On the Selection of Fitness Landscape Analysis Metrics for Continuous Optimization Problems

Yuan Sun, Saman K. Halgamuge
Department of Mechanical Engineering
The University of Melbourne
Parkville, Victoria, 3010, Australia
Email: yuans2@student.unimelb.edu.au

Michael Kirley
Department of Computing and Information System
The University of Melbourne
Parkville, Victoria, 3010, Australia

Mario A. Muñoz
School of Mathematical Sciences
Monash University
Clayton, Victoria, 3800, Australia

***Abstract*—Selecting the best algorithm for a given optimization problem is non-trivial due to large number of existing algorithms and high complexity of problems. A possible way to tackle this challenge is to attempt to understand the problem complexity. Fitness Landscape Analysis (FLA) metrics are widely used techniques to extract characteristics from problems. Based on the extracted characteristics, machine learning methods are employed to select the optimal algorithm for a given problem. Therefore, the accuracy of the algorithm selection framework heavily relies on the choice of FLA metrics. Although researchers have paid great attention to designing FLA metrics to quantify the problem characteristics, there is still no agreement on which combination of FLA metrics should be employed. In this paper, we present some well-performed FLA metrics, discuss their contributions and limitations in detail, and map each FLA metric to the captured problem characteristics. Moreover, computational complexity of each FLA metric is carefully analysed. We propose two criteria to follow when selecting FLA metrics. We hope our work can help researchers identify the best combination of FLA metrics.**

***Keywords*—Continuous optimization problem, problem characteristics, fitness landscape analysis, problem difficulty**

I. Introduction

Algorithm Selection Problem (ASP), first defined by Rice [1], is to identify the most efficient algorithm for a given computational task. Due to large number of algorithms and high complexity of problems, algorithm selection is non-trivial. A possible way to tackle this challenge is to attempt to understand the problem complexity. Fitness landscape analysis (FLA) is a widely used technique to investigate the characteristics of problems.

The original idea of fitness landscape dates back to Wright in 1932 [2] while a widely accepted formal definition is given by Stadler [3], which consists of three parts:

- a solution space $\mathbf{X}$,
- a definition of distance d,
- a fitness function f: $\mathbf{X} \rightarrow \mathbf{R}$.

For combinatorial optimization problems, candidate solutions are encoded before calculating the distance between them. There are many definition of distances, each of which results in a different fitness landscape. Therefore, the fitness landscapes can be regarded as a set of two functions f and d [4]:

$$\mathbf{F} = (\mathbf{X}, f, d). \tag{1}$$

The aim of a global optimization problem is to find the optimal solution in the search space. Without loss of generality, we assume minimization problems in this paper. When both the solution space and the fitness value space are real numbers, it becomes a continuous optimization problem. The definition of continuous optimization problems is

$$x = \arg\min f(x) \tag{2}$$

where x and $f(x)$ are real numbers. For continuous optimization problems, euclidean distance is often employed as the distance metric. In that case, the fitness landscapes can be regarded as a function of f.

The main difference between continuous and combinatorial optimization problems is that continuous optimization problems have an infinite number of candidate solutions, while combinatorial optimization problems have a finite number of candidate solutions. Therefore, the search space of continuous optimization problems is much larger than that of combinatorial optimization problems, which makes continuous optimization problems much harder to solve.

In the last twenty years, a number of FLA metrics have been proposed (see survey papers in FLA [4] [5] [6]). However, there is little research on the selection of FLA metrics. In this paper, we summarize important problem characteristics and investigate how these characteristics contribute to problem difficulty in Section II. In Section III, a number of well-performed FLA metrics for continuous optimization problems are discussed. We show in detail how these metrics work, discuss their contributions and limitations, and map FLA metrics to problem characteristics they captured. Moreover, the computational complexity of each FLA metric is analysed. We propose two criteria when selecting the best combination of FLA metrics. In Section IV, we present three benchmark suites, and discuss how these benchmark functions can be employed to compare the performances of FLA metrics. Finally, we conclude the paper and show future directions.

II. Problem Characteristics

What makes a problem difficulty to solve? To answer this question, we summarize in this section the problem characteristics which are related to problem difficulty, and explain how these characteristics contribute to problem difficulty.

