Exploratory Landscape Analysis of Continuous Space Optimization Problems Using Information Content

Mario A. Muñoz, Michael Kirley, and Saman K. Halgamuge, *Senior Member, IEEE*

***Abstract*—Data-driven analysis methods, such as the information content of a fitness sequence, characterize a discrete fitness landscape by quantifying its smoothness, ruggedness, or neutrality. However, enhancements to the information content method are required when dealing with continuous fitness landscapes. One typically employed adaptation is to sample the fitness landscape using random walks with variable step size. However, this adaptation has significant limitations: random walks may produce biased samples, and uncertainty is added because the distance between observations is not accounted for. In this paper, we introduce a robust information content-based method for continuous fitness landscapes, which addresses these limitations. Our method generates four measures related to the landscape features. Numerical simulations are used to evaluate the efficacy of the proposed method. We calculate the Pearson correlation coefficient between the new measures and other well-known exploratory landscape analysis measures. Significant differences on the measures between benchmark functions are subsequently identified. We then demonstrate the practical relevance of the new measures using them as class predictors on a machine learning model, which classifies the benchmark functions into five groups. Classification accuracy greater than 90% was obtained, with computational costs bounded between 1% and 10% of the maximum function evaluation budget. The results demonstrate that our method provides relevant information, at a low cost in terms of function evaluations.**

***Index Terms*—Fitness landscape, function classification, knowledge acquisition, landscape analysis, unconstrained optimization.**

I. Introduction

EXPLORATORY landscape analysis (ELA) are for data-driven methods that produce numerical measures related to the features of the fitness landscape; hence, they provide insights into the nature of a given optimization problem [1]. Originally developed in the context of combinatorial optimization [2], ELA methods are used to differentiate between classes of problems, and to automatically select the most efficient algorithm for a given problem [1], [3]–[5].

In the continuous optimization domain, there is some evidence that ELA methods provide information about the neutrality, smoothness, or ruggedness of the landscape [1]. This information is useful because, in a smooth landscape, the difference in fitness values between two neighboring candidate solutions, or observations, is small but typically nonzero. Thus, an optimization algorithm can generate new and fitter observations from its neighbors, requiring fewer function evaluations to solve the problem. This is untrue for neutral landscapes where the difference in fitness is zero or in rugged landscapes where the fitness can fluctuate widely. However, it is difficult to measure these features because the neighborhood is not well defined in continuous spaces [6]. In addition, most of the existing ELA methods for continuous spaces do not analyze the changes between neighbors, limiting the extent to which meaningful analysis can be carried out.

ELA methods that analyze fitness changes between neighbors on discrete landscapes have been proposed. Some of them use random walks to sample the input space, i.e., the space defined by the input variables of the optimization problem. For example, Weinberger's random walk correlation function [7], Stadler's correlation length [8], and Vassilev *et al.*'s information content-based method [9]. these methods assume that a random walk captures the fitness changes in the landscape because the walk generates a sequence of neighboring observations.

Independently and almost simultaneously, Steer *et al.* [10] and Malan and Engelbrecht [11] adapted Vassilev *et al.*'s [9] method to continuous spaces. In addition, they summarize the results using four measures that quantify the smoothness of the landscape. However, there are limitations in their work. First, they use a random walk with an adjustable step size as sampling method, which requires a large sample to guarantee unbiasedness. Hence, the method is expensive in terms of function evaluations. With a limited size, the sample is highly concentrated in some areas of the input space, while other areas are not sampled at all. Therefore, the sample and the results are biased. Second, they did not account for the distance between two observations in the analysis. Here, distance is a random variable and a source of uncertainty.

Manuscript received May 31, 2013; revised October 17, 2013; accepted January 14, 2014. Date of publication January 22, 2014; date of current version January 28, 2015. This work was supported in part by the DAAD/Go8 Academic Exchange Grant and in part by the research scholarships awarded to M. A. Muñoz by The University of Melbourne and the Ministry of Information and Communication Technologies, Colombia.

M. A. Muñoz and S. K. Halgamuge are with the Department of Mechanical Engineering, The University of Melbourne, Melbourne VIC 3010, Australia (e-mail: mariom@alumni.unimelb.edu.au; saman@unimelb.edu.au).

M. Kirley is with the Department of Computer and Information Systems, The University of Melbourne, Melbourne VIC 3010, Australia (e-mail: mkirley@unimelb.edu.au).

Digital Object Identifier 10.1109/TEVC.2014.2302006

A. Contributions of the Paper

In this paper, we improve Vassilev *et al.*'s [9] method for continuous spaces by addressing the limitations found in [10], [11]. We call our new version information content of fitness sequences (ICoFiS). In our approach, we include the distance between two observations in the analysis, and we use Latin hypercube design (LHD) to generate an unbiased sample and a sorting algorithm to sequence the observations. We also compare the effectiveness of two different sorting algorithms and we identify their advantages and disadvantages.

We carry out detailed experiments to analyze the data generated by ICoFiS. We use the comparing continuous optimizers (COCO) noiseless benchmark set [12] as test functions. To minimize the cost of the analysis compared with the cost of running an algorithm, we set the lower bound for the sample size at $10^2 \times D$ function evaluations and the upper bound at $10^3 \times D$, where D is the dimensionality of the problem. These bounds correspond to 1% and 10% of the maximum number of function evaluations usually employed for algorithm testing and analysis [12].

In addition, we calculate the Pearson correlation coefficient between the ICoFiS measures and a group of ELA measures [1], [5], [13]–[16]. Significant differences on the measures between benchmark functions are subsequently identified. We then demonstrate the practical relevance of the measures. We evaluate the accuracy of a machine learning classifier whose inputs are the ICoFiS measures and the output is one of the five function classes defined by Hansen *et al.* [12]. With an accuracy above 90%, the classifier can be used to select an appropriate algorithm to solve a given optimization problem [17].

An important conclusion drawn from our experiments is that there is strong correlation (≈ 0.96) between the entropy of the probability distribution of the fitness values, $H(\mathbf{Y})$, and the maximum change of fitness found during a sequence of observations, ϵ_S. Our results indicate that ϵ_S is an important measurement for the classification task.

B. Outline of the Paper

The paper is organized as follows. Section II describes Vassilev *et al.*'s method for discrete spaces and its adaptations to continuous spaces. Section III presents an experimental evaluation of these adaptations and their limitations. Section IV presents the details of ICoFiS, which we have proposed to address the limitations of the methods in [10] and [11]. Section V describes the numerical experiments used to evaluate ICoFiS and Section VI presents the results. Section VII discusses the findings, emphasizing the advantages and disadvantages of ICoFiS. Section VIII presents the conclusions.

II. Background and Related Work

A. Information Content Of the Fitness Sequences On Discrete Space Problems

The information content of an object is a measure of how difficult it is to describe such an object. Applying this concept to fitness landscapes, a neutral landscape is a low information object because we can describe it using the input space bounds and the fitness of one observation. In contrast, a rugged landscape is a high information object because the number of observations required to describe it depends on the number of local optima, which can be very high.

Let $f : \mathcal{X} \mapsto \mathcal{Y}$ be a function where $\mathcal{X}$ is the input space and $\mathcal{Y}$ is the output space, and $\mathbf{x}_i \in \mathcal{X}$ is an observation. To measure the information content of the landscape of f, Vassilev *et al.* [9] proposed a method to analyze the sequence of fitness values $\mathbf{S} = \{y_1, \ldots, y_n\}$ resulting from a random walk over the landscape, where $y_i = f(\mathbf{x}_i), i = 1, \ldots, n$. Let $\Phi(\epsilon) = \{\phi_1, \ldots, \phi_{n-1}\}$ be a symbol sequence where $\phi_i \in \{\bar{1}, 0, 1\}$. The sequence of fitness values is converted into the symbol sequence by following the rule:

$$\phi_i = \begin{cases} \bar{1} \text{ if } & y_{i+1} - y_i < -\epsilon \\ 0 \text{ if } & |y_{i+1} - y_i| \leq \epsilon \\ 1 \text{ if } & y_{i+1} - y_i > \epsilon \end{cases} \tag{1}$$

where $\epsilon \geq 0$ is the information sensitivity, an accuracy parameter of the symbol sequence. For example, if $\epsilon = 0$, the symbol sequence has a zero iff there are neutral areas in the landscape. On the other hand, the symbol sequence has only zeros if ϵ is larger than the maximum difference between consecutive fitness values.

Two consecutive symbols compose a block, which represents an object in the fitness landscape, i.e., a slope, peak or neutral area. The information content of the sequence is defined as

$$H(\epsilon) = -\sum_{a \neq b} p_{ab} \log_6 p_{ab} \tag{2}$$

where $a, b \in \{\bar{1}, 0, 1\}$ and p_{ab} is the probability of finding the block ab in the symbol sequence. The logarithm base is six because this is the number of possible blocks where $a \neq b$. The information content is bound between $[0, 1]$ and $0 \log_6 0 \equiv 0$.

According to Vassilev *et al.* [9], the information content calculated with (2) measures the variety of objects in the fitness landscape, but it does not explicitly measure the smoothness of the landscape. To estimate the smoothness, a new sequence $\Phi'(\epsilon)$ is constructed from $\Phi(\epsilon)$ by removing all the zeros and any repeated symbols. The new sequence has the form $\ldots \bar{1}1\bar{1}1\bar{1} \ldots$ and represents the changes of concavity encountered during the random walk. Vassilev *et al.* [9] defined the partial information content, $M(\epsilon)$, as a measure of smoothness, which is calculated as

$$M(\epsilon) = \frac{\left|\Phi'\right|}{n-1} \tag{3}$$

with $n \gg 1$. The method assumes that the fitness landscape is statistically isotropic, i.e., the probability distribution of the fitness values obtained during a random walk does not change between walks. This is an important assumption, as it implies that the method requires an unbiased sample from the input space.

B. Information Content Of Fitness Sequences On Continuous Space Problems

In both adaptations of Vassilev *et al.*'s method to continuous optimization problems [10], [11], the random walk is

constructed by sampling the first observation $\mathbf{x}_1$ at random, and the next $n-1$ observations using the rule

$$\mathbf{x}_{i+1} = \mathbf{x}_i + \varrho \cdot \delta \tag{4}$$

where δ is a vector of random variables with $\|\delta\| = 1$, which controls the direction; and ϱ is a random variable that controls the step size and it is uniformly distributed between $[0, r]$, where r is the maximum step size. The resulting sequence of fitness values is discretized using (1). The information content is calculated using (2) and the partial information content is calculated using (3).

