested combination between an EA and either of the two approaches, with a topological tool to detect unifiable groups at the end of the clustering process. Section 4 presents the conducted experimentation and obtained results. The final section contains the conclusions and outlook.

2 Clustering representatives

A simple EA is appointed to generate a population oriented towards the basins of the optima. This resulting collection of individuals is next subject to a clustering procedure. Two partitional high-speed clustering techniques have been chosen as the means to discover the attraction clusters in the population of potential solutions.

Jarvis-Patrick is a general clustering procedure for the determination of non-globular, compact groups, while Nearest-Better is a simple method that makes use of the fitness landscape when grouping individuals into clusters. It will thus supplementarily be investigated which of the two approaches performs better in terms of the goals of this paper.

2.1 Jarvis-Patrick clustering

The Jarvis-Patrick (JP) algorithm [5] is a non-hierarchical, non-iterative type of clustering, based on a "nearest neighbor" mechanism. The method selfdetermines the number of groups, every cluster contains at least one example and there is no overlapping. The JP approach is suitable for those problems where the goal is to identify non-globular clusters or compact clusters within large dispersed groups and when the computational runtime is important. All of the above represent a strong motivation to employ JP as a pre-multimodal EA information detector on the number and location for the basins of attraction of an objective function.

JP is laid out in high-level pseudocode in Algorithm 1, while visualization of the clustering steps can be observed in Figure 1. J nearest neighbors, in terms of (Euclidean) distance, are collected for each individual. The basic procedure takes every two pairs of samples and performs the following verification: If the two are contained in each other's neighbor list and have at least K neighbors in common, they are placed in the same cluster. A point cannot belong to more than one cluster. Moreover, if x and y meet the condition to belong to the same cluster and x and z also pass the two criteria, all three will be clustered together, indifferent of the fact of whether y and z also respect the conditions. After the formation of clusters, the prototypes are determined as the fittest individuals in each of the groups.

The drawback of the algorithm consists in the two parameters J and K that resulting clusters are very dependent upon. J indicates the number of neighbors to be examined for each considered individual. A low value for this parameter will lead to a good runtime but many small clusters, while a high number will cause fewer larger groups of samples but longer computational effort. K specifies the number of mutual neighbors and a low value will correspond to tighter clusters, while a high one results in looser groups.

The JP algorithm does not take fitness landscape into account when forming clusters, but as to counteract this disadvantage, the initial EA will have already, after some generations, focused the individuals towards the promising areas of the search space.

2.2 Nearest-Better clustering

The recently introduced Nearest-Better (NB) clustering mechanism [7] also relies on a "nearest neighbor" principle, however topological information is included in addition to location of points. For each individual, it considers the connection to one immediate neighbor, which is also better in terms of fitness. NB essentially assumes that the best individuals in different attraction basins are much further away from each other than the average distance of all individuals to their nearest better neighbors. Every individual connects to its nearest better neighbor, once more in terms of (Euclidean) distance. The longest edges – those higher than $\phi \cdot mean$(lengths of all edges) – are removed and the prototypes for each cluster are represented by those individuals that do not connect to others. Clusters are thus formed around these individuals.

The NB technique is thoroughly described in Algorithm 2. It is probably more appropriate for the current task, as it makes use of the search space

Algorithm 1 The Jarvis-Patrick algorithm applied for an evolutionary generated sample collection.

Require: A population of individuals x.
Ensure: Clusters of individuals: number, grouping, prototypes
  for $i = 1$ to pop_size **do**
    compute J nearest neighbors for $x[i]$ and store them in $N[i]$;
  end for
  for $i = 1$ to pop_size **do**
    for $j = 1$ to pop_size **do**
      if (either $x[i]$ or $x[j]$ not clustered) and ($x[i] \in N[j]$ and $x[j] \in N[i]$) and $|N[i] \bigcap N[j]| >= K$ **then**
        $x[i]$ and $x[j]$ in the same cluster, either new or one already belonging to it;
      end if
    end for
  end for
  return number of clusters, composition and fittest individual in each;

and fitness information along with an easier parametrization. This approach possesses only one additional parameter to be tuned, with 2 being a good default value [7].