978-1-4799-4598-6/14/$31.00 ©2014 IEEE

A. Modality

A local optimum is a solution where its fitness value is smaller than or equal to all its neighborhoods': $f(x^*) \leq f(x)$, where $x \in N(x^*)$ and $N(x)$ is the neighborhood of x^*. When the fitness value of an optimum is less than or equal to the fitness values of all the solutions in the search space, it becomes a global optimum: $f(x^*) \leq f(x)$, where $x \in \mathbf{X}$ [7]. Functions can be classified as unimodal or multi-modal according to the number of optima in the landscapes.

Intuitively, search algorithms may get stuck into local optima, for there is no information guiding the search out of local optima. Therefore, the number and distribution of local optima are important landscape features. Garnier and Kallel proposed a sampling and probability method to estimate the number and distribution of local optima [8], based on the assumption that each candidate solution in the search space belongs to a basin of an optimum, which is not always the case in real world problems.

Some researches argued that multi-modal problems are less difficult than unimodal problems in some cases [9]. This may due to the fact that there are other characteristics, for example, the size and shape of basins [10], that are also related to problem difficulty. We should take into consideration all the features when discussing problem difficulty.

B. Basins of Attraction

Basins of attraction refer to the search regions leading directly to the local optima during a local search. It is defined as $\mathbf{B}(x) = \{x : \mu(x) = x_o\}$, where x are candidate solutions, x_o is a local optimum, and μ represents a local search. An optimum is more likely to be found by a local search algorithm if it has a larger basin of attraction. The number of basins of attraction as well as the size and the shape of the basins of attraction are important landscape features [10]. Actually, the number of basins of attraction is equal to the number of local optima, and the size of basins of attraction is related to ruggedness as well as smoothness, in the way that larger basins of attraction tends to have smoother landscapes.

C. Ruggedness, Smoothness and Neutrality

Landscapes can be classified into three categories according to the fitness changes in neighborhoods, namely rugged, smooth and neutral landscapes. Landscapes with high fitness changes in neighborhoods are called rugged landscape, with small fitness changes in neighborhoods called smooth landscapes, and with equal or nearly equal fitness values in the neighborhoods called neutral landscapes. Ruggedness and smoothness for a particular problem may range widely for different encoding strategies and different distance measures, while neutrality is independent with them.

Ruggedness and smoothness are related to the modality and basins of attraction. High rugged landscapes tend to have high modality, while smooth landscapes are likely to have large basins of attraction. Intuitively, rugged landscapes are more difficult than smooth landscapes for a local search to locate at the global optimum. In the landscapes with high neutrality, the performances of all algorithms are approximately the same with random search, for there is no guiding information to navigate the search. That may be the reason why adding neutrality to deceptive landscapes decreases problem difficulty, while adding neutrality to straightforward landscapes may increase problem difficulty [11].

D. Dimensionality

Dimensionality refers to the number of decision variables in the search space. As the dimensionality grows, the search space increases intensively. The optimization problem is called large-scale when the number of dimensions is significantly large. The number of decision variables is a factor to problem difficulty [12]. The performances of evolutionary algorithms deteriorate as the dimensionality increases [13], which is due to the curse of dimensionality. The distance metrics become less meaningful when dimensionality increases significantly. Cooperative coevolutionary algorithms can solve large-scale optimization problems efficiently by decomposing a problem into sub-problems and dealing with them separably [13], [14].

E. Separability

Separability is related to epistasis [15]. A problem is separable if there is no interactions between decision variables, otherwise, it is non-separable. Therefore, a separable problem can be divided into d subproblems, where d is the number of decision variables. The mathematical definition of separable problems is given by: a function f is separable if

$$\arg \min_{(x_1,\ldots,x_d)} f(x) = (\arg \min_{x_1} f(x), \ldots, \arg \min_{x_d} f(x)). \quad (3)$$

It is computationally less intensive for a cooperative coevolution algorithm to solve separable problems comparing with non-separable problems [16]. Covariance Matrix Adaptation - Evolutionary Strategy (CMA-ES) performs well on non-separable problems, for it takes variable interaction into consideration [17].