The results are plotted against ϵ resulting in the curves presented in Fig. 1. Steer *et al.* [10] proposed four measures to summarize the curves: maximum information content (5), settling sensitivity (6), initial partial information (7) and half partial information sensitivity (8). On the other hand, Malan and Engelbrecht [11] calculate only on the maximum information content, and normalize the values of ϵ over the value of $\epsilon^\star = \min_\epsilon H(\epsilon) = 0$

$$H_{\max} = \max_{\epsilon} \{H(\epsilon)\} \tag{5}$$

$$\epsilon_s = \log_{10}\left(\min_{\epsilon} \{\epsilon : H(\epsilon) < 0.05\}\right) \tag{6}$$

$$M_0 = M(\epsilon = 0) \tag{7}$$

$$\epsilon_{0.5} = \log_{10}\left(\max_{\epsilon} \{\epsilon : M(\epsilon) > 0.5 M_0\}\right). \tag{8}$$

These measures correspond to four points on the information content curves as illustrated on Fig. 1. At H_{max} the symbol sequence $\Phi(\epsilon)$ has the highest diversity, i.e., the sequence has the maximum number of rugged elements. According to Malan and Engelbrecht [11], a rugged landscape will have a higher value of H_{max}. At ϵ_S, the sequence is nearly all zeros, and it represents the maximum change of fitness found during the sequence of observations, giving indication of the scaling of the problem. At M_0, the sequence has the highest number of inflexion points, which is reduced to half at $\epsilon_{0.5}$. Both of these measures provide information about the landscape ruggedness. By increasing the number of distinct values of ϵ used, we increase both the accuracy of the measures and the smoothness of the curves.

III. Evaluation of the Method on Continuous Space Problems

In this section, we evaluate the method proposed in [10] and [11] to identify possible limitations. Section III-A presents the numerical experimental procedure and its results. Section III-B presents the implications of these results.

A. Preliminary Experimental Evaluation

As described in Section II-B, in previous work a random walk was generated following the rule in (4) [10], [11]. This rule requires the selection of a maximum step size r, raising the question: what should the value of the step size be? In addition, Vassilev *et al.* assumed statistical isotropy of the fitness landscape [9], implying that the sample should represent the diversity of the output space. It is well known that the probability of uniformly covering the input space with a random walk is only achieved with a sample size converging to infinity. Hence, in [10] and [11], the measures are extracted by averaging the result of several walks initiated from different random starting points. This approach is intended to improve the coverage of input space, raising another question: does this approach produce an unbiased sample when the number of observations is limited?

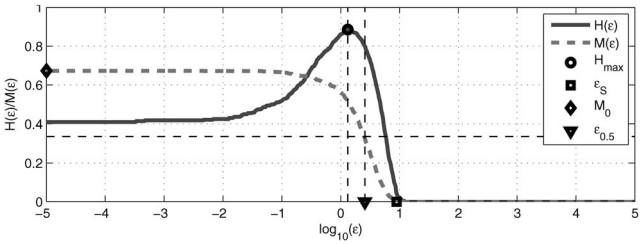

Fig. 1. Typical curves obtained using the method proposed in [10] and [11], and the measures obtained with (5) to (8). On the horizontal axis is the base-10 logarithm of the sensitivity parameter ϵ. The black dashed lines are visualization aids.

To answer these questions, we have generated three groups of ten independent random walks of 200 observations following the rule in (4), for a total of 2000 observations per group. The input space is 2-D with bounds at $[-5, 5]^2$. For each group, the maximum step size r is equal to $\{0.01, 0.055, 0.10\} \times R$, where $R = \sqrt{200}$ is the diagonal of the input space. The values of r are equally spaced on a linear scale. We have evaluated the observations obtained through the random walks using three functions from the COCO benchmark set [12]: Sphere (unimodal and smooth function), Rastrigin (multimodal, rugged and symmetrical function), and Gallagher Gaussian 21-hi Peaks (multimodal and asymmetrical function).

The procedure described in Section II-A was used to generate the curves for $H(\epsilon)$ and $M(\epsilon)$, with ϵ equal to zero and to 1000 different values between 10^{-5} and 10^{15} equally spaced in a base-10 logarithmic scale. We use a larger proportion of small values of ϵ because the small changes in $H(\epsilon)$ and $M(\epsilon)$ can be significant for characterizing the ruggedness with respect to the neutrality of the landscape [18]. The results (Fig. 2) match the results presented in [10], and they show that for three different values of r there are different values for each measure depending on the function. In addition, the results show that the curves appear to converge, indicating the existence of an optimal r for each function.

To further explore these results, we train a Gaussian mixture model to identify the probability distribution of the observations extracted during the random walk. We used a grid search method to identify an appropriate number of components for the model. The results are illustrated in Fig. 3. An inspection of this figure reveals that the group of random walks does not produce an unbiased sample, unless a large step size is selected. A small step size, such as $r = 0.01 \times R$ produces a highly concentrated sample in small areas of the input space, as illustrated in Fig. 3(a) and (b). While increasing the

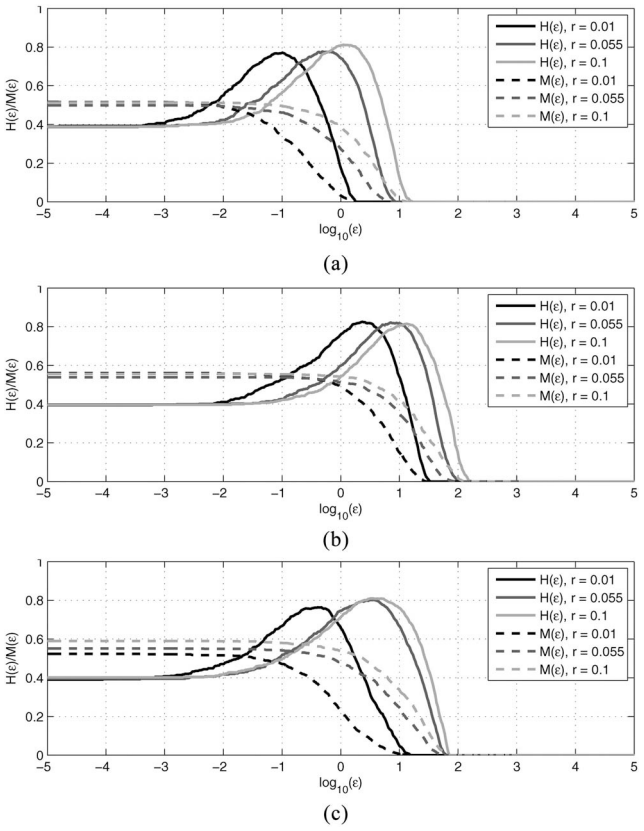

Fig. 2. Information analysis of the Sphere (f_1), Rastrigin (f_3), and Gallagher's Gaussian 21-hi Peaks (f_{22}) functions. The results are the average of three groups of ten random walks of 200 observations with $r = \{0.01, 0.055, 0.10\} \times R$, where R is the diagonal of the input space. (a) Information analysis for the Sphere function (f_1). (b) Information analysis for the Rastrigin function (f_3). (c) Information analysis for the Gallagher's Gaussian 21-hi Peaks Function (f_{22}).

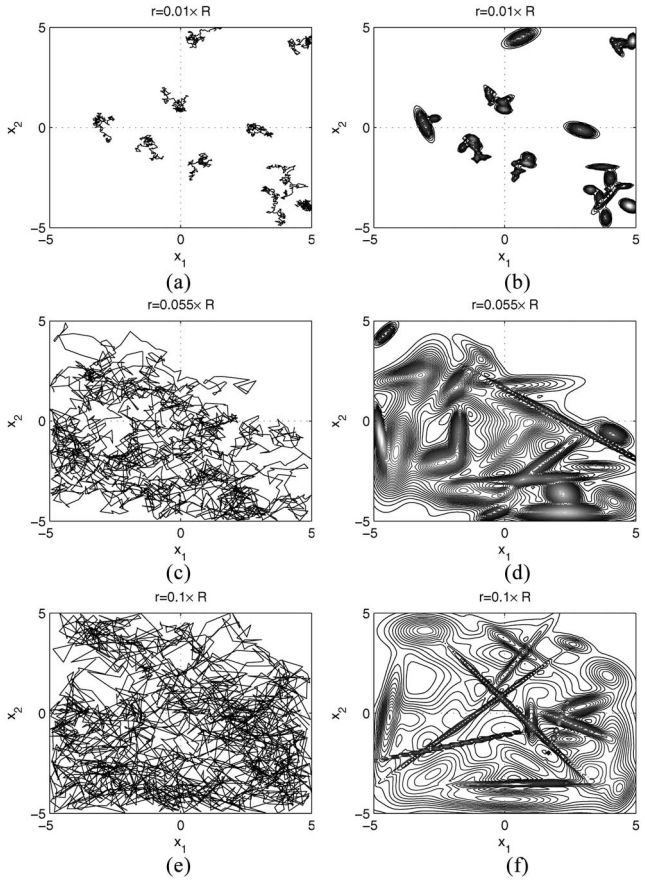

Fig. 3. Trajectories of the random walks and probability distribution of the resulting sample. (a), (c), and (e) Trajectories of four groups of ten random walks of 200 observations over the input space. (b), (d), and (f) Contour plots of the probability distributions of the respective random walk.

step size improves the spread of the observations, areas with high concentrations remain. For example, Fig. 3(c) and (d) illustrate the results for $r = 0.055 \times R$. Several areas of high concentration of observations can be identified, while one large area does not have any observations at all. Fig. 3(e) and (f) illustrate the results $r = 0.10 \times R$. The sample in these figures is better distributed, but there are a few areas with high concentrations.

We extract the results for the four measures for the average, as well as the maximum and minimum value over the group with $r = 0.1 \times R$. We include the value of $\epsilon_{\max}$ which is equal to $\max_\epsilon \{\epsilon : H(\epsilon) = H_{max}\}$. These results are presented in Table I.

