Technical Report

eman ta zabal zazu

Universidad del País Vasco
Euskal Herriko Unibertsitatea

UNIVERSITY OF THE BASQUE COUNTRY
Department of Computer Science and Artificial Intelligence

Using network mesures to test evolved NK-landscapes

R. Santana, A. Mendiburu, J. A. Lozano

July 2012

San Sebastian, Spain
www.ccia-kzaa.ehu.es
hdl.handle.net/10810/4562

Using network mesures to test evolved NK-landscapes

R. Santana, A. Mendiburu, and J. A. Lozano
email:{roberto.santana,alexander.mendiburu,ja.lozano}@ehu.es
Department of Computer Science and Artificial Intelligence
University of the Basque Country

Abstract

In this paper we empirically investigate which are the structural characteristics that can help to predict the complexity of NK-landscape instances for estimation of distribution algorithms. To this end, we evolve instances that maximize the estimation of distribution algorithm complexity in terms of its success rate. Similarly, instances that minimize the algorithm complexity are evolved. We then identify network measures, computed from the structures of the NK-landscape instances, that have a statistically significant difference between the set of easy and hard instances. The features identified are consistently significant for different values of N and K.
keywords: EBNA, EDAs, NK-landscapes, network measures, problem difficulty

1 Introduction

One of the questions that have traditionally occupied researchers in the evolutionary computation (EC) community is how to characterize, and predict if possible, the difficulty that a given instance poses for an evolutionary algorithm (EA) [3, 4, 11, 21, 26]. Different fitness measures have been proposed [12, 22, 13] that quantify a variety of elements related to the behavior of the EAs.

Frequently underestimated, the question of how to generate a set of representative instances with controlled complexity is also relevant for the study of EA behavior [7]. Usually, a set of instances is randomly generated according to a number of parameters that influence their complexity, and then EAs are tested on them. Although instances generated according to a given parametrization are expected to share a number of characteristics, there is usually a wide variability among them and the behavior of the EA can be also considerably variable for instances of the same class. This is so because parametrizations may be not sufficiently fine to capture the characteristics that make the problem easy or hard.

One way to characterize instances of a given problem for which a putative structure of the interactions between its variables is known is by computing a detailed description of its underlying topology. This can be done by computing network measures that capture different salient features of the way variables interact. The structural description of a problem is not sufficient to characterize its complexity. Problems of different difficulty may share a common structure. However, in some situations an analysis of the structural description can be used as a first stage to predict the problem difficulty.

In this paper we propose an approach for the empirical analysis of problem difficulty in instances of the NK-landscape [14] problem that comprises three main steps. Firstly, to evolve the instance structures, keeping the parametrical part intact and with the aim to maximize, or minimize, the instance complexity. Secondly, to extract from each evolved instance a detailed characterization in terms of networks measures. Finally, use statistical analysis to identify which instance features have a different distribution between the set of easy and hard instances.

We apply this procedure to an original set of 9000 instances of the NK-landscape model that were proposed and studied in previous works [25, 26, 27]. The statistical analysis detected two network measures that were consistently and significantly different between easy and hard instances across different values of N and K. This type of finding would allow researchers a better characterization of NK-landscape instances and could serve as the basis for the conception of the measures of instance problem difficulty.

The rest of the paper is organized as follows. In the next section, we review the NK-landscape model and explain the main components of the approach to evolve complex NK-landscape instances. Section 3 explains the network measures used for extracting structural information from the evolved instances. Work related to our proposal is reviewed in Section 4. The experiments, the results of the statistical tests and the relevant features identified are presented in Section 5. The conclusions of the paper are discussed in Section 6.

2 Evolving complexity

2.1 NK fitness landscape model

The *NK fitness landscape model* is a parametrized model of a fitness landscape that allows to explore the way in which the neighborhood structure and the strength of interactions between neighboring variables determine the ruggedness of the landscape. For given parameters, it consists of finding the global maximum of the function.

An NK fitness landscape [14] is defined by the following components:

- Number of variables, N.
- Number of neighbors per variable, K.
- A set of K neighbors $\Pi(X_i)$ for X_i, $i \in \{1, \ldots, n\}$.
- A subfunction f_i defining a real value for each combination of values of X_i and $\Pi(Xi)$, $i \in \{1, \ldots, n\}$.

The objective function f_{NK} to maximize is defined as:

$$f_{NK}(\mathbf{x}) = \sum_{i=1}^{n} f_i(x_i, \Pi(x_i)). \tag{1}$$

The complexity of the NK fitness landscape problem depends on all its components. For $K > 1$ it is NP-complete. This problem is particularly suitable to investigate the use of network measures since it has been extensively analyzed to study EAs and other heuristic algorithms [1, 18, 25, 36, 37].