F. Evolvability and Deception

Evolvability refers to the ability of a candidate solution to find a better solution during the search. Deception refers to the information that guides the search away from the global optimum. Evolvability and deception are algorithm dependent. A landscape with high evolvability or deception for a given algorithm is not necessary deceptive to other algorithms. Evolvability and deceptions technically are not landscape features. They reflect the performances of algorithms on landscapes.

III. Fitness Landscape Analysis Metrics

FLA is a widely used technique to extract characteristics from problems. The choice of FLA metrics is of vital importance to the accuracy of ASP. Ideally, the selected FLA metrics should capture all the important problem characteristics. Computational complexity is another consideration when selecting FLA metrics, for the computational cost of calculating these metrics should be less than 10 percent of the total computational cost of solving the problems [18]. In the last twenty years, a number of FLA metrics have been proposed, most of which were first proposed for combinatorial optimization problems, and some of them were modified for continuous optimization problems later on. In this section, we discuss some well performed FLA metrics for continuous optimization problems in detail, point out the problem characteristics they capture and analyse their computational complexity.

A. Fitness Distance Correlation

Fitness Distance Correlation (FDC) is a well known and widely used metric [19], [20], which was first proposed by Jones et al. [21] to study GA performance:

$$FDC = \frac{C_{fd}}{S_f S_d} \tag{4}$$

where $C_{fd} = \frac{1}{n}\sum_{i=1}^{n}(f_i - \tilde{f})(d_i - \tilde{d})$, s_f s_d, $\tilde{f}$, $\tilde{d}$ are standard deviations and means of f, d, and f is the fitness value, d is the distance between the candidate solution and the global optimum. In fact, FDC is the correlation between f and d. If $FDC > 0$, the fitness value decreases as the algorithm approaches the global minimum. On the other hand, if $FDC < 0$, the fitness value increases when the algorithm approaches the global minimum, which misleads the search away from the global minimum. Therefore, FDC perfectly reflects the global structure, evolvability and deception. However, the implementation of FDC requires the knowledge of the global optimum, which makes FDC less useful. Moreover, it has been shown that different fitness landscapes have very similar fitness distance correlation [22], which may due to the fact of biased samples. The computational complexity of FDC is $O(n)$, where n is the sample size.

B. The Distribution of Fitness Values

The distribution of fitness values, $p(y)$, first proposed by Rose et al. [23], is the probability density function of fitness values. It can be used to approximate the optimal solution by letting $p(y) = 0$. Rose et al. pointed out that the distribution of fitness values can be used as a classification method, in the way that a problem with high decay $p(y)$ should be hard to solve, for that the probability of finding a better solution decays very fast as well [23]. In our opinion, it may be true for random search, however, for most algorithms, it is not the case. The problems with fast decay fitness distribution only means that the number of high fitness values are small, but not necessarily hard to find.

The skewness, kurtosis and the number of peaks are three important properties of distributions employed to characterize problems [24]. The kurtosis reflects the sharpness of the distribution of fitness values, therefore, it may refers to ruggedness, smoothness and neutrality. The skewness refers to the symmetry of the distribution of fitness values, and the number of peaks refers to modality of landscapes. The computational complexity of skewness and kurtosis are both $O(n)$, while the complexity of number of peaks is $O(mn)$, where m is the number of segments y divided when approximating the distribution. The main limitation of the distribution metrics is that it does not capture the locations of candidate solutions.

C. Length Scale

Length Scale (LS) measures the ratio of changes in the fitness values to steps between candidate solutions in the search space [25]:

$$LS = \frac{|f(x_1) - f(x_2)|}{\|x_1 - x_2\|} \tag{5}$$

where x_1, x_2 are candidate solutions in search space, and $f(x_1)$, $f(x_2)$ are corresponding fitness values. In fact, LS is approximately equal to the derivative of fitness function when x_1 and x_2 are very close to each other: $\lim_{\delta x \to 0} r = f'(x)$. Therefore, LS reflects ruggedness, smoothness and neutrality. If the object is to calculate LS of each pair of candidate solutions, the computational complexity is $O(n^2)$.