Figure 2 sketches the formation of the clusters. It can be observed that the resulting groups are similar to those deriving from JP in Figure 1. Nonetheless, on a different configuration of points, as opposed to JP, NB is less reliant on distances in the search space and obviously independent of a preset vicinity.

3 Evolutionary algorithms and clustering for fitness landscape early examination

The proposed approach targets the application of each of the two clustering methods to different stages of evolved populations. It could be argued that the direct use of clustering would be sufficient, but this can only detect different clusters representing optima after somehow progressing towards good regions (as e.g. demonstrated in [6]); a random sample is hard to cluster meaningfully.

In order to move into promising areas, some optimization method has to initiate an expressive set of samples. Conversely, marching too far by means of an EA implies the danger of missing some optima on which the subpopulations go extinct.

Therefore, it is suggested to design a methodology that employs a several steps canonical EA to produce examples to be fed to either of the two clustering approaches and finally unify the resulting clusters with space topology in view.

Comparing the number of detected optima against the total number of obtained clusters during pre-experimentation lead to the insight that the two methods largely overestimate the number of attraction basins for both consid-

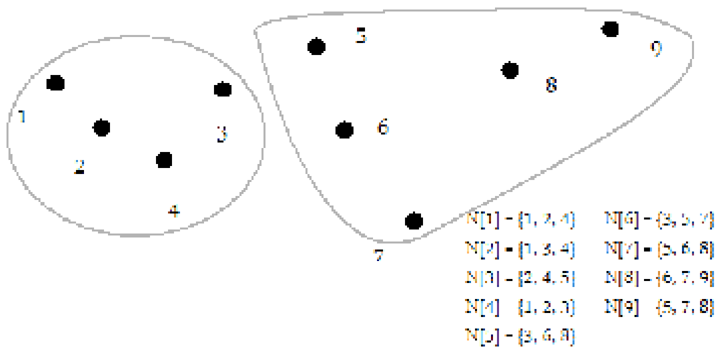

Fig. 1. Cluster formation within JP. The N matrix gives the list of neighbors for $J = 3$ and $K = 1$ and curves shape the resulting clusters.

ered test functions, with an advantage on the NB side which is less deceiving. The number of clusters was approx. 3 times higher than the amount of discovered optima. The overrating clustering action of the two techniques had to be resolved, in order to achieve the proposed goal. As a consequence, we applied the topological inspection mechanism after clusters are determined, in order to unify groups within the same basin. The fitness evaluations consumed in this final step are also counted within the totally allowed value for an efficient pre-multimodal EA information tool.

3.1 The topological cluster unification mechanism

The routine receives two cluster prototypes, checks their relative position within the fitness landscape and returns a boolean value, which specifies whether there is a valley between them or not. In the former circumstances, the conclusion is that they climb different hills, so the corresponding clusters should remain separate. In the latter case, the prototypes lie on the same hill, so the respective clusters should be merged. In order to reach a decision, a set of interior points between the two is generated. If the fitness of all these is higher than the minimal fitness of the two tested individuals, it is concluded that the clusters track the same optimum. Conversely, if there exist such a point whose fitness is smaller than the minimal fitness of the considered prototypes, then it is assessed that the two clusters follow different peaks. The mechanism is thoroughly described in Algorithm 3 [9].

A single required parameter refers to the number of interior points that should be taken taken into account between two given prototypes. In all undertaken experiments, such gradations are values taken equidistantly from the interval [0,1]. It is obvious that, the higher the number of interior points, the more precise the detection of clusters lying in the same basin of attraction or

Algorithm 2 The Nearest-Better Clustering Algorithm

Require: A population of individuals x.
Ensure: Clusters of individuals: number, grouping, prototypes.

```
for i = 1 to pop_size do
    compute distances from x[i] to all individuals
end for
for i = 1 to pop_size do
    find nearest individual that is fitter than x[i], i.e. x[j];
    if found then
        edge[i] = j;
    else
        edge[i] = 0;
    end if
end for
m = avg_{i=1}^{pop_size}(distance(x[i], x[edge[i]]));
for i = 1 to pop_size do
    if edge[i] ≠ 0 and distance(x[i], x[edge[i]]) > φ· m then
        edge[i] = 0;
    end if
end for
return the prototypes – x[i], where edge[i] = 0 – and membership to clusters – x[i] ∈ cluster[j], where edge^n[i] = prototype[j], i = 1, 2, ..., pop_size, j = 1, 2, ..., noOfClusters;
```