2.2 Measuring complexity

We will measure the instance complexity in terms of the success rate needed by an estimation of distribution algorithm (EDA) [15, 17, 20] to solve it. The Estimation of Bayesian networks algorithm (EBNA) [6] enhanced by a local optimization algorithm described in [25] is used for this purpose. For a given NK-landscape instance we assume that the optimum value is known and run EBNA 100 times to determine how many times the optimum is found. This number of times is the fitness $f(G)$ associated to instance G.

Pseudocode for EBNA is shown in Algorithm 1. The selection method used by our EBNA is truncation selection and the $T = 50\%$ best percentage of the population (highest objective values) is selected. Learning of the Bayesian network structure from the selected set of solutions is done using the BIC metric. Learning of the parameters is done applying maximum likelihood estimation. To sample the solutions from the network, probabilistic logic sampling is used.

Algorithm 1: **EBNA**

Generate an initial population D_0 of individuals and evaluate them
t $\leftarrow$ 1
do {
    $D^{Se}_{t-1} \leftarrow$ Select N individuals from D_{t-1} using a selection procedure
    Using D^{Se}_{t-1} as the data set, apply local search to find one BN structure that optimizes the scoring metric
    Calculate the parameters of the BN using D^{Se}_{t-1} as the data set
    $D_t \leftarrow$ Sample M individuals from the BN and evaluate them
    t $\leftarrow$ t+1
} **until** Stopping criterion is met

2.3 Instance generation

To find the optimal instances, we used a random hill climbing algorithm (RHC) that starts from a random instance and randomly modifies its neighborhood structure. The numerical values describing the function potentials are not modified. If the optimization function is improved, then the new instance is accepted, otherwise another possible modification of the neighborhood structure is proposed. The RHC pseudocode is shown in Algorithm 2 for the case where the goal is minimizing the instance complexity (i.e. maximizing the success rate of EBNA). The maximum number of evaluations allowed to RHC was set to 50.

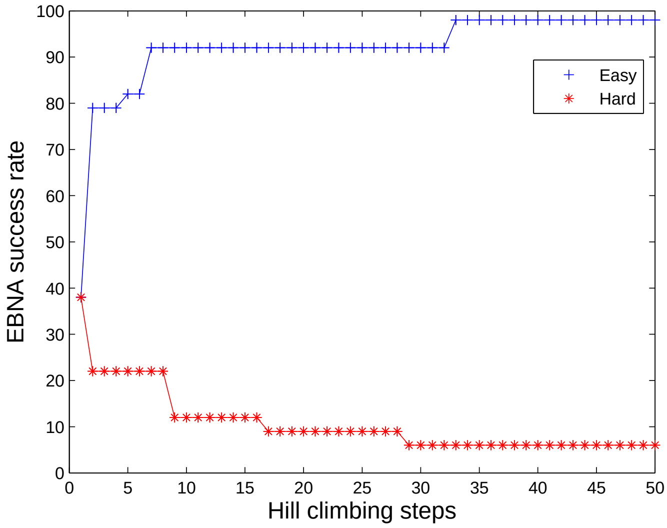


Figure 1: Evolution of the RHC algorithm during the search of easy and hard instances.

Algorithm 2: **Random Hill Climbing**

1 Run EBNA 100 times on instance G and compute the success rate of the algorithm $f(G)$
2 **do** {
3 Randomly select an arc (j, k) from instance G
4 Select a vertex l such that (j, l) is not in G
5 $G' = G$
6 Replace (j, k) by (j, l) in G'
7 Run EBNA 100 times on instance G' and compute the success rate of the algorithm $f(G')$
8 If $f(G') > f(G)$ then $G = G'$
9 } **until** Maximum number of evaluations is achieved
10 Return G.

EBNA is used in steps 1 and 7 of Algorithm 2 to evaluate the complexity of the instances but not to evolve them. Figure 1 shows two typical RHC runs in the search of easy and hard instances. Notice that both runs start from the same initial instance.

To generate our benchmark, we used an initial dataset of 9000 instances, 1000 for every possible combination of $n \in \{20, 28, 34\}$ and $K \in \{4, 5, 6\}$. All instances were solved using a branch-and-bound algorithm as described in [25]. The branch-and-bound algorithm[1] guarantees that the optimal solutions are found but since it is a complete algorithm its complexity grows exponentially fast and solving large NK instances with this algorithm rapidly becomes intractable [25]. Every time a new instance is generated using RHC we need to run the branch-and-bound algorithm to compute the new optimum of the NK fitness landscape function. This was necessary in order to identify the number

[1]We use the implementation by the author of [25], available from `http://medal.cs.umsl.edu/software.php`

of successful runs by EBNA. Starting from each of the $9,000$ instances we generated two additional instances, one easy and one hard instance. The final benchmark comprises these additional $18,000$ instances.