In fact, the distribution of fitness values can be calculated in use of the derivative of fitness functions as long as the fitness function is continuous:

$$p(y) = \sum_{x_i} \frac{p(x)}{f'(x)}, f'(x) \neq 0 \tag{6}$$

where $p(x)$ is the probability density function of candidate solutions, and $f'(x)$ is the derivative. It maybe an indication of relationships between the distribution of fitness values and length scale. Instead of studying the ratio between δy and δx, we can investigate the conditional probability $p(\delta y/\delta x)$, which may be a good FLA metric.

D. Fitness Cloud and Negative Slope Coefficient

Fitness Cloud (FC) is the graph of fitness values of parents verses fitness values of offspring for an evolutionary algorithm [26]. It can be used to visualize the search space and characterize the set of local optima [27]. In the region under the line: $y = x$, the fitness value of offspring is smaller than the fitness value of parent, which means the fitness value is improved. Therefore, the graph with large region under $y = x$ reflects an easy landscape. The computational complexity of FC is $O(qn)$, where q is the number of offspring of each parent.

Negative Slope Coefficient (NSC) is a quantified FLA metric based on FC [28]. Firstly, partition x axis of FC into m segments. Secondly, calculate the average fitness values of parents in each partition, denoted by M_i, and the corresponding average fitness values of offspring, denoted by N_i, where $1 \leq i \leq m$. Thirdly, calculate the slopes between two sequential partitions, denoted by k_i, where $1 \leq i \leq m-1$:

$$k_i = \frac{N_{i+1} - N_i}{M_{i+1} - M_i}. \tag{7}$$

Finally, calculate NSC:

$$NSC = \sum_{i=1}^{m-1} \min(p_i, 0). \tag{8}$$

Actually, NSC is the summation of negative slopes in the partitioned FC. A negative slope means that an improvement in the fitness values of parents will result in worse fitness values of offspring, which makes the problem misleading. Therefore, NSC and FC refers to evolvability and deception of landscapes. The computational complexity of NSC is $O(\max\{qn, mn\})$. The main limitation of NSC is that it can not be used to compare different problems, for NSC is not normalized to a given range. Besides, the number of segments in which FC is partitioned and their size are chosen arbitrarily. Although Vanneschi et al. further investigated the limitations of NSC and made some modifications [29], none normalization technique has been found. Poli et al. [30] further modified NSC, namely fitness-proportional NSC, which created fitness clouds using fitness proportional selection.

E. Dispersion Metrics

Dispersion Metrics (DM) measures the average distance between pairs of high quality solutions [31]:

$$DISP_{\alpha\%} = \frac{1}{(\alpha n)(\alpha n-1)} \sum_{i=1}^{\alpha n-1} \sum_{j=i+1}^{\alpha n} ||x_i - x_j|| \quad (9)$$

where α represents top $\alpha\%$ population is chosen. Therefore the computational complexity of DM is $O(\alpha n^2)$. If the dispersion increases when the samples are restricted to better regions of the search space, the landscape has a weak global structure, and the problem is difficult to solve. Therefore, DM reflects global structure, evolvability and deception. DM demonstrated why CMA-ES algorithm is ineffective in multimodal problems [31]. The limitation of dispersion metrics is that it converges to $\frac{1}{\sqrt{6}}$ when dimensionality increases significantly [32].

F. Information Characteristics Analysis

Information Characteristics Analysis (ICA) was first proposed by Vassilev et al. to explore the characteristics of fitness landscapes [33]. It assumes statistically isotropic landscapes, which is not always correct. A random walk is performed to obtain a sequence of fitness values $\{f(x_i)\}$, where $0 \leq i \leq n$. Then, a string $S(\epsilon) = s_1 s_2 \dots s_n$ is calculated according to the following principle:

$$S_i(\epsilon) = \begin{cases} -1, & if\ f(x_i) - f(x_{i-1}) < -\epsilon \\ 0, & if\ |f(x_i) - f(x_{i-1})| \leq \epsilon \\ 1, & if\ f(x_i) - f(x_{i-1}) > \epsilon \end{cases} \quad (10)$$

where ϵ is a parameter. If ϵ is small, $S_i(\epsilon)$ is sensitive to the change of fitness values. Three FLA metrics namely Information Content (IC), Partial Information Content (PIC) and Information Stability (IS) are proposed to gain insights into the structure of fitness landscapes.