TABLE I

VALUES OF FOUR MEASURES FOR SPHERE (f_1), RASTRIGIN (f_3), AND GALLAGHER'S GAUSSIAN 21-HI PEAKS (f_{22}) FUNCTIONS

		$H_{\max}$	$\varepsilon_{\max}$	ε_S	M_0	$\varepsilon_{0.5}$
f_1	Maximum	0.849	0.170	1.232	0.583	0.551
	Average	0.823	0.056	1.128	0.513	0.407
	Minimum	0.796	-0.070	0.972	0.452	0.291
f_3	Maximum	0.866	1.373	2.234	0.598	1.673
	Average	0.827	1.140	2.150	0.556	1.387
	Minimum	0.792	0.952	2.014	0.472	1.253
f_{22}	Maximum	0.886	0.952	1.854	0.628	1.313
	Average	0.833	0.665	1.820	0.590	1.094
	Minimum	0.796	0.371	1.754	0.558	0.611

B. Limitations of the Method

Fig. 3 demonstrates that the approach in [10] and [11] generates biased samples, breaking the assumption of statistical isotropy. This means that the results in Fig. 2 may lead to invalid conclusions when the sample size is limited. For example, the average values of $H_{\max}$ presented in Table I for the three functions converges to ≈ 0.827, even though the functions have different levels of smoothness/ruggedness. In fact, it appears that the range of $H_{\max}$ provides more information than the average. A wider range implies a rugged landscape. In addition, the range of ϵ_S is large, indicating that the scale of Δy changes between walks. We can also observe that for functions of low symmetry, such as f_{22}, the start position of the random walk influences the final result.

In previous work, a difference between measures was found when large sample sizes were employed, e.g., 500 independent walks of 1000 observations for $D = 10$ [10] or 30 independent walks of 10 000 observations for $D = 30$ [11]. This computational cost is unacceptable, as some algorithms such as IDEA, BIPOP-CMA-ES, and BFGS can solve a large

number of problems under $10^4 \times D$ function evaluations [17]. Besides, sampling bias cannot be removed by averaging the results from biased experiments. Instead, methods such as stratified and weighted sampling are used [19], [20]. Fig. 3 also demonstrates that there is no guarantee that the random walk will not over-explore some regions while neglecting others. Furthermore, a random walk may back-track to a nearby observation, creating the illusion of local optima in the sequence. Although these observations apply to input spaces of two dimensions, such issues are likely to be compounded in higher dimensions.

Even if we assume that the random walk produces an unbiased sample, the step size is not accounted for in the quantification rule in (1). The rule is appropriate in a discrete space where the neighborhood is finite and well defined, and the distance between neighbors is equal to one. However, in continuous space, the neighborhood is infinite, not well defined and dependent on the radius of a hypersphere centered in $\mathbf{x}_i$ [6]. The random walk generates observations whose distance is ϱ, which is a random variable. By ignoring the value of ϱ, an important source of uncertainty is not considered in the method.

IV. Information Content of Fitness Sequences

We propose the information content of fitness sequences (ICoFiS) method to address the two limitations identified on Section III-B. We address the first limitation by changing the quantification rule from (1) to

$$\Psi(i,\epsilon) = \begin{cases} \bar{1} \text{ if } \frac{\Delta y}{\|\Delta \mathbf{x}\|} < -\epsilon \\ 0 \text{ if } \left|\frac{\Delta y}{\|\Delta \mathbf{x}\|}\right| \leq \epsilon \\ 1 \text{ if } \frac{\Delta y}{\|\Delta \mathbf{x}\|} > \epsilon \end{cases} \tag{9}$$

where $\epsilon \geq 0$ is the sensitivity parameter, Δy is the difference between y_{i+1} and y_i, and $\|\Delta \mathbf{x}\|$ is the Euclidean distance between $\mathbf{x}_{i+1}$ and $\mathbf{x}_i$. Therefore, ICoFiS accounts for the uncertainty added by the step size, which is represented by $\|\Delta \mathbf{x}\|$. The value of $\frac{\Delta y}{\|\Delta \mathbf{x}\|}$ converges to the derivative if $\|\Delta \mathbf{x}\|$ converges to zero. This approach shares some similarities with the length scale method [21].

To address the second limitation, we replace the random walk with Latin hypercube design (LHD) sampling. Therefore, ICoFiS uses an unbiased, space filling sample with a uniform distribution. However, a LHD sample does not generate a sequence of observations, which is required by the method.

Hence, we propose two alternative sorting algorithms to construct the sequence after sampling. For both of them, the initial element of the sequence is randomly selected. In the first algorithm, the subsequent observations are also selected at random. In the second algorithm, the next observation is the nearest neighbor of the current observation. To avoid repetitions, we keep a candidate list of those observations not in the sequence. Due to the two proposed sorting algorithms, there are two variations of ICoFiS which we compare.

V. Experimental Methodology

A series of numerical experiments are carried out to evaluate whether the modifications in ICoFiS address the limitations discussed in Section III-B, and whether the measures generated by ICoFiS are useful to discriminate between functions. The data necessary for these experiments is collected following the procedure described in Section V-A. The validation of the modifications in ICoFiS is described in Section V-B, whereas the validation of the measures generated by ICoFiS is described in Section V-C.

A. Data Collection

We use the COCO noiseless benchmark set [12] at $D = \{2, 5, 10, 20\}$ dimensions as test functions. The software implementation of the benchmark set generates instances by translating and rotating the function in the input and output spaces. For example, let $f(\mathbf{x}) = \|\mathbf{R}(\mathbf{x} - \mathbf{x}^\star)\|^2 + y^\star$ be one of the functions in the benchmark set, where $\mathbf{R}$ is an orthogonal rotation matrix, $\mathbf{x}^\star$ and $y^\star$ cause translational shifts on the input and output space, respectively. A function instance is generated by providing values for $\mathbf{R}$, $\mathbf{x}^\star$ and $y^\star$. To allow replicable experiments, the software uniquely identifies each instance using an index. For each function at each dimension, we analyze instances $[1, \ldots, 15]$.

Ideally, the additional computational cost of the analysis should be a fraction of the cost associated with a single search algorithm, which is usually bounded at $10^4 \times D$ function evaluations [12]. Therefore, we set the lower bound for the sample size at $10^2 \times D$ (1% of the maximum algorithm cost) and the upper bound at $10^3 \times D$ (10% of the maximum algorithm cost), which we consider to be reasonable costs for the analysis stage. We divided the range between $10^2 \times D$ and $10^3 \times D$ into five equally sized intervals using a base-10 logarithmic scale. As a result, we have five samples for each dimension. The samples have sizes $n = \{100, 178, 316, 562, 1000\} \times D$ observations, where each one is roughly twice the size of the previous one. We generate the input samples using LHD sampling.

An output sample is the result of evaluating an input sample in one of the 15 instances of the 24 functions from the COCO benchmark. A dataset is the combination of an input and an output sample; therefore, a dataset of n observations has $D + 1$ variables. For example, a dataset for a 2-D function has three variables: the two input variables, $[x_{1,i}, x_{2,i}]$, and the output variable, y_i. With one dataset per each instance of each function, at each dimension and for each sample size, we have a total of 7200 datasets to analyze ($24 \times 15 \times 4 \times 5 = 7200$). To calculate the ICoFiS measures, we generate two sequences from each input sample, one using random sorting and one using Nearest Neighbor sorting. Then, we extract the $H(\epsilon)$ and $M(\epsilon)$ curves and the measures as described in Section II-B with ϵ equal to zero and to 1000 different values between 10^{-5} and 10^{15} equally spaced in a base-10 logarithmic scale. We have a total of eight measures. Four measures obtained with random sorting and four with nearest neighbor sorting.

B. Validation of ICoFiS Modifications

To evaluate whether our modifications address the limitations discussed in Section III-B, we carry out numerical experiments to answer the following questions.

- Q1. What is the effect of replacing the quantification rule in (1) with (9) on the curves of $H(\epsilon)$ and $M(\epsilon)$ as the sample size increases?
- Q2. What are the effects that random and nearest neighbor sorting have on the measures?

To answer question Q1, we analyze the probability distributions of Δy and $\frac{\Delta y}{\|\Delta \mathbf{x}\|}$ for the LHD samples sequenced with the nearest neighbor sorting method. The variance of the distributions have an effect on the curves of $H(\epsilon)$ and $M(\epsilon)$. The higher the variance, the higher the value of ϵ must be to obtain a symbol sequence of only zeros. As a result, the curves appear to shift to the right. On the other hand, the lower the variance, the lower the value of ϵ. As a result, the curves appear to shift to the left. Ideally, for two samples of different size but with distributions of equal variance, the curves of $H(\epsilon)$ and $M(\epsilon)$ converge.

We use a nonparametric two sample Ansari–Bradley test of equal variance at the 95% significance level to compare the spread, whilst avoiding the assumption of normality. The null hypothesis, the variances of the distributions are equal, cannot be rejected if the p-value is higher than 0.05. We hypothesize that there is not a significant difference in variance for the distribution of $\frac{\Delta y}{\|\Delta \mathbf{x}\|}$ for two different sample sizes. For each function and instance at each dimension, we test the distributions resulting from the five sample sizes between each other, resulting in ten independent tests per instance, and family size of 150 tests per function. To avoid a multiple testing problem and an increased number of Type I errors, we use Benjamini and Yekuteli's method to correct the p-values resulting from the tests over each family [22], setting the False Discovery Rate (*FDR*) at 5%. According to Groppe *et al.* [23], if the method rejects some hypotheses using conventionally low alpha levels, such as 5%, we can be confident that the number of false discoveries is smaller than the number of correct rejections. We summarize the results by counting the percentage of tests which are statistically significant for each function at each dimension. The results are presented in Section VI-A1.

To answer question Q2, we examine the probability distribution of the change in position, $\|\Delta \mathbf{x}\|$, for three LHD samples of size $\{100, 562, 1000\} \times D$ sequenced with both sorting methods, with $D = 2$. The values of $\|\Delta \mathbf{x}\|$ are normalized between $[0, 1]$ using the diagonal of the input space equal to $\sqrt{200}$ as a scale factor. In addition, we calculate the probability of finding any block in the sequence for the LHD sample of 2000 observations, where a block is a sequence of any two symbols as already defined in Section II-A. We evaluate the sample on the Sphere (f_1), Rastrigin (f_3) and Gallagher Gaussian 21-hi Peaks (f_{22}) functions. The probabilities are calculated at $\epsilon = \{0, \epsilon_{\max}\}$, where $\epsilon_{max} = \max_{\epsilon} \{\epsilon : H(\epsilon) = H_{max}\}$. We selected these two values of ϵ because they are the values where M_0 and $H_{\max}$ are calculated. The results are presented in Section VI-A2.

C. Validation of ICoFiS Measures

To evaluate whether ICoFiS generates information useful to discriminate between functions, we carry out numerical experiments to answer the following questions.

- Q3. Is there a linear correlation between the ICoFiS measures?
- Q4. Is there a linear correlation between the ICoFiS measures and other ELA measures?
- Q5. If we compare the value of a measure for two functions, instances or sample sizes, would the difference between them be statistically significant?
- Q6. Are the ICoFiS measures good predictors of the function structure?