3 Network measures and characterization of instances

It is clear that some structural characteristics of the NK-landscape instances influence the problem difficulty. For example, for a fixed N, K determines the network density and has a direct influence on the problem complexity. Nevertheless, for fixed values of N and K there is a wide variability in the complexity of the instances. In this context, it makes sense to produce a more detailed structural characterization of the instances that may help to identify features that are good complexity descriptors. This is the approach we follow. First, we compute for each instance a large set of network measures that serve as topological descriptors. Then, we apply a statistical test to each of the features to identify those that have a significantly different distribution between easy and hard instances.

3.1 Network measures

Network measures are computed from the original directed graph of NK-instances, where the direction of the arc goes from the vertex to its neighbor. Some network measures were also computed from the associated undirected graph where directions are dropped. We extract measures from both representations to maximize the amount of extracted structural information.

Table 1 describes the topological measures extracted from the NK-landscape structures. The second column in the table gives pnemotecnic name for the measure and the third columns describes how many values where computed. Some network measures are global and a single value is computed for the network, other measures are associated to the vertices (e.g. indegree) or edges (e.g. distance between vertices). In some cases we computed the average of some of the measures defined on the edges connected to the same vertices. The computation of the number of structural and functional motifs was implemented using the brain connectivity toolbox [33]. The meaning of some of the measures shown in Table 1 is straightforward, a brief explanation of the other measures follows.

The *assortativity coefficient* is a correlation coefficient for the degree of nodes that are joined by an arc (linked nodes). The *edge betweenness centrality* is the fraction of all shortest paths in the network that traverse a given edge [2]. The *mean distance* (defined as the length of the shortest path between two vertices) between each node and the rest of vertices. Disconnected vertices are assigned a very high, unattainable, distance value. The *characteristic path length* of a graph is the average shortest path length between every pair of reachable vertices in the graph.

Node eccentricity is the maximal shortest path length between a node and any other node. *Network radius* is the minimum eccentricity and *network diameter* is the maximum eccentricity.

The *clustering coefficient* is the ratio of actually existing connections between the node neighbors and the maximal number of such possible connections. The *range* g_{ij} of an arc

Id	Property	Feature Number
1	degree	N
2	indegrees	N
3	outdegree	N
4	density und.	1
5	density dir.	1
6	assortativity und.	1
7	assortativity dir.	1
8	betweenness	N
9	mean reachability	N
10	mean distance	N
11	characteristic path length	1
12	eccentricity	N
13	radius	1
14	diameter	1
15	clustering coefficient	N
16	shortcuts prob.	$N \cdot (N-1)$
17	range vertex	$N \cdot (N-1)$
18	mean edge range	1
19	fraction shortcuts	1
20	mean motif number $Z=3$	13
21	vertex motif number $Z=3$	$13N$
22	Newmann modularity	1
23	node part. coefficient	N

Table 1: Topological measures extracted from the NK-landscape structures.

e_{ij} [38] is the length of the shortest path from j to i after arc e_{ij} has been removed from the graph. *Shortcuts* are arcs which significantly reduce the characteristic path length. If $g_{ij} > 2$, then the arc forms a shortcut from j to i. A *module* is generally associated to a densely connected subset of nodes that is only sparsely linked to the remaining network. Score functions are defined to measure the *degree of network modularity*. In this paper we use the Newmann's spectral algorithm [16] for community detection in large networks. *Node participation coefficient*: The participation coefficient [8] defines how well distributed the links of a node are between different modules. It is close to 1 if the links are uniformly distributed among the modules and 0 if all the links fall within one module. The same modules used to compute the maximum modularity value have been employed to compute the node participation coefficient.

A *structural motif of size* Z [19, 34] is a connected graph with Z vertices. For each Z there is a limited set of distinct structural motifs which are called motif classes.