Algorithm 3 The topological unification mechanism for two cluster prototypes x and y

Require: Two individuals x and y.
Ensure: Whether x and y track the same optimum or two different ones.

```
i = 1;
found = FALSE;
while i < number of gradations and not found do
    for j = 1 to number of dimensions do
        interior_j = x_j + (y_j − x_j) · gradation_i;
    end for
    if f(interior) < min(f(x), f(y)) then
        found = TRUE;
    end if
end while
return found;
```

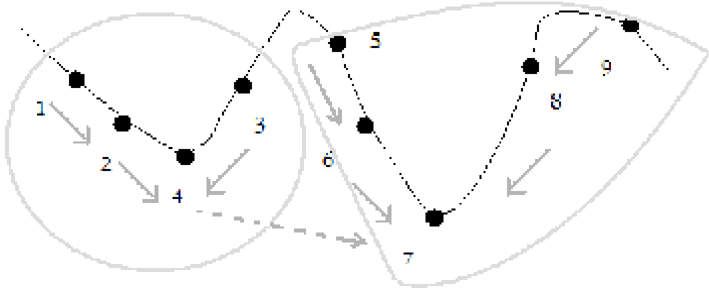

Fig. 2. Cluster formation within NB for a function minimization problem. Straight arrows point to edge creation, the dotted arrow shows that the corresponding edge is removed, as it is longer than the given threshold, and curves shape the resulting clusters.

within two different ones is. Contrarily, a small amount of considered gradations roughens the final result, but it is more economical in terms of function evaluations.

3.2 An evolutionary-clustering unified approach for an a priori optima count

A canonical EA evolves a population of individuals for a number of fitness evaluations and clustering is subsequently applied to the final generation. The estimated number of basins is given by the resulting number of clusters, while the approximate optima are given by the prototypes. The final approach is given in Algorithm 4.

Algorithm 4 EA-Clustering for the estimation of the number of basins of attraction for an objective function

Require: An objective function f to be optimized.
Ensure: Number of expected attraction basins.

```
apply a canonical EA on a population of individuals uniformly generated in the domain of f;
if choice = JP then
    apply JP to evolved population;
else
    apply NB to evolved population;
end if
apply topological unification to resulting cluster prototypes;
return number of basins of attraction and approximate peaks;
```

4 Experimental results

Experimentation aims to validate the proposed technique against a couple of functions whose number and location of optima is known and compare the performance of the two possible clustering options.

4.1 Test functions

The Waves function ($F1$) is asymmetric and some peaks are difficult to find as they lie on the border or on flat hills. The Six-Hump Camel Back function ($F2$) exhibits two local optima that are not actually much higher than their neighboring regions and thus can easily be missed (Table 1).

Table 1. Considered Benchmark Functions

Function	Optima
$F1(x,y) = (0.3x)^3 - (y^2 - 4.5y^2)xy - 4.7cos(3x - y^2(2+x))sin(2.5\Pi x))$ $-0.9 \le x \le 1.2, -1.2 \le y \le 1.2$	10
$F2(x,y) = -((4 - 2.1x^2 + \frac{x^4}{3})x^2 + xy + (-4+4y^2)y^2)$ $-1.9 \le x \le 1.9, -1.1 \le y \le 1.1$	6

4.2 Experimental setup

The same budget of fitness evaluations was used for either JP or NB, ranging from 200 to a maximum of 2000. The values for all parameters were generated using a Latin Hypersquare Design, i.e. 30 space-filling configurations were produced. The parameters of the evolutionary algorithm were generated within the following intervals for both methods: Population size is between 2 and 200, mutation and recombination probabilities between 0 and 1 and mutation strength between 0 and 5. Additionally for the JP method, the values for the two parameters J and K were both created between 1 and 25 with the constraint that $J > K$. Plus, as the number of neighbors cannot be higher than the population size, the latter is between 25 and 200. The number of interior points considered for the cluster unification method is a positive integer and is generated between 1 and 15. The Φ parameter of NB remains at the default value of 2.