High positive correlation between any two measures implies that they are equivalent; whereas high negative correlation implies that they are diametrically opposed. Hence, the computational effort of the landscape analysis can be reduced by calculating only the simplest measure. Differences between two measures that are not statistically significant indicate that the measure converges to the same value. Ideally, the difference should be significant when the measures for two different problems are compared, but not when the measures are from two instances of the same problem, or when the measure is calculated with two samples of different sizes. Finally, low accuracy of the classifier implies that the measures have little predictive power, and they might not provide useful information for further analysis.

To answer question Q3, we calculate the Pearson correlation between each of the eight measures. We follow the rule of thumb by Hinkle *et al.* [24], which considers a correlation between 0.7 and 0.9 to be high, and between 0.9 and 1.0 to be very high. Those measures with very high correlation are removed from the next analysis stages. The results for this question are presented in Section VI-B1.

To answer question Q4, we calculate the Pearson correlation between each of the remaining measures and the following ELA measures.

1) Fitness distance correlation (*FDC*) measures the correlation between the distance to the fittest observation and the fitness value. It identifies whether a landscape is unimodal or multimodal, and whether it has a strong global structure or not [13], [25].
2) Dispersion at 1% ($DISP_{1\%}$) is the average Euclidean distance between the 1% fittest observations normalized over the diagonal of the input space. It assumes that a well-correlated landscape has observations with similar fitness close to each other [14].
3) Skewness ($\gamma(\mathbf{Y})$), kurtosis ($\kappa(\mathbf{Y})$), and entropy ($H(\mathbf{Y})$) of the $\mathcal{Y}$-distribution are indicators of the fitness landscape smoothness and global structure, as well as the diversity of the fitness values [1], [15].
4) The adjusted coefficient of determination $\left(\bar{R}^2_L\right)$, for linear (L), linear with iterations (LI), quadratic with interactions (QI) and quadratic (Q) models are used to evaluate the modality and the global structure of the landscape [1]. This approach can be thought of as

measuring the distance between a reference problem and the problem under analysis [26].

5) The minimum and maximum of the absolute value of the linear model coefficients $(\min(|\beta_L|), \max(|\beta_L|))$, and the ratio between the minimum and the maximum absolute values of the quadratic term coefficients in the quadratic model (CN) measure the scaling of the problem [1].
6) The information significance of first $\left(\xi^{(1)}\right)$, second $\left(\xi^{(2)}\right)$ and D-th $\left(\xi^{(D)}\right)$ orders and the standard deviation for the significance of first $\left(\sigma_\xi^{(1)}\right)$ and second $\left(\sigma_\xi^{(2)}\right)$ orders measure nonlinear dependencies between the input variables [5], [16].

To calculate each of these measures, we employ the same datasets used to calculate the ICoFiS measures. The results are presented in Section VI-B2.

To answer question Q5, we estimate the probability distribution of each measure for each function instance. Instead of generating a limited number of different LHD samples, we use bootstrapping [27] to generate 2000 bootstrap datasets from each one of the original datasets. From each bootstrap dataset, we generate two sequences, one with random and one with nearest neighbor sorting. To avoid backtracking with Nearest Neighbor sorting, we temporarily remove all repeats of the last two observations in the sequence from the candidate list. Then, we extract the $H(\epsilon)$ and $M(\epsilon)$ curves with ϵ equal to zero and to 1000 different values between 10^{-5} and 10^{5} equally spaced in a base-10 logarithmic scale. Then, we calculate the measures as described in Section II-B, resulting in 2000 different values of each measure from which we estimate the probability distribution.

We employ the Wilcoxon rank-sign test of equal medians at the 95% significance level to identify whether the difference in a measure is statistically significant. As it was the case with the Ansari–Bradley test described in Section V-B, we employ this test to avoid the assumption of normality on the distribution of the measures. The null hypothesis, the median, and the mean if it exists, of the two distributions are equal, cannot be rejected if the p-value is higher than 0.05. We test the significance of the difference of a measure between functions, instances and sample sizes for a given dimension as follows.

1) Between functions, we compared one instance at the time for each function, resulting in 276 independent tests. With 15 instances and five sample sizes the family size is 20 700 tests per dimension.
2) Between instances, we compared 15 instances for each function, resulting in 105 independent tests. With 24 functions and five sample sizes the family size is 12 600 tests per dimension.
3) Between sample sizes, we compared the five sample sizes for each instance, resulting in ten independent tests. With 15 instances and 24 functions the family size is 3600 tests per dimension.

We apply Benjamini and Yekuteli's [22] method to correct the p-values resulting from the each one tests over the complete family, setting the *FDR* at 5%. Then, we summarize the results by counting the number of times the null hypothesis was rejected. If a measure has a high number of null hypotheses not rejected between function comparisons, then it is removed from the next analysis stages. The results are presented in Section VI-B3.

To answer question Q6, we train a random forest classifier for each of the sorting algorithms. The inputs to the classifiers are the dimension and the sample size and those measures remaining after the previous analysis stages. We use as output of the classifier one of the five categories in which Hansen *et al.* [12] classified the 24 functions from the COCO benchmark set. These categories are: separable functions $\{f_1, \ldots, f_5\}$, functions with low or moderate conditioning $\{f_6, \ldots, f_9\}$, functions with high conditioning and unimodal $\{f_{10}, \ldots, f_{14}\}$, multimodal functions with adequate global structure $\{f_{15}, \ldots, f_{19}\}$, and multimodal functions with weak global structure $\{f_{20}, \ldots, f_{24}\}$. According to Hansen *et al.* [17], each class has a group of best performing algorithms. Hence, the classifier can be used as a solution for the algorithm selection problem [2], [28].

To calculate the prediction accuracy of the model and the influence of each input, we use the out-of-the-bag (OOB) observations as validation data. When a decision tree is generated, approximately 2/3 of the observations are used for training. The remaining 1/3 are the OOB observations. We calculate the confusion matrix to evaluate the prediction accuracy of the model, and the OOB change in the prediction error as a measure of the influence of each input [29]. This measure is calculated as follows. After training the tree, the OOB observations are evaluated and the prediction accuracy is recorded. Then, the values for one input are randomly permuted over the OOB observations, nullifying the effect of the input. The effect is similar to set a coefficient in a linear model to zero [29]. Then, the prediction accuracy is recorded again. The procedure is repeated for all inputs and trees. The change in the prediction accuracy is averaged over all trees and transformed into a normalized, dimensionless z-score. A number higher than 1.96 indicates that the change in the prediction accuracy is statistically significant. The results are presented in Section VI-B4.

VI. Results

A. Validation of ICoFiS Modifications

Two questions were presented in Section V-B as a road map to identify the usefulness of the measures extracted using ICoFiS. In this section, we present the answers to each of these questions.

1) *Effect of Replacing the Quantification Rule:* In this section, we provide an answer to question Q1. The results show that by replacing the quantification rule in (1) with (9), we improve the convergence of the curves of $H(\epsilon)$ and $M(\epsilon)$, regardless of the sample size.

Fig. 4(a) shows the probability distributions for Δy for three the functions from the COCO benchmark set in two dimensions, with sample sizes equal to $100, 562, 1000 \times D$. These functions are the Sphere (f_1), Rastrigin (f_3), and Gallagher's

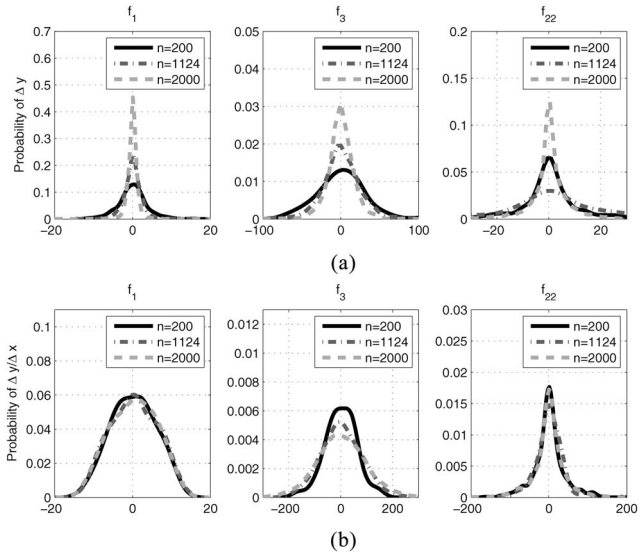

Fig. 4. Effect of the quantification rule on the probability distributions of the changes between sequential observations. The distributions are three LHD samples of size $\{100, 562, 1000\} \times D$ observations. The functions under evaluation are the Sphere (f_1), Rastrigin (f_3), Gallagher's Gaussian 21-hi Peaks (f_{22}). (a) Distribution of Δy. (b) Distribution of $\frac{\Delta y}{\|\Delta \mathbf{x}\|}$.

Gaussian 21-hi Peaks (f_{22}). Fig. 4(b) shows the probability distributions of $\frac{\Delta y}{\|\Delta \mathbf{x}\|}$ for the same functions. To estimate the distributions, we have employed a kernel density estimator with a Gaussian kernel and bandwidth $h = 1.06\hat{\sigma} n^{-1/5}$, where $\hat{\sigma}$ is the sample standard deviation. Table II shows the sample mean ($\hat{\mu}$) and the sample standard deviation ($\hat{\sigma}$) of the distributions. Fig. 4(a) shows that the distributions have similar $\hat{\mu}$ but different $\hat{\sigma}$, which according to Table II decreases with the step size. This implies that the curves will shift to the left as the sample size increases, because the step size decreases. On the other hand, Fig. 4(b) shows that the probability distributions appear to have a similar mean and standard deviation. Using the results in Table II, we find that the difference between the standard deviation after normalization is lower for f_1 and f_{22}. For f_3 the difference is larger.

Table III shows the percentage of tests for which the Ansari–Bradley test could not reject the null hypothesis, i.e., the variances are equal, for each function at each dimension. We have boldfaced the results for those functions for which at least 50% of the null hypothesis were not rejected. We observe that, with the exception of f_{16} and f_{23}, the normalization reduces the difference in the variance of the distributions; hence, the curves of $H(\epsilon)$ and $M(\epsilon)$ converge. In some cases, such as f_{16} and f_{23}, the normalization is detrimental to the convergence. These two functions are highly rugged with the peaks and nadirs of the function closely located. When we increase the sample size, we capture more peaks and nadirs of the function. Therefore, we obtain larger values of $\frac{\Delta y}{\|\Delta \mathbf{x}\|}$. These results indicate that there is a manifestation of the No-Free Lunch Theorem at work in this experiment [30].