A *motif frequency spectrum* records the number of occurrences of each motif of a given class for a size Z. To compute the motif frequency, the network is inspected and all occurrences of the given motif are counted. *Motif number* is the total number of all motifs of all classes (for a given size Z) encountered in a network. The motif number is obtained as the sum over the motif frequency spectrum. One can view motif analysis as a kind of generalization of the clustering coefficient [29]. Figure 2 shows all structural

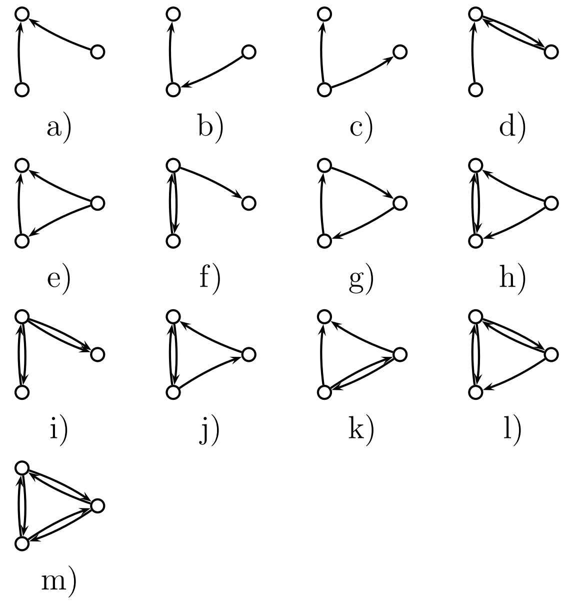


Figure 2: All structural motifs for $Z = 3$.

motifs for a motif class of size $Z = 3$. For computational reasons, we constrained our analysis to motifs of $Z = 4$.

4 Related work

Before presenting the experimental results we discuss some related work on the analysis of the NK-landscape model and the use of network measures to investigate different issues related to EAs' behavior.

Recently [23, 35, 37], network measures have been used to analyze an inherent network describing the fitness landscape of the NK-model for small instances. The inherent network is the graph where the vertices are all the local maxima and edges mean basin adjacency between two maxima. Edges were weighted according to the probability to move between basins of attractions. The networks were exhaustively extracted for a number of representative instances and some network descriptors were computed. In [37], the clustering coefficient, the disparity measure and the shortest path were considered. An interesting finding presented in [37] is that the clustering coefficient decreases with the degree of epistasis K while, for a fixed K, it tends to increase with increasing locality. One important difference with the work presented in this paper is that we consider as network the original structure of the problem. Therefore, in our approach each network serves to characterize the instance and not its fitness landscape. Another relevant difference with the work presented in [23, 35, 37] is that we consider in the analysis a more extensive set of network descriptors.

In a number of works [10, 28, 26, 27] different issues related to the behavior of EDAs and GAs for NK-landscapes have been analyzed. In [26], random instances of the NK-landscape are used to evaluate the behavior of the hierarchical BOA [24] enhanced by a local optimization algorithm. As problem difficulty measures the fitness distance correlation, the correlation coefficient, the distance of local and global optima, and the escape rate are used. All these distances, that in some cases do not provide a clear indication of what problem instances are difficult and what instances are easy [26, 27], require the computation of the fitness values, at least for a number of points, i.e. they are not structural characteristics of the problem.

In [5, 4], authors investigate the impact that different network topological characteristics have, both in the hardness of the problem and in the performance of different EDAs. The selected functions are defined in network topologies such as grids, small-world networks and random graphs. The clustering coefficient and the characteristic path length are used in order to quantify the topological properties of the function structure and analyze their relation with the behavior of EDAs. Network measures are also used in [32, 31] in the context of extracting structural information from probabilistic models learned during the evolution. This type of information can be used to characterize the complexity of the instances that are being optimized but its accuracy is influenced by other factors like the population size, selection method, etc.

5 Experiments

The objective of the experiments is to determine whether the network measures extracted from the evolved instances capture the differences between the sets of easy and hard instances.

In order to identify the set of significant features, we applied, for each feature, a statistical test to determine whether there exists significant difference between the easy and hard instances for the given feature. The statistical test of choice was the Wilcoxon rank sum test of equal medians and the parameter $\alpha = 0.05$ was fixed for all the statistical tests. The test outputs the p-value corresponding to the statistics and we use these values to further characterize the differences between the features.

All experiments were computed in a cluster of over 240 cores and we use C++ implementations of the NK fitness model [25], EBNA [6], and of the brain connectivity toolbox [30].

5.1 Numerical results for EBNA

Table 2 shows, for each of the network measures described in Table 1 whether it was identified as significant for any combination of N and K. When groups of features were considered, the table shows how many of the variables in the group were detected as significant. It can be seen in Table 2 that out of all possible statistical tests only statistical differences between the groups of easy and hard instances are found only 26 times for 8 network measures. There are two coincidences for the clustering coefficient (row number 15) and the node participation coefficient (row number 23) respectively but in every case, only one test of N (there is one coefficient for every node in the network) found significant differences. Therefore 2 tests out of $9(20 + 28 + 34) = 738$ tests might be due to multiple testing.