- IC refers to information entropy, which is defined as

$$H(\epsilon) = -\sum_{p \neq q} Pr_{[pq]} log_6 Pr_{[pq]} \quad (11)$$

  where $Pr_{[pq]}$ is the frequency of the sequence $[pq]$ occurred in the whole string $S_i(\epsilon)$. The computational complexity of IC is $O(n)$. IC measures the number of sequential fitness values which are different for the given parameter ϵ. Therefore, It refers to the ruggedness, smoothness and neutrality of the landscapes.
- PIC refers to the modality of the fitness landscapes. Removing all the $0s$ and the symbols which are equal to the preceding one from the string$S_i(\epsilon)$,we obtain a new string $S'(\epsilon)$. PIC is defined as

$$M(\epsilon) = \frac{\mu}{n} \quad (12)$$

  where μ is the length of the string $S'(\epsilon)$. Therefore, the computational complexity of PIC is $O(n)$. PIC reflects the change of slopes in the random walk, which implies local optima in the landscape. The number of local optima is approximately equal to $\frac{M(\epsilon)n}{2}$.
- IS is the range of two sequential fitness values during the random walk. It can be defined by:

$$\epsilon^* = \min\{\epsilon,\ S_i(\epsilon) = 0\}. \quad (13)$$

  High rugged landscapes tend to have large IS, while the IS for smooth landscape is small. Therefore, IS captures ruggedness, smoothness and neutrality of landscapes. If the number of candidate ϵ is e, the computational complexity of IS is $O(en)$.

ICA was adapted for the continuous optimization domain later on [34], [35]. Our previous work [18] pointed out two limitations of ICA: firstly, it will increase the uncertainly by ignoring the distance ρ; secondly, random walk may generate biased samples, which breaks the assumption of statistic isotropy. To address these limitations, we introduced the distance Δx to the quantification rule:

$$S_i(\epsilon) = \begin{cases} -1, & if \frac{\Delta y}{\|\Delta x\|} < -\epsilon \\ 0, & if \left|\frac{\Delta y}{\|\Delta x\|}\right| \leq \epsilon \\ 1, & if \frac{\Delta y}{\|\delta x\|} > \epsilon \end{cases} \quad (14)$$

where Δy is the difference between y_{i+1} and y_i, $\|\Delta x\|$ is the euclidean distance between x_{i+1} and x_i. Besides, Latin Hypercube Design (LHD) sampling method was implemented to remove the sampling bias.

In summary, we have discussed in detail a number of well-performed FLA metrics. As shown in Table I, the problem characteristics captured by each FLA metric are different. We investigate how these FLA metrics capture certain problem characteristics, and analyse the computational complexity of each FLA metric. When selecting the combination of FLA metrics, two criteria should be followed. Firstly, the chosen metrics should capture all the problem characteristics presented in Section II. Secondly, the total computational cost should be minimal. Following these rules, it is possible to identify the best combination of FLA metrics.

IV. Benchmark Functions

When a new FLA metric is proposed, benchmark functions are employed to compare the performance with others. We

TABLE I: COMPARISON OF FLA METRICS REGARDING RELATED PROBLEM CHARACTERISTICS AND COMPUTATIONAL COMPLEXITY.

FLA metrics	Related problem characteristics	Computational complexity
Number of decision variables	Dimensionality	$O(1)$
Fitness distance correlation	Evolvability, Deception	$O(n)$
The number of peaks in fitness distribution	Modality	$O(mn)$
The skewness of fitness distribution	Symmetry	$O(n)$
The kurtosis of fitness distribution	Ruggedness, Smoothness, Neutrality	$O(n)$
Length scale	Ruggedness, Smoothness, Neutrality	$O(n^2)$
Fitness cloud	Evolvability, Deception	$O(kn)$
Negative slope coefficient	Evolvability, Deception	$O(\max\{qn, mn\})$
Dispersion metric	Evolvability, Deception	$O(\alpha n^2)$
Information content	Ruggedness, Smoothness, Neutrality	$O(n)$
Partial information content	Modality	$O(n)$
Information stability	Ruggedness, Smoothness, Neutrality	$O(en)$

present in this section three benchmark suites, discuss the main characteristics of the benchmark problems, and show the way in which these benchmark suites can be used in FLA.