4.3 Results/observations

Table 2 gives the number of optima that were detected by the two compared techniques on F1 and F2. Interest primary lies on the results obtained by the best of the 30 configurations (*Best* columns), nevertheless, in order to have an idea of a general trend, the average number of expected attraction basins and

Table 2. Attraction basins found by JP and NB in the best configuration and average over 30 configurations for $F1$ and $F2$ with different fitness evaluation budgets.

Fitness evaluation calls	**F1**				**F2**			
	JP		**NB**		**JP**		**NB**	
	Best	Average	Best	Average	Best	Average	Best	Average
200	8.13	5.15	8.36	5.95	4	3.33	4	3.37
500	8.2	4.33	8.26	5.16	4	2.62	4	2.57
1000	8.1	2.68	8.56	3.24	3.96	1.75	3.83	1.7
2000	2	1.07	2.5	1.21	1.93	1.22	1.96	1.24

found solutions were also computed. Each configuration is repeated 30 times and the average number of basins over all runs is calculated (*Average* columns).

The two potential choices for a clustering action behave quite similarly, with a slight advantage on the NB side which additionally estimated a number of clusters that is closer to the known value.

5 Conclusions and future directions

In this paper, an evolutionary-clustering hybrid approach was employed as an informative tool prior to the use of a multimodal technique on the search space of an objective function. The purpose was to obtain a rough approximation on the number of potential basins of attraction of the function which comes of immense help in the interactions of subpopulation within a multimodal EA.

The proposed technique achieved the necessary knowledge through an initial EA generation of potential candidates for a clustering procedure. Two suitable clustering methods for the given task were investigated and tests were conducted on a couple of benchmark functions for multimodal optimization.

The novel approach accomplishes the intended goal, offering a simple and even preferential means to acquire essential information on the landscape of problems to be solved by evolutionary optimization.

In the future, a hybridized version between the two clustering methodologies could lead to a more accurate estimation of the number of probable optima of an objective function. More importantly, a means to additionally approximate the size of the attraction basins would increase the ease in further subpopulation differentiation.

References

1. **Deb, K., Goldberg, D. E.**, An Investigation of Niche and Species Formation in Genetic Function Optimization, 3rd International Conference on Genetic Algorithms, pp. 42–50, Morgan Kaufman, San Mateo, California 1989
2. **Eiben A. E., Smith J. E.**, Introduction to Evolutionary Computing, Springer–Verlag, Berlin Heidelberg New York, 2003

3. **Garnier, J., Kallel, L.**, Efficiency of Local Search with Multiple Local Optima, SIAM Journal of Discrete Mathematics, vol. 15, no. 1, pp. 122–141, 2002
4. **Goldberg, D. E., Richardson, J.**, Genetic Algorithms with Sharing for Multimodal Function Optimization, 2nd International Conference on Genetic Algorithms, pp. 41–49, Lawrence Erlbaum Associates, Hillsdale, NJ, 1987
5. **Jarvis, R. A., Patrick, E. A.**, Clustering Using a Similarity Measure Based on Shared Near Neighbours, IEEE Transactions on Computers, pp. 1025-1034, 1973
6. **Parmee, I. C.**, The Maintenance of Search Diversity for Effective Design Space Decomposition Using Cluster-Oriented Genetic Algorithms (COGA) and Multi-Agent Strategies (GAANT), 2nd International Conference on Adaptive Computing in Engineering Design and Control (ACEDC '96), pp. 128–138, University of Plymouth, 1996
7. **Preuss, M., Schoenemann, L., Emmerich, M.**, Counteracting Genetic Drift and Disruptive Recombination in $(\mu \overset{+}{,} \alpha)$-EA on Multimodal Fitness Landscapes, 7th Annual Conference on Genetic and Evolutionary Computation (GECCO '05), pp. 865-872, ACM Press, 2005
8. **Reeves, C. R., Eremeev, A. V.**, Statistical Analysis of Local Search Landscapes, Journal of the Operational Research Society, vol. 55, no. 7, pp. 687–693, 2004
9. **Ursem, R., K.**, Multinational Evolutionary Algorithms, Proceedings of the Congress of Evolutionary Computation (CEC-99), vol. 3, pp. 1633–1640, 1999