A different scenario arises for the motif number (row number 20) for which two motif classes were identified as relevant for all combinations of N and K. These row comprises

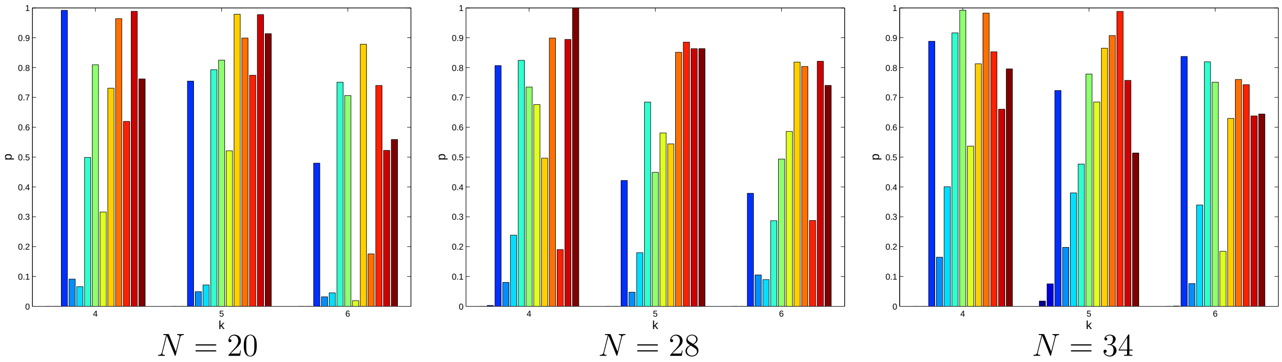


Figure 3: P-values obtained from the application of the statistical test for the mean motif number.

13 different features, one for each of the motifs shown in Figure 2. To compute this feature, the motif frequency spectrum of every node is computed and then the mean is computed. Of the 13 motifs, there were two motifs classes that were respectively identified as significant 8 and 9 times. These two motifs, whose frequencies significantly change between easy and hard instances, correspond to motifs a) and b) in Figure 2.

Figure 3 shows the p-values obtained from the application of the statistical test to the 13 motif features for all combinations of K and N. The p-values for these two motifs are so small that they can only be appreciated for $N = 34$, $k = 5$. Also p-values corresponding to motifs d) and e) in Figure 2 are very small but they do not satisfy the threshold to be considered significant. Interestingly, motif d) is the result of adding an arc to motif a). Similarly, motif e) can be obtained by adding an arc to motif a) or motif b).

In the next step we investigate the sign of the differences between the means for all the motifs features. We focus on motifs a), b), c) and d). The mean frequencies corresponding to the easy instances are subtracted from the means frequencies computed for the hard instances. These results are shown in Figure 4. The results clearly indicate that there is an increase in the frequencies of these motifs in the hard instances. We have not been able to find a causal relationship between the increase in complexity for EBNA and the unequal distribution of these motifs.

5.2 Numerical results for GA

In this section we present preliminary results for a similar analysis done for a simple GA. We applied the same experimental protocol but instead of using EBNA to compute the fitness of the instances in Algorithm 2, we used a simple GA as described in [25]. The GA uses binary tournament selection, two-point crossover, and bit-flip mutation. New candidate solutions are incorporated into the original population using restricted tournament replacement [9]. Here we present results for $N = 20$ and $N = 34$.

The experimental results are very similar to those obtained using EBNA although there are some differences. The only network measures for which statistical differences are found in 5 of the 6 comparison were motifs a) and b). The p-values computed for all motifs are shown in Figures 5 and 6 for $N = 20$ and $N = 28$, respectively. It can be seen in Figure 5 that for $K = 4$ many other motifs were below the 0.05 significance

	NK								
n	20			28			34		
Id/k	4	5	6	4	5	6	4	5	6
1									
2									
3									
4									
5									
6									
7									
8									
9							1		
10									
11			1						
12									
13									
14									
15		1						1	
16			1						
17									
18									
19									
20	2	2	2	2	2	2	2	1	2
21					1				
22		1							
23		1				1			

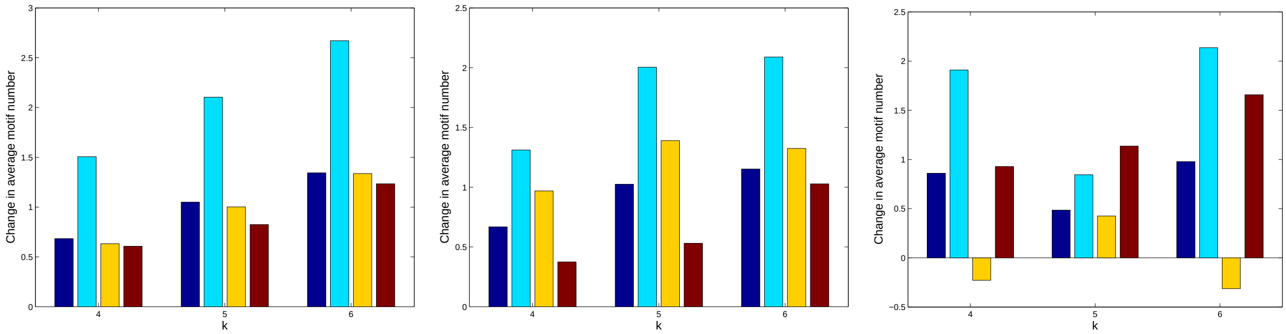
Table 2: Relevant features identified by the application of the statistical test.