A. Comparison Continuous Optimizer

COmparison Continuous Optimizer (COCO) is a platform for comparison of real-parameter global optimizer. It has been used for the Black-Box Optimization Benchmarking (BBOB) workshop during GECCO from 2009 to 2013. It provides 24 noiseless functions [36], which were classified into 5 groups: separable function ($f1$ to $f5$), functions with low and moderate conditioning ($f6$ to $f9$), functions with high conditioning and unimodal ($f10$ to $f14$), multi-modal functions with adequate global structure ($f15$ to $f19$), multi-modal function with weak global structure ($f20$ to $f24$). The COCO benchmarking suite is widely used for comparison of the performances of algorithms [37]. The COCO benchmark suite has also been successfully employed to compare the performances of FLA metrics [18], [25].

B. CEC'2013 Special Session on Large Scale Global Optimization

Benchmark functions for the CEC'2013 special session on large scale global optimization are designed to compare the performances of various large-scale global optimization algorithms [38]. The benchmark suite consists of 15 functions, which are classified into 4 groups: fully-separable functions ($f1$ to $f3$), partially additively separable functions ($f4$ to $f11$), overlapping functions ($f12$ to $f14$), and fully non-separable functions ($f15$).

The benchmark suite has been widely used to investigate the performance of global optimization algorithms on large scale problems [39]. However, it hasn't been used to evaluate FLA metrics. For the variable interactions of the benchmark functions are already known, we can employ the benchmark suite to investigate performance of the newly proposed FLA metrics regarding separability.

C. CEC'2013 Special Session on Niching Methods for Multimodal Function Optimzation

The benchmark suite for CEC'2013 special session on niching methods for multi-modal function optimization consists of 20 functions [40]. It has been employed to compare the performances of niching algorithms [41]. Researchers can also use this benchmark suite to investigate the performance of FLA metrics regarding modality.

V. CONCLUSION AND FUTURE WORK

A. Conclusion

In this paper, we bring together problem characteristics, FLA metrics and benchmark functions used in the continuous optimization domain. We present the most important problem characteristics and discuss how these characteristics are related to problem difficulties. We analyse some well-performed and computationally less expensive metrics and discuss each of them in detail. The contributions and limitations of each metric have been carefully investigated. We map the FLA metrics with problem characteristics and analyse the computational complexity of each FLA metric. Two criteria are proposed to follow when selecting the best FLA metrics. Finally, three widely used benchmark suites have been presented and how these benchmark suites can be employed in FLA has been pointed out.

B. Future Work

One possible direction for future work is to apply the FLA metrics on the BBOB suite and compare the performances of each metric. The design of metrics, especially the computationally less intensive and well-performed metrics, should be focused on. Moreover, more work needs to be done to adapt the metrics, which were originally designed for combinatorial optimization problems to the continuous optimization domain. Finally, the limitations of the metrics we have discussed should be addressed.

REFERENCES

[1] J. Rice, "The algorithm selection problem," *Advances in Computers*, vol. 15, no. c, pp. 65–118, 1976.

[2] S. Wright, "The roles of mutations, inbreeding, crossbreeding and selection in evolution," *Proceedings of the 11th International Congress of Genetics*, vol. 1, pp. 356–366, 1932.

[3] P. F. Stadler, "Fitness Landscapes," *Biological Evolution and Statistical Physics*, pp. 187–207, 2002.

[4] E. Pitzer and M. Affenzeller, "A comprehensive survey of Fitness Landscape Analysis," in *Recent Advances in Intelligent Engineering Systems*, pp. 161–190, 2011.