2) *Effect of the Sorting Algorithm on Measures:* In this section, we provide an answer to question Q2. The results demonstrate that the sorting algorithm has an influence on the values of $H_{\max}$ and M_0. In particular, random sorting produces

TABLE II
MEAN AND STANDARD DEVIATION OF PROBABILITY DISTRIBUTIONS OF CHANGES BETWEEN SEQUENTIAL OBSERVATIONS

		Δy			$\Delta y/\|\Delta \mathbf{x}\|$		
		200	1124	2000	200	1124	2000
$\hat{\mu}$	f_1	0.177	-0.006	-0.002	0.287	0.261	0.337
	f_3	-1.160	0.173	0.023	0.753	1.457	2.972
	f_{22}	-0.097	-0.011	0.020	0.195	0.381	1.159
$\hat{\sigma}$	f_1	4.463	2.237	1.744	5.591	5.928	5.993
	f_3	40.186	20.887	21.738	60.703	83.230	94.454
	f_{22}	18.961	10.554	8.940	32.920	41.875	43.331

TABLE IV
MEAN AND STANDARD DEVIATION OF PROBABILITY DISTRIBUTIONS OF STEP SIZE FOR A LHD SAMPLE. THE STEP SIZE HAS BEEN NORMALIZED OVER DIAGONAL OF INPUT SPACE. RESULTS DEMONSTRATE THAT DISTRIBUTIONS HAVE FASTER CONVERGENCE FOR RANDOMLY SORTED SAMPLES

	Random			Nearest Neighbor		
	200	1124	2000	200	1124	2000
μ	0.359	0.362	0.371	0.047	0.018	0.014
σ	0.163	0.174	0.178	0.056	0.021	0.021

similar values of $H_{\max}$ and M_0 regardless of the function, while nearest neighbor sorting produces different values for $H_{\max}$ and M_0 between functions.

Table IV shows the values of $\hat{\mu}$ and $\hat{\sigma}$ for the distributions of $\|\Delta \mathbf{x}\|$. With constant dimension and increasing sample size, both $\hat{\mu}$ and $\hat{\sigma}$ converge toward zero when using nearest neighbor sorting. On the other hand, $\hat{\mu}$ converges toward 0.370 and $\hat{\sigma}$ converges toward 0.170 when using random sorting. The value of $\hat{\mu}$ is close to the theoretical convergence limit of the average distance of $\frac{1}{\sqrt{6}}$ [31], meaning that we quickly lost any contrast between observations in terms of distance. We conclude that the speed of convergence is higher with random sorting, because the relative difference on $\hat{\mu}$ and $\hat{\sigma}$ between sequences is smaller with random sorting than with nearest neighbor sorting.

It is possible to analyze the results from Table IV from a different angle. Keeping the dimension constant while the sample size decreases has a similar effect as increasing the dimension while the sample size is constant. This implies that, for nearest neighbor sorting, the distribution of the step size does not converge toward zero as the dimension increases. On the other hand, for random sorting, the distribution of the step size is not strongly affected by the sample size. This suggests that the results of ICoFiS are less prone to change with the sample size, when random sorting is used instead of nearest neighbor sorting. In addition, random sorting is computationally simpler. Conceptually, random sorting is similar to using a random walk with a maximum step size $r = R$ that produces representative samples.

Arguably, it is desirable to have a method whose results are only dependent on the value of the function and not on internal or external factors. Therefore, random sorting

TABLE III

PERCENTAGE OF TESTS FROM A FAMILY OF 150 ANSARI–BRADLEY TESTS OF EQUAL VARIANCE FOR WHICH NULL HYPOTHESIS, VARIANCES ARE EQUAL, COULD NOT BE REJECTED AT A 5% SIGNIFICANCE LEVEL. VALUES HIGHER THAN 50% ARE HIGHLIGHTED IN BOLDFACE. RESULTS SHOW THAT NORMALIZATION PRODUCES EQUAL VARIANCE DISTRIBUTIONS FOR MOST FUNCTIONS

	Δy				$\Delta y/\lvert\Delta \mathbf{x}\rvert$			
	2	5	10	20	2	5	10	20
f_1	7.3%	14.7%	22.7%	39.3%	**100.0%**	**100.0%**	**100.0%**	**98.0%**
f_2	10.0%	25.3%	37.3%	**56.0%**	**100.0%**	**100.0%**	**100.0%**	**99.3%**
f_3	19.3%	29.3%	46.0%	**54.0%**	**73.3%**	**99.3%**	**100.0%**	**100.0%**
f_4	16.7%	49.3%	**58.0%**	**67.3%**	**58.7%**	**100.0%**	**100.0%**	**100.0%**
f_5	0.0%	10.7%	18.7%	24.7%	**100.0%**	**100.0%**	**100.0%**	**100.0%**
f_6	22.0%	21.3%	22.7%	38.0%	**100.0%**	**100.0%**	**100.0%**	**100.0%**
f_7	8.7%	24.7%	28.7%	42.0%	29.3%	**100.0%**	**100.0%**	**100.0%**
f_8	24.7%	26.0%	37.3%	**72.7%**	**100.0%**	**100.0%**	**100.0%**	**82.0%**
f_9	22.0%	30.7%	46.7%	**100.0%**	**100.0%**	**100.0%**	**95.3%**	**54.0%**
f_{10}	10.0%	21.3%	29.3%	43.3%	**100.0%**	**100.0%**	**100.0%**	**100.0%**
f_{11}	9.3%	26.0%	36.0%	**58.0%**	**100.0%**	**100.0%**	**100.0%**	**100.0%**
f_{12}	26.7%	46.7%	**64.0%**	**76.0%**	**100.0%**	**100.0%**	**100.0%**	**100.0%**
f_{13}	0.7%	13.3%	23.3%	40.0%	**100.0%**	**100.0%**	**100.0%**	**100.0%**
f_{14}	20.0%	31.3%	38.7%	**57.3%**	**100.0%**	**100.0%**	**100.0%**	**100.0%**
f_{15}	19.3%	24.7%	35.3%	48.0%	**80.0%**	**100.0%**	**100.0%**	**100.0%**
f_{16}	36.7%	**100.0%**	**100.0%**	**100.0%**	26.0%	16.7%	20.7%	28.7%
f_{17}	**60.0%**	**86.7%**	**66.7%**	**68.7%**	36.7%	44.0%	**90.7%**	**100.0%**
f_{18}	**86.0%**	**93.3%**	**77.3%**	**70.7%**	24.0%	42.7%	**82.7%**	**100.0%**
f_{19}	**74.7%**	31.3%	**52.7%**	**97.3%**	14.7%	**100.0%**	**97.3%**	**68.0%**
f_{20}	15.3%	23.3%	30.0%	**54.0%**	**100.0%**	**100.0%**	**100.0%**	**93.3%**
f_{21}	14.0%	38.7%	**62.0%**	**100.0%**	**96.7%**	**90.0%**	**80.0%**	**79.3%**
f_{22}	10.7%	31.3%	**61.3%**	**96.0%**	**100.0%**	**100.0%**	**100.0%**	**100.0%**
f_{23}	**100.0%**	**100.0%**	**100.0%**	**100.0%**	4.0%	11.3%	20.0%	28.0%
f_{24}	**99.3%**	46.0%	44.7%	**96.0%**	6.7%	32.0%	**69.3%**	48.7%

seems a preferable approach. However, let us analyze the effect that sorting has on the symbol sequence. Fig. 5 shows the probability distribution of finding any symbol block in the sequence, for a LHD sample of 2000 observations. The sample was evaluated using the f_1, f_3 and f_{22} functions. The probabilities were calculated at $\epsilon = \{0, \epsilon_{\max}\}$. For Fig. 5(a) and (c) the sequence is sorted at random, while for Figs. 5(b) and 4(d) the sequence is sorted using nearest neighbor.

Fig. 5(a) shows that the probability of finding a block with $\phi_i = 0$ is zero for either function. The probability of finding an inflexion block, i.e., either $(\bar{1}, 1)$ or $(1, \bar{1})$, is close to $\frac{1}{3}$ for either function. The probability of a repetition block, i.e., either $(\bar{1}, \bar{1})$ or $(1, 1)$, is approximately $\frac{1}{6}$. The results imply that the value of M_0 is similar between functions when random sorting is employed and it converges to $\frac{2}{3}$. Fig. 5(b) shows that the value of M_0 is different between functions when Nearest Neighbor sorting is employed. Although the probability of finding a block with $\phi_i = 0$ is still zero for either function, the probability of finding an inflexion block is lower for f_1 than for the f_3, implying a lower value of M_0 in the former than in the later. Fig. 5(c) and (d) shows that the difference between functions in the probabilities is less evident when random sorting is employed. This implies that three different functions will have similar values of $H_{\max}$ when random sorting is employed.

These results provide some evidence of the advantages and disadvantages of ICoFiS. For example, random sorting is a simpler algorithm for which the average step size quickly converges to the theoretical estimate of $\frac{1}{\sqrt{6}}$ [31]. Hence, its results may not be negatively affected by the sample size. However, we have evidence that ICoFiS produces similar values of $H_{\max}$ and M_0 regardless of the function with this sorting algorithm, which is undesirable. On the other hand, nearest neighbor sorting may be negatively affected by the sample size. However, we have evidence that ICoFiS produces different values of $H_{\max}$ and M_0 between functions. These results do not provide evidence of the behavior of ϵ_S and $\epsilon_{0.5}$. We gather further evidence in the second stage of the experimental study, described in the following section.

B. Validation of ICoFiS Measures

Four questions were presented in Section V-C as a road map to identify the usefulness of the measures extracted using ICoFiS. In this section, we present the answers to each of these questions.

1) *Correlation Between ICoFiS Measures:* In this section, we provide an answer to question Q3. Table V presents the Pearson correlation ($\rho_{x,y}$) between the measures. We use the superscript *R* for random sorting and *NN* for nearest neighbor sorting to differentiate the measures depending on the sorting algorithm. In addition, we have boldfaced those values that represent a high positive correlation, i.e., $\rho_{x,y} \geq 0.7$ [24]. The table shows that ϵ_S and $\epsilon_{0.5}$ are almost perfectly correlated, i.e., $\rho_{x,y} \approx 1.0$, regardless of the sorting algorithm employed. This implies that one of these two measures may be sufficient for further analysis. Since ϵ_S is the simplest of the two to be calculated, and even it can be replaced by the maximum value of $\frac{\Delta y}{\|\Delta \mathbf{x}\|}$, it will be used from now on.