threshold. On the other hand, for $K = 6$ none of the motifs was found to be significantly different. More extensive experimentation is needed to try to find an explanation to this variability.

6 Conclusions

In this paper we have introduced an empirical method for investigating some factors that could predict differences in the complexity of NK-landscape instances for EAs. Our method is based on the direct evolution of easy and hard instances using the success rate of an evolutionary algorithm to estimate the instance complexity. Although the evolved instances are not guaranteed to be easy or hard, the evolutionary process guarantees that the evolved instance will be "easier" or "harder" than the initial instance. Therefore, if for each random instance in the dataset we apply evolution in the two directions of difficulty we can guarantee that the two final sets will differ in terms of complexity with respect to the original set, and more significantly, between them.

We have focused on the analysis of EBNA, an advanced EDA that learns Bayesian



$N = 20$ $N = 28$ $N = 34$

Figure 4: Difference in the average motif frequencies between easy and hard instances (hard-easy) for a) $n = 20$, b) $n = 28$, and $n = 34$. Information about only four motifs is shown.

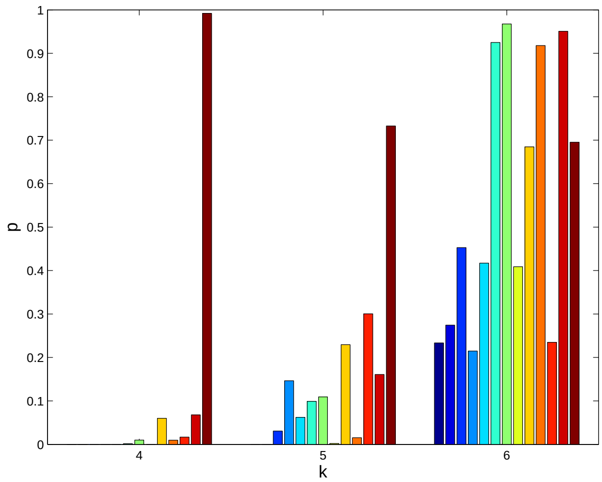


Figure 5: Easy and hard instances for the GA. P-values obtained from the application of the statistical test for the mean motif number.

networks but we have also presented preliminary results for a simple GA. These preliminary results show the existence of similarities between the detected factors that have been identified to have a different distribution between easy and hard distances. Our approach can be extended to investigate other EAs and other optimization algorithms in general.

Concerning the application of these results to predict the instance difficulty, we do not expect that a single structural feature could be used with a high accuracy to predict the complexity of an instance. One reason is that the structural description of the problem does not capture all the complexity aspects. However, a more detailed instance characterization, joining structural descriptions in the form of several network measures to fitness-difficulty measures as those traditionally applied in EAs [12, 22, 13] could improve the prediction capability of current methods. By investigating the correlation between some topological characteristics (e.g. motif frequencies) and measures of difficulty defined for EAs, we could determine which patterns of interactions are likely to pose a challenge for EAs. Furthermore, topological measures could serve to compare different classes of

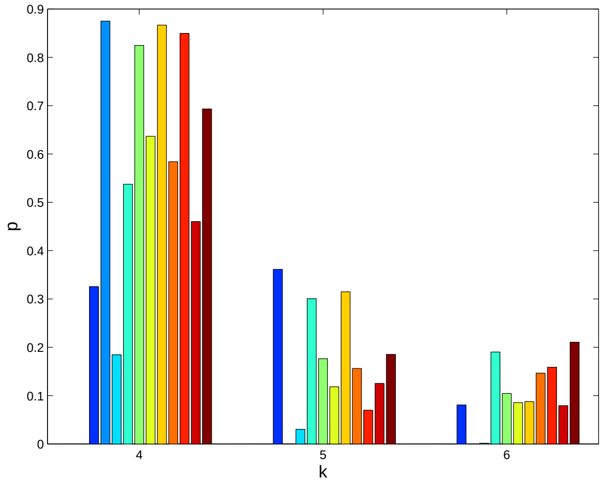


Figure 6: Easy and hard instances for the GA. P-values obtained from the application of the statistical test for the mean motif number.

problems, and not only instances, in terms of their structural similarity. This would in turn contribute to facilitate the transference of information and solution strategies between different problem classes.