[5] K. M. Malan and A. P. Engelbrecht, "A survey of techniques for characterising fitness landscapes and some possible ways forward," *Information Sciences*, vol. 241, pp. 148–163, 2013.

[6] K. M. C. Clymont, "Recent Advances in Problem Understanding : Changes in the Landscape a Year On," *GECCO Companion*, no. 24, pp. 1071–1077, 2013.

[7] J. Horn and D. Goldberg, "Genetic algorithm difficulty and the modality of fitness landscapes," in *Foundations of Genetic Algorithms, Volume 3* (L. D. Whitley and M. D. Vose, eds.), pp. 243–269, Morgan Kaufmann, 1995.

[8] J. Garnier and L. Kallel, "How to detect all maxima of a function," *Theoretical aspects of evolutionary computing*, 2001.

[9] T. Weise, M. Zapf, R. Chiong, and A. J. Nebro, "Why is optimization difficult?," *Studies in Computational Intelligence*, vol. 193, pp. 1–50, 2009.

[10] L. Kallel, B. Naudts, and C. Reeves, "Properties of fitness functions and search landscapes," *Theoretical aspects of evolutionary computing*, 2001.

[11] W. Beaudoin, S. Verel, P. Collard, and C. Escazut, "Deceptiveness and neutrality the ND family of fitness landscapes," *Proceedings of the 8th annual conference on Genetic and evolutionary computation*, pp. 507–514, 2006.

[12] T. Weise, R. Chiong, and K. Tang, "Evolutionary Optimization: Pitfalls and Booby Traps," *Journal of Computer Science and Technology*, vol. 27, pp. 907–936, Nov. 2012.

[13] Y. Liu, X. Yao, Q. Zhao, and T. Higuchi, "Scaling up fast evolutionary programming with cooperative coevolution," *Proceedings of the 2001 Congress on Evolutionary Computation (IEEE Cat. No.01TH8546)*, vol. 2, pp. 1101–1108, 2001.

[14] M. Omidvar, X. Li, Y. Mei, and X. Yao, "Cooperative co-evolution with differential grouping for large scale optimization," *IEEE Transcations on Evolutionary Computation*, vol. 18, no. 3, pp. 378–393, 2014.

[15] Y. Davidor, "Epistasis variance: Suitability of a representation to genetic algorithms," *Complex Systems*, vol. 4, pp. 369–383, 1990.

[16] R. Salomon, "Re-evaluating genetic algorithm performance under coordinate rotation of benchmark functions. A survey of some theoretical and practical aspects of genetic algorithms.," *Bio Systems*, vol. 39, pp. 263–78, Jan. 1996.

[17] 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. 11, no. 1, pp. 1–18, 2003.

[18] M. Munoz, M. Kirley, and S. Halgamuge, "Exploratory landscape analysis of continuous space optimization problems using information content," *IEEE Transactions on Evolutionary Computation*, no. c, pp. 1–1, 2014.

[19] Y. Li, "understand behavior and performance of real coded optimization algorithms via nk-linkage model," *Evolutionary Computation*, pp. 455–464, 2008.

[20] M. Tomassini, L. Vanneschi, P. Collard, and M. Clergue, "A study of fitness distance correlation as a difficulty measure in genetic programming.," *Evolutionary computation*, vol. 13, no. 2, pp. 213–239, 2005.

[21] T. Jones and S. Forrest, "Fitness Distance Correlation as a Measure of Problem Difficulty for Genetic Algorithms.," *ICGA*, vol. 95, pp. 184–192, 1995.

[22] L. Kallel, "On functions with a given fitness-distance relation," *Evolutionary Computation, 1999. CEC 99. Proceedings of the 1999 Congress on*, pp. 1910–1916, 1999.

[23] H. Rose, W. Ebeling, and T. Asselmeyer, "The Density of States - A Measure of the Difficulty of Optimisation Problems," in *Parallel Problem Solving from Nature - {PPSN IV}* (H.-M. Voigt et al., ed.), vol. 1141 of *Lecture Notes in Computer Science*, pp. 208–217, Springer, Berlin, Heidelberg, New York, 1996.