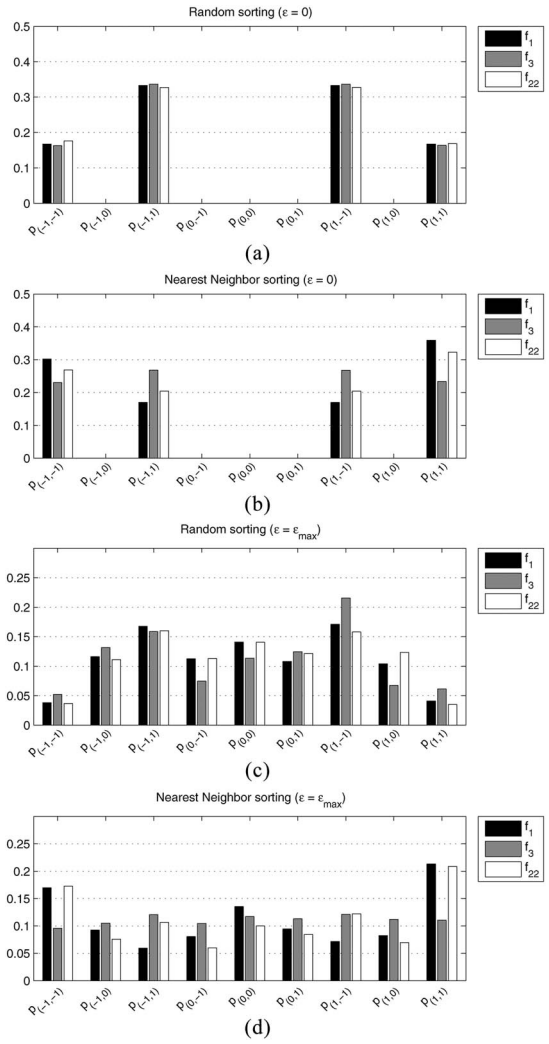

Fig. 5. Probabilities of finding the different blocks in the symbol sequence. (a) Random sorting with $\epsilon = 0$. (b) Nearest neighbor sorting with $\epsilon = 0$. (c) Random sorting with $\epsilon = \epsilon_{max}$. (d) Nearest neighbor sorting with $\epsilon = \epsilon_{max}$.

If we focus on the results for M_0^R, we see that it is not strongly correlated with any other measure. With the results shown in Fig. 5, it is possible to assume that M_0^R might produce the same result for several functions. This would explain why M_0^R is uncorrelated with any other measure. We provide additional evidence in Section VI-B3 by testing the statistical significance of the difference between functions. On the other hand, $H_{\max}^{NN}$ has high correlation with M_0^{NN} and moderate correlation with $H_{\max}^R$. Fig. 6 illustrates these results. Although M_0^{NN} and $H_{\max}^R$ increase when $H_{\max}^{NN}$ increases, we are unable to conclude based on those results only that H_{max}^{NN} can be substituted by either $H_{\max}^R$ or M_0^{NN}. The figure also shows that $H_{\max}^R$ has a very narrow range of variation, as expected from the results on Fig. 5. In addition, there is little indication that this measure is strongly discriminative as the results of very different functions are localized in different parts of the measure range. This result provides some confirmation of the importance of sorting the observations by nearest neighbors if we are to use $H_{\max}$ and M_0.

2) *Correlation Between ICoFiS Measures and Other ELA Measures:* In this section, we provide an answer to question Q4. Table VI shows the values of the Pearson correlation between $H_{\max}$, ϵ_S and M_0 and the additional ELA measures. The table shows that ϵ_S is almost perfectly correlated to the entropy of the $\mathcal{Y}$-distribution, $H(\mathbf{Y})$. This result was not expected as the two measures are obtained through two completely different methods.

If we take $\epsilon_S^R = \{4, 6\}$ as thresholds, we can classify the functions in to three groups. Functions with $4 < \log_{10}(\epsilon) \leq 6$ are $\{f_6, f_8, f_9, f_{20}\}$, functions with $\log_{10}(\epsilon) > 6$ are $\{f_2, f_{10}, f_{11}, f_{12}\}$, while the remaining functions have $4 \leq \log_{10}(\epsilon)$. Using the qualitative features described by Mersmann *et al.* [32], we can identify the first group as having

TABLE V

PEARSON CORRELATION ($\rho_{x,y}$) BETWEEN MEASURES FROM ICOFIS FOR FUNCTIONS ON COCO BENCHMARK SET. IN BOLDFACE ARE THOSE MEASURES WITH HIGH CORRELATION, I.E., $\rho_{x,y} \geq 0.7$. VALUES OF ϵ_S AND $\epsilon_{0.5}$ ARE ALMOST PERFECTLY CORRELATED REGARDLESS OF THE SORTING ALGORITHM

	ε_S^R	M_0^R	$\varepsilon_{0.5}^R$	$H_{\max}^{NN}$	ε_S^{NN}	M_0^{NN}	$\varepsilon_{0.5}^{NN}$
H_{max}^R	-0.402	0.056	-0.350	0.671	-0.409	0.346	-0.304
ε_S^R		-0.015	**0.994**	-0.472	**0.995**	-0.329	**0.988**
M_0^R			-0.021	0.021	-0.014	0.023	-0.017
$\varepsilon_{0.5}^R$				-0.457	**0.984**	-0.346	**0.992**
$H_{\max}^{NN}$					-0.464	**0.807**	-0.406
ε_S^{NN}						-0.306	**0.988**
M_0^{NN}							-0.294

TABLE VI

PEARSON CORRELATION BETWEEN MEASURES FROM INFORMATION CONTENT METHOD AND ADDITIONAL ELA MEASURES FOR FUNCTIONS ON COCO BENCHMARK SET. IN BOLDFACE ARE THOSE MEASURES THAT HAVE VERY STRONG POSITIVE CORRELATION. VALUES OF ϵ_S^R AND ϵ_S^{NN} ARE ALMOST PERFECTLY CORRELATED TO VALUES OF $H(\mathbf{Y})$

	$H_{\max}^R$	ε_S^R	M_0^R	$H_{\max}^{NN}$	ε_S^{NN}	M_0^{NN}
FDC	0.113	0.044	-0.039	-0.206	0.005	-0.478
$DISP_{1\%}$	0.259	-0.049	-0.003	0.534	-0.065	0.667
$\bar{R}_L^2$	0.005	0.177	-0.075	-0.190	0.134	-0.411
$\bar{R}_{LI}^2$	-0.124	0.319	-0.061	-0.244	0.268	-0.431
$\bar{R}_{QI}^2$	-0.103	0.347	-0.021	-0.294	0.295	-0.497
$\bar{R}_Q^2$	0.023	0.204	-0.033	-0.234	0.160	-0.464
$\min(\beta_L)$	-0.199	0.206	0.016	-0.172	0.210	-0.069
$\max(\beta_L)$	-0.235	0.271	0.015	-0.151	0.274	-0.027
CN	0.269	-0.303	-0.002	0.143	-0.294	-0.024
$H(\mathbf{Y})$	-0.290	**0.984**	-0.019	-0.367	**0.969**	-0.257
$\xi^{(D)}$	0.070	0.248	-0.010	-0.138	0.259	-0.279
$\xi^{(1)}$	-0.106	0.127	-0.029	-0.394	0.128	-0.501
$\sigma\left(\xi^{(1)}\right)$	-0.199	-0.021	-0.031	-0.374	-0.024	-0.504
$\xi^{(2)}$	-0.046	0.212	-0.030	-0.329	0.218	-0.482
$\sigma\left(\xi^{(2)}\right)$	-0.029	-0.129	0.021	0.042	-0.154	0.020
$\gamma(\mathbf{Y})$	-0.330	0.462	-0.001	-0.182	0.467	-0.044
$\kappa(\mathbf{Y})$	-0.155	0.214	-0.005	-0.057	0.215	0.028

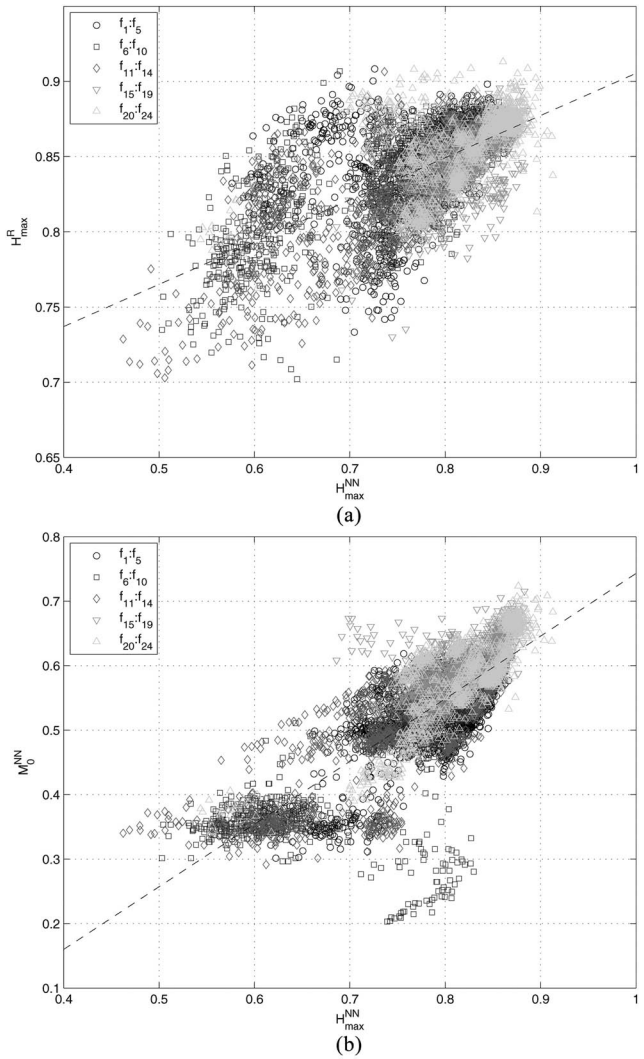

Fig. 6. Scatter plot of $H^{NN}_{\max}$ versus (a) $H^{R}_{\max}$ and (b) M^{NN}_{0}. The observations have been classified following the groups defined by Hansen *et al.* [12]. Although $H^{NN}_{\max}$ has a strong correlation to both measures, there is not a discernible pattern that might be exploited. (a) $H^{R}_{\max}$ versus $H^{NN}_{\max}$. (b) M^{NN}_{0} versus $H^{NN}_{\max}$.

moderate variable scaling, the second as high variable scaling and the third group has a mixture of functions from low to high variable scaling. To illustrate this result, Fig. 7 shows a scatter plot of the values of ϵ_S and $H(\mathbf{Y})$ for the 24 functions from the COCO benchmark, at all dimensions and sample sizes. The figure demonstrate that, although they have different scale, these two measures are linearly related, which implies that ϵ_S can be replaced by $H(\mathbf{Y})$.