References

[1] H. Aguirre and K. Tanaka. Genetic algorithms on NK-landscapes: Effects of selection, drift, mutation, and recombination. *Applications of Evolutionary Computing*, pages 131–142, 2003.

[2] U. Brandes. A faster algorithm for betweenness centrality. *Journal of Mathematical Sociology*, 25:163–177, 2001.

[3] J. M. Daida, J. A. Polito, S. A. Stanhope, R. R. Bertram, J. C. Khoo, and S. A. Chaudhary. What makes a problem gp-hard? analysis of a tunably difficult problem in genetic programming. In *Proceedings of the Genetic and Evolutionary Computation Conference CEC-1999*, volume 2, pages 982–989, Orlando, Florida, USA, 13-17 July 1999. Morgan Kaufmann.

[4] C. Echegoyen. *Contributions to the Analysis and Understanding of Estimation of Distribution Algorithms*. PhD thesis, Department of Computer Science and Artificial Intelligence. University of the Basque Country, 2012.

[5] C. Echegoyen, A. Mendiburu, R. Santana, and J. A. Lozano. Estimation of Bayesian networks algorithms in a class of complex networks. In *Proceedings of the 2010 Congress on Evolutionary Computation CEC-2010*, Barcelone, Spain, 2010. IEEE Press.

[6] R. Etxeberria and P. Larrañaga. Global optimization using Bayesian networks. In A. Ochoa, M. R. Soto, and R. Santana, editors, *Proceedings of the Second Symposium*

on Artificial Intelligence (CIMAF-99), pages 332–339. Editorial Academia. Havana, Cuba, 1999.

[7] M. Gallagher and B. Yuan. A general-purpose tunable landscape generator. *Evolutionary Computation, IEEE Transactions on*, 10(5):590–603, 2006.

[8] R. Guimera and L. A. N. Amaral. Functional cartography of complex metabolic networks. *Nature*, 433:895–900, 2005.

[9] G. R. Harick. Finding multimodal solutions using restricted tournament selection. In *Proceedings of the International Conference on Genetic Algorithms (ICGA-95)*, pages 24–31. Morgan Kaufmann, 1997.

[10] M. Hauschild and M. Pelikan. Performance of network crossover on NK landscapes and spin glasses. *Parallel Problem Solving from Nature–PPSN XI*, pages 462–471, 2011.

[11] J. Horn. *Genetic Algorithms, Problem Difficulty, and the Modality of Fitness Landscapes*. PhD thesis, University of Illinois at Urbana-Champaign, Urbana, IL, July 1995.

[12] T. C. Jones and S. Forrest. Fitness distance correlation as a measure of problem difficulty for genetic algorithms. In L.J.Eshelman, editor, *Proceedings of the Sixth International Conference on Genetic Algorithms*, 1995.

[13] L. Kallel, B. Naudts, and R. Reeves. Properties of fitness functions and search landscapes. In L. Kallel, B. Naudts, and A. Rogers, editors, *Theoretical Aspects of Evolutionary Computing*, pages 177–208. Springer, 2000.

[14] S. Kauffman. *Origins of Order*. Oxford University Press, 1993.

[15] P. Larrañaga and J. A. Lozano, editors. *Estimation of Distribution Algorithms. A New Tool for Evolutionary Computation*. Kluwer Academic Publishers, Boston/Dordrecht/London, 2002.

[16] E. A. Leicht and M. E. J. Newman. Community structure in directed networks. *Physical Review Letters*, 100:118703, 2008.

[17] J. A. Lozano, R. Sagarna, and P. Larrañaga. Parallel estimation of distribution algorithms. In P. Larrañaga and J. A. Lozano, editors, *Estimation of Distribution Algorithms. A New Tool for Evolutionary Computation*, pages 125–142. Kluwer Academic Publishers, Boston/Dordrecht/London, 2002.

[18] B. Manderick, M. de Weger, and P. Spiessens. The genetic algorithm and the structure of the fitness landscape. In *Proceedings of the fourth international conference on genetic algorithms*, pages 143–150. San Mateo, CA: Morgan Kauffman, 1991.

[19] R. Milo, S. Shen-Orr, S. Itzkovitz, N. Kashtan, D. Chklovskii, and U. Alon. Network motifs: Simple building blocks of complex networks. *Science*, 298:824–827, 2002.