[24] O. Mersmann, B. Bischl, and H. Trautmann, "Exploratory landscape analysis," *Proceedings of the 13th annual conference on Genetic and evolutionary computation*, pp. 829–836, 2011.

[25] R. Morgan and M. Gallagher, "Length scale for characterising continuous optimization problems," *Parallel Problem Solving from Nature-PPSN XII*, pp. 407–416, 2012.

[26] S. Verel, P. Collard, and M. Clergue, "Where are bottlenecks in NK fitness landscapes?," *The 2003 Congress on Evolutionary Computation, 2003. CEC '03.*, vol. 1, 2003.

[27] P. Collard, S. Verel, and M. Clergue, "Local search heuristics: Fitness cloud versus fitness landscape," *arXiv preprint arXiv:0709.4010*, pp. 3–4, 2007.

[28] L. Vanneschi, M. Clergue, P. Collard, M. Tomassini, and S. Vérel, "Fitness Clouds and Problem Hardness in Genetic Programming," in *Genetic and Evolutionary Computation GECCO2004 Part II*, vol. 3103 of *Lecture Notes in Computer Science*, pp. 690–701, Springer-Verlag, 2004.

[29] L. Vanneschi, M. Tomassini, P. Collard, and S. V??rel, "Negative slope coefficient: A measure to characterize genetic programming fitness landscapes," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 3905 LNCS, pp. 178–189, 2006.

[30] R. Poli and L. Vanneschi, "Fitness-Proportional Negative Slope Coefficient as a Hardness Measure for Genetic Algorithms," in *Proceedings of the 9th Genetic and Evolutionary Computation Conference, GECCO 2007*, pp. 1335–1342, ACM Press, 2007.

[31] M. Lunacek and D. Whitley, "The dispersion metric and the CMA evolution strategy," *Proceedings of the 8th annual conference on Genetic and evolutionary computation*, pp. 477–484, 2006.

[32] R. Morgan and M. Gallagher, "Sampling Techniques and Distance Metrics in High Dimensional Continuous Landscape Analysis: Limitations and Improvements," *Evolutionary Computation, IEEE Transactions on*, vol. 18, no. 3, pp. 456–461, 2014.

[33] V. K. Vassilev, T. C. Fogarty, and J. F. Miller, "Information characteristics and the structure of landscapes.," *Evolutionary computation*, vol. 8, pp. 31–60, Jan. 2000.

[34] K. Steer, A. Wirth, and S. Halgamuge, "Information theoretic classification of problems for metaheuristics," in *Simulated Evolution and Learning*, pp. 319–328, Springer Berlin heidelberg, 2008.

[35] K. Malan and A. Engelbrecht, "Quantifying ruggedness of continuous landscapes using entropy," in *Evolutionary Computation, 2009. CEC'09. IEEE Congress on. IEEE*, pp. 1440–1447, 2009.

[36] N. Hansen, A. Auger, S. Finck, and Raymond Ros, "Real-Parameter Black-Box Optimization Benchmarking 2009: Experimental Setup," tech. rep., 2009.

[37] N. Hansen, A. Auger, R. Ros, S. Finck, and P. Pošík, "Comparing results of 31 algorithms from the black-box optimization benchmarking BBOB-2009," *Proceedings of the 12th annual conference comp on Genetic and evolutionary computation - GECCO '10*, p. 1689, 2010.

[38] X. Li, K. Tang, M. Omidvar, Z. Yang, K. Qin, and H. China, "Benchmark functions for the CEC 2013 special session and competition on large-scale global optimization," *gene*, pp. 1–23, 2013.

[39] M. Omidvar, X. Li, Y. Mei, and X. Yao, "Cooperative co-evolution with differential grouping for large scale optimization," vol. 18, no. 3, pp. 378–393, 2014.

[40] X. Li, A. Engelbrecht, and M. Epitropakis, "Benchmark functions for cec'2013 special session and competition on niching methods for multimodal function optimization," *Evolutionary Computation*, pp. 1–10, 2013.

[41] I. Loshchilov, "CMA-ES with restarts for solving CEC 2013 benchmark problems," *2013 IEEE Congress on Evolutionary Computation*, pp. 369–376, June 2013.