This confirms that ϵ_S is a variable scaling indicator. We can provide an intuitive explanation for this phenomenon. Let us assume that a random walk is taking place on the input space, $\mathcal{X}$, of a function with high variable scaling. If the step size is a random variable in the $[0, R]$ range, there is the probability that a small step in $\mathcal{X}$ will produce a large change in the fitness value, if the direction of the step is the same as the variable with low scale. ICoFiS calculates $\frac{\Delta y}{\|\Delta \mathbf{x}\|}$, which for small steps with large changes in fitness is expected to be high. The higher the values of $\frac{\Delta y}{\|\Delta \mathbf{x}\|}$, the higher the value of ϵ_S.

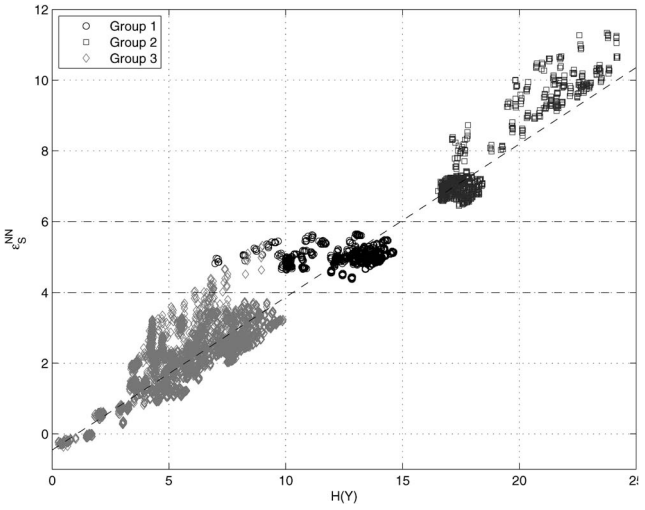

Fig. 7. Scatter plot of ϵ_S^{NN} versus $H(\mathbf{Y})$, for 15 instances of the 24 functions from the COCO benchmark set at $\{2, 5, 10, 20\}$ dimensions with a sample size equal to $\{100, 178, 316, 562, 1000\} \times D$. The plot demonstrates the high correlation between ϵ_S^{NN} and $H(\mathbf{Y})$, which implies that one of the measures can be replaced with the other.

TABLE VII
RESULTS OF TEST OF STATISTICAL SIGNIFICANCE FOR ICOFIS MEASURES. IDEALLY, COMPARISONS BETWEEN FUNCTIONS SHOULD HAVE HIGH NUMBER OF REJECTIONS, WHILE COMPARISONS BETWEEN INSTANCES AND SAMPLE SIZE SHOULD HAVE LOW NUMBER OF REJECTIONS. NUMBER BETWEEN PARENTHESES REPRESENTS SIZE OF THE FAMILY OF TESTS

D	$H^{R}_{\max}$	ε^{R}_{S}	M^{R}_{0}	$H^{NN}_{\max}$	ε^{NN}_{S}	M^{NN}_{0}
Function (20700)						
2	98.2%	99.9%	**1.1%**	98.9%	99.8%	97.4%
5	98.0%	99.9%	**0.0%**	98.8%	99.9%	98.1%
10	97.9%	99.9%	**0.0%**	99.0%	99.9%	98.6%
20	97.9%	99.9%	**0.0%**	99.2%	100.0%	98.7%
Instances (12600)						
2	89.2%	89.1%	**0.0%**	91.0%	92.3%	82.8%
5	90.4%	93.7%	**0.0%**	93.9%	96.2%	89.7%
10	90.2%	96.3%	**0.0%**	93.4%	96.5%	90.9%
20	89.0%	95.5%	**0.0%**	92.3%	96.2%	90.8%
Sample size (3600)						
2	85.3%	69.2%	54.6%	93.7%	91.0%	92.9%
5	83.9%	86.3%	**21.8%**	91.2%	95.5%	94.1%
10	83.6%	90.4%	**5.3%**	92.5%	96.3%	95.0%
20	82.1%	90.8%	**0.1%**	92.2%	95.3%	95.7%

This result implies that the entropy of the distribution over the output space $\mathcal{Y}$ is strongly related to the maximum value of $\frac{\Delta y}{\|\Delta \mathbf{x}\|}$, and they are both related to the variable scaling of the function.

3) *Analysis of Statistical Significance of Difference Between Measures:* In this section, we provide an answer to question Q5. Table VII shows the results of the tests of statistical significance. The results for each measure are grouped by dimension. The table shows that $H_{\max}$ and ϵ_S does produce statistically different results when two functions are compared; however, M^{R}_{0} does not. The effect is stronger as the sample

TABLE VIII
CONFUSION MATRIX FOR TWO RANDOM FOREST CLASSIFICATION MODELS. EACH ONE OF THEM USES AS INPUTS MEASURES WITH EITHER RANDOM OR NEAREST NEIGHBOR SORTING. BOTH MODELS HAVE A SUCCESS RATE OVER 89%

Random sorting								
		Output class					Total rates of success/failure	
		$\{f_1,\ldots,f_5\}$	$\{f_6,\ldots,f_9\}$	$\{f_{10},\ldots,f_{14}\}$	$\{f_{15},\ldots,f_{19}\}$	$\{f_{20},\ldots,f_{24}\}$		
Target class	$\{f_1,\ldots,f_5\}$	17.1%	0.4%	1.9%	1.4%	0.1%	81.9%	18.1%
	$\{f_6,\ldots,f_9\}$	0.4%	15.1%	0.0%	1.0%	0.1%	90.6%	9.4%
	$\{f_{10},\ldots,f_{14}\}$	1.1%	0.0%	19.2%	0.5%	0.0%	92.3%	7.7%
	$\{f_{15},\ldots,f_{19}\}$	1.3%	1.0%	0.6%	17.6%	0.3%	84.4%	15.6%
	$\{f_{20},\ldots,f_{24}\}$	0.1%	0.0%	0.0%	0.1%	20.6%	98.8%	1.2%
Total rates of success/failure		85.5%	91.3%	87.9%	85.5%	97.6%	89.6%	
		14.5%	8.7%	12.1%	14.5%	2.4%		10.4%
Nearest Neighbor sorting								
		Output class					Total rates of success/failure	
		$\{f_1,\ldots,f_5\}$	$\{f_6,\ldots,f_9\}$	$\{f_{10},\ldots,f_{14}\}$	$\{f_{15},\ldots,f_{19}\}$	$\{f_{20},\ldots,f_{24}\}$		
Target class	$\{f_1,\ldots,f_5\}$	18.5%	0.3%	1.3%	0.7%	0.0%	88.7%	11.3%
	$\{f_6,\ldots,f_9\}$	0.3%	15.8%	0.0%	0.5%	0.0%	94.9%	5.1%
	$\{f_{10},\ldots,f_{14}\}$	0.8%	0.0%	19.3%	0.7%	0.0%	92.9%	7.1%
	$\{f_{15},\ldots,f_{19}\}$	0.7%	0.5%	0.8%	18.8%	0.1%	90.1%	9.9%
	$\{f_{20},\ldots,f_{24}\}$	0.0%	0.0%	0.0%	0.0%	20.8%	99.7%	0.3%
Total rates of success/failure		91.3%	95.4%	89.8%	90.6%	99.1%	93.2%	
		8.7%	4.6%	10.2%	9.4%	0.9%		6.8%

size increases. This is strong evidence that M_0^R will converge to the same value regardless of the function, confirming the validity of the interpretation of Fig. 5. This result might be a consequence of the law of large numbers. Regardless of the function and due to random sorting, the probability of finding an inflexion block in the sequence converges to $\frac{1}{3}$ as the number of observations in the sequence increases.

The table also shows that for two instances of the same problem, there is significant difference between the results of a measure. This evidence suggests that the measures capture the dependencies between variables at some level, and it implies that the results of one instance cannot be generalized over the problem. On the other hand, the table shows that for two different sample sizes, there is significant difference between the results of a measure on the same instance. Although in Section VI-A1, we demonstrated that the curves of $H(\epsilon)$ and $M(\epsilon)$ converge regardless of the sample size, this was not reflected on the ICoFiS measures. We can attribute this result to the compounding of precision errors through out the calculations. These results are marginally better for random sorting, implying that there is few practical advantages of random sorting besides speed.

4) *Function Classification Using Measures as Predictors:* In this section, we provide an answer to question Q6. The results demonstrate that ICoFiS could be simplified by only calculating the $H(\epsilon)$ curves, as only $H_{\max}$ and ϵ_S are strong predictors of the function classes. Table VIII shows the confusion matrix for the two classification models, one for random sorting and one for the nearest neighbor sorting. We have used the out-of-the-bag (OOB) observations to validate the models. Both models have high prediction accuracy, with a success rate of 89.6% for random sorting and 93.2% for the nearest neighbor sorting. The class that is the easiest to discriminate is the one composed of the functions $\{f_{20},\ldots,f_{24}\}$, with a success rate of 97.6% for random sorting and 99.1% for the nearest neighbor sorting. This implies that these functions are close together in the space defined by the ICoFiS measures. The table shows that we can predict the class to which the problem belongs using ICoFiS measures as predictors.

TABLE IX
INPUT VARIABLE INFLUENCE FOR CLASSIFICATION MODELS. BOLDFACE SHOWS INPUTS WHICH DO NOT HAVE A STATISTICALLY SIGNIFICANT INFLUENCE ON RESULT

Model with Random Sorting				
n	S	$H_{\max}^R$	ε_S^R	
5.743	2.652	6.088	18.304	
Model with Nearest Neighbor Sorting				
n	S	$H_{\max}^{NN}$	ε_S^{NN}	M_0^{NN}
5.932	**0.795**	9.741	33.994	**1.746**

Table IX shows the influence of each measure on the result. The values less than 1.96 are boldfaced, which means that removing the measure does not have a significant influence on the change of error. We find that M_0^{NN} does not have a significant influence. Considering that M_0^R has the same value regardless of the function, we can say with confidence that $M(\epsilon)$ is not needed for further analysis. In addition, the table shows that the most important measure is ϵ_S.