[20] H. Mühlenbein and G. Paaß. From recombination of genes to the estimation of distributions I. Binary parameters. In H.-M. Voigt, W. Ebeling, I. Rechenberg, and H.-P. Schwefel, editors, *Parallel Problem Solving from Nature - PPSN IV*, volume 1141 of *Lectures Notes in Computer Science*, pages 178–187, Berlin, 1996. Springer.

[21] B. Naudts and J.Naudts. Fitness landscapes and problem difficulty: the effect of spin-flip symmetry on the performance of the simple ga. In *Parallel Problem Solving from Nature - PPSN V International Conference*, pages 67–76, Amsterdam, The Netherlands, 1998. Springer. Lecture Notes in Computer Science 1498.

[22] B. Naudts and L. Kallel. A comparison of predictive measures of problem difficulty in evolutionary algorithms. *IEEE Transactions on Evolutionary Computation*, 4(1):1–15, 2000.

[23] G. Ochoa, M. Tomassini, S. Vérel, and C. Darabos. A study of NK landscapes' basins and local optima networks. In *Proceedings of the 10th annual conference on Genetic and evolutionary computation (GECCO-2008)*, pages 555–562. ACM, 2008.

[24] M. Pelikan. *Hierarchical Bayesian Optimization Algorithm. Toward a New Generation of Evolutionary Algorithms*, volume 170 of *Studies in Fuzziness and Soft Computing*. Springer, 2005.

[25] M. Pelikan. Analysis of estimation of distribution algorithms and genetic algorithms on NK landscapes. In *Proceedings of the Genetic and Evolutionary Computation Conference GECCO-2009*, pages 1033–1040. ACM, 2008.

[26] M. Pelikan. NK landscapes, problem difficulty, and hybrid evolutionary algorithms. In *Proceedings of the 12th annual conference on Genetic and evolutionary computation*, pages 665–672. ACM, 2010.

[27] M. Pelikan. Analysis of epistasis correlation on NK landscapes with nearest neighbor interactions. In *Proceedings of the 13th annual conference on Genetic and evolutionary computation*, pages 1013–1020. ACM, 2011.

[28] M. Pelikan, K. Martin, D. E. Goldberg, M. V. Butz, and M. Hauschild. Performance of evolutionary algorithms on NK landscapes with nearest neighbor interactions and tunable overlap. In *Proceedings of the Genetic and Evolutionary Computation Conference GECCO-2009*, pages 851–858, New York, NY, USA, 2009. ACM.

[29] J. C. Reijneveld, S. C. Ponten, H. W. Berendse, and C. J. Stam. The application of graph theoretical analysis to complex networks in the brain. *Clinical Neurophysiology*, 118(11):2317–2331, 2007.

[30] M. Rubinov, S. A. Knock, C. J. Stam, S. Micheloyannis, A. W. F. Harris, L. M. Williams, and M. Breakspear. Complex network measures of brain connectivity: Uses and interpretations. *Neuroimage*, 52(3):1059–1069, 2010.

[31] R. Santana, R. Armañanzas, C. Bielza, and P. Larrañaga. Network measures for information extraction and transfer learning in evolutionary algorithms. *IEEE Transactions on Evolutionary Computation*, 2012. Submmitted to publication.

[32] R. Santana, C. Bielza, and P. Larrañaga. Network measures for re-using problem information in EDAs. Technical Report UPM-FI/DIA/2010-3, Department of Artificial Intelligence, Faculty of Informatics, Technical University of Madrid, June 2010.

[33] O. Sporns. *Neuroscience Databases. A Practical Guide*, chapter Graph theory methods for the analysis of neural connectivity patterns, pages 171–186. Kluwer, 2002.

[34] O. Sporns and R. Kötter. Motifs in brain networks. *PLoS Biology*, 2(11):e369, 2004.

[35] M. Tomassini, S. Vérel, and G. Ochoa. Complex-network analysis of combinatorial spaces: The NK landscape case. *Physical Review E*, 78(6):066114, 2008.

[36] S. Verel, P. Collard, and M. Clergue. Where are bottlenecks in NK fitness landscapes? In R. Sarker, R. Reynolds, H. Abbass, K. C. Tan, B. McKay, D. Essam, and T. Gedeon, editors, *Proceedings of the 2003 Congress on Evolutionary Computation CEC-2003*, volume 1, pages 273–280. IEEE, 2003.

[37] S. Verel, G. Ochoa, and M. Tomassini. Local optima networks of NK landscapes with neutrality. *Evolutionary Computation, IEEE Transactions on*, 15(6):783–797, 2011.

[38] D. J. Watts. *Small Worlds: The Dynamics of Networks between Order and Randomness.* Princeton University Press, 1999.