Hansen *et al.* [17] provides a group of best performing algorithms for each function class. For $\{f_1,\ldots,f_5\}$,

these algorithms are NEWUOA, LSfminbnd, and LSstep. For $\{f_6, \ldots, f_9\}$, these algorithms are NEWUOA and IPOP-SEP-CMA-ES. For $\{f_{10}, \ldots, f_{14}\}$, these algorithms are GLOBAL, iAMaLGaM, and BIPOP-CMA-ES. For $\{f_{15}, \ldots, f_{19}\}$, these are IPOP-SEP-CMA-ES and BIPOP-CMA-ES. Finally, for $\{f_{20}, \ldots, f_{24}\}$, these are GLOBAL and BIPOP-CMA-ES. Hence, the classifier can be used to identify a subgroup of algorithms that might perform well in a new problem.

VII. Discussion

In Section VI, we presented the results of the experiments carried out to validate the modifications in ICoFiS and the practical relevance of the generated measures. Using a function classification experiment and the results by Hansen *et al.* [17], we have proposed an approach to the algorithm selection problem. In this section, we discuss the practical implications and the limitations of our work.

To measure the smoothness of a landscape, defined as the magnitude of change in fitness value within a neighborhood, then the nearest neighbor sorting should be used. We demonstrated that the values of $H^{NN}_{\max}$, ϵ^{NN}_S, and M^{NN}_0 are significantly different between functions. In addition, $H^{NN}_{\max}$ and ϵ^{NN}_S have significant influence on the accuracy of the classification model. These results demonstrate the suitability of ICoFiS to identify and classify groups of functions. However, the nearest neighbor sorting is computationally expensive as it requires several steps. First, we have to calculate the distance between all observations. Then, we have to find the nearest neighbor of each observation. Finally, we have to construct a candidate list, which we must update at each iteration of the sorting algorithm. To justify the computational effort required for the sorting algorithm, additional ELA measures should be calculated from the obtained sequences, e.g., adaptations of the correlation length [8] and correlation function of a random walk [7] to the continuous space. An alternative to using LHD sampling and nearest neighbor sorting is to develop a random walk algorithm that produces an unbiased sample, i.e., uniformly distributed over the input space. This sample can be used to extract the additional ELA measures described in Section V-C.

Although random sorting is computationally simpler, we demonstrated that it negatively affects the values of $H^{R}_{\max}$ and M^{R}_0. In particular, M^{R}_0 produces the same value regardless of the function, instance, or sample size, and the range of $H^{R}_{\max}$ is smaller than $H^{NN}_{\max}$. This suggests that the random elements in the sequence supersede the information extracted from the function. However, $H^{R}_{\max}$ and ϵ^{R}_S have sufficient information to produce a classification with good accuracy. The partial information content provides little if any additional information because M^{NN}_0 has little relevance in the classification model, M^{R}_0 does not have valuable information, and $\epsilon^{NN}_{0.5}$ $\epsilon^{R}_{0.5}$ are collinear with ϵ^{R}_S. Therefore, we propose to remove the computation of $M(\epsilon)$ from ICoFiS to reduce the overall computational cost.

The results demonstrated that the ϵ^{NN}_S and ϵ^{R}_S strongly influence the accuracy of the classification model, and they are almost perfectly correlated with the entropy of the $\mathcal{Y}$-distribution, $H(\mathbf{Y})$. This result implies that $H(\mathbf{Y})$ is related to $\frac{\Delta y}{\|\Delta \mathbf{x}\|}$. Since the entropy is a measure of the difference between fitness values found in the landscape, and ϵ_S is an estimation of the largest difference of fitness found in the landscape, it is evident that this two measures are related somehow. However, we do not have a formal proof to generalize this empirical observation beyond the set of functions examined in this paper. Considering that the entropy is estimated using an algorithm of $\mathcal{O}(p \log p)$ complexity [33], it is computationally faster to calculate the entropy than ϵ_S.

After we investigated the validity of our measures, we evaluated how these measures can be related to the algorithm selection problem [2], [28]. We do not claim to have the ultimate solution to this problem; however, we provide some directions that can be used to solve this problem based on our experiments. Several complementary algorithms under controlled conditions must be evaluated. Also, more measures than the ones presented in this paper should be investigated. There must be a balance between measure extraction and problem solving, which can be seen as a balance between exploration and exploitation. The improvements in ICoFiS help to shift the balance toward problem solving, as it is no longer necessary to take another sample to calculate the additional ELA measures. A final step is to select the most appropriate machine learning model (i.e., regression, classification, cost sensitive, or other) to map the relationship between ELA measures and Algorithms.

VIII. Conclusion

In this paper, we have proposed a new data-driven method based on the information content for the exploratory landscape analysis of continuous space problems. ICoFiS incorporates two significant enhancements compared to the previous adaptations [10], [11]. In ICoFiS, we include the distance between two observations in the analysis, and we use LHD sampling to generate an unbiased sample and a sorting algorithm to create the sequence of observations. In addition, we have demonstrated that the measures extracted with ICoFiS are useful to classify a group of functions into categories. This means that ICoFiS can be part of a system that uses several ELA methods to identify and classify functions, as well as to automatically select the most efficient algorithm [1], [4], [5]. However, some open questions remain. For example, it would be useful to have a formal explanation of the correlation between ϵ_S and the entropy of the $\mathcal{Y}$-distribution, and a deeper understanding of the results provided by $H_{\max}$. This work would improve the interpretability of the measures resulting from ICoFiS, and allow us to generalize the empirical observations made in this paper beyond the set of benchmark functions that we have employed.

Our results highlight the importance of thoroughly testing the usefulness of a measure, and the relationships that exist with other ELA methods. We have demonstrated the importance of comparing as many ELA methods as possible, taking into account the different sources of uncertainty in the experimental procedures. Further work should focus on finding faster and simpler sequencing algorithms, ideally guaranteeing a uniformly distributed sample. In addition,

new measures that use sequences may provide new information that the methods evaluated in this paper do not capture. We believe that the evaluation of each new method should include classification experiments as presented in this paper.

ACKNOWLEDGMENTS

The authors would like to thank M. Gallagher, R. Morgan, X. Li, and K. Steer for their valuable comments.

REFERENCES

[1] O. Mersmann, B. Bischl, H. Trautmann, M. Preuss, C. Weihs, and G. Rudolph, "Exploratory landscape analysis," in *Proc. 13th GECCO*, 2011, pp. 829–836.
[2] K. Smith-Miles, "Cross-disciplinary perspectives on meta-learning for algorithm selection," *ACM Comput. Surv.*, vol. 41, no. 1, pp. 6:1–6:25, Jan. 2009.
[3] K. Leyton-Brown, E. Nudelman, and Y. Shoham, "Empirical hardness models: Methodology and a case study on combinatorial auctions," *J. ACM*, vol. 56, pp. 22:1–22:52, Jul. 2009.
[4] B. Bischl, O. Mersmann, H. Trautmann, and M. Preuß, "Algorithm selection based on exploratory landscape analysis and cost-sensitive learning," in *Proc. 14th GECCO*, 2012, pp. 313–320.
[5] M. Muñoz, M. Kirley, and S. Halgamuge, "A meta-learning prediction model of algorithm performance for continuous optimization problems," in *Proc. PPSN XII*, LNCS 7941. 2012, pp. 226–235.
[6] E. Pitzer and M. Affenzeller, "A comprehensive survey on fitness landscape analysis," in *Recent Advances in Intelligent Engineering Systems (Studies Comput. Intell.)*, vol. 378. Berlin/Heidelberg, Germany: Springer, 2012, pp. 161–191.
[7] E. Weinberger, "Correlated and uncorrelated fitness landscapes and how to tell the difference," *Biol. Cybern.*, vol. 63, no. 5, pp. 325–336, Sep. 1990.
[8] P. Stadler, "Landscapes and their correlation functions," *J. Math. Chem.*, vol. 20, no. 1, pp. 1–45, 1996.
[9] V. Vassilev, T. Fogarty, and J. Miller, "Information characteristics and the structure of landscapes," *Evol. Comput.*, vol. 8, no. 1, pp. 31–60, 2000.
[10] K. Steer, A. Wirth, and S. Halgamuge, "Information theoretic classification of problems for metaheuristics," in *Proc. Simulated Evol. Learn.*, LNCS 5361. 2008, pp. 319–328.
[11] K. Malan and A. Engelbrecht, "Quantifying ruggedness of continuous landscapes using entropy," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, May 2009, pp. 1440–1447.
[12] N. Hansen, A. Auger, S. Finck, and R. Ros, "Real-parameter black-box optimization benchmarking BBOB-2010: Experimental setup," INRIA, Paris, France, Tech. Rep. RR-7215, Sep. 2010.
[13] T. Jones and S. Forrest, "Fitness distance correlation as a measure of problem difficulty for genetic algorithms," in *Proc. 6th Int. Conf. Genetic Algorithms*, 1995, pp. 184–192.
[14] M. Lunacek and D. Whitley, "The dispersion metric and the CMA evolution strategy," in *Proc. 8th Annu. Conf. Genetic Evol. Comput.*, 2006, pp. 477–484.
[15] J. Marin, "How landscape ruggedness influences the performance of real-coded algorithms: A comparative study," *Soft Comput.*, vol. 16, no. 4, pp. 683–698, 2012.
[16] D. Seo and B. Moon, "An information-theoretic analysis on the interactions of variables in combinatorial optimization problems," *Evol. Comput.*, vol. 15, no. 2, pp. 169–198, 2007.
[17] N. Hansen, A. Auger, R. Ros, S. Finck, and P. Pošík, "Comparing results of 31 algorithms from the black-box optimization benchmarking bbob-2009," in *Proc. Conf. Genetic Evol. Comput.*, 2011, pp. 1689–1696.
[18] V. Vassilev, T. Fogarty, and J. Miller, "Smoothness, ruggedness and neutrality of fitness landscapes: From theory to application," in *Advances in Evolutionary Computing*. New York, NY, USA: Springer, 2003, pp. 3–44.
[19] B. Zadrozny, "Learning and evaluating classifiers under sample selection bias," in *Proc. 21st Int. Conf. Mach. Learn.*, 2004, p. 114.
[20] C. Cortes, M. Mohri, M. Riley, and A. Rostamizadeh, "Sample selection bias correction theory," in *Algorithmic Learning Theory*, LNCS 5254. Berlin/Heidelberg, Germany: Springer, 2008, pp. 38–53.
[21] R. Morgan and M. Gallagher, "Length scale for characterising continuous optimization problems," in *Proc. PPSN XII)*, 2012, pp. 407–416.
[22] Y. Benjamini and D. Yekutieli, "The control of the false discovery rate in multiple testing under dependency," *Ann. Stat.*, vol. 29, no. 4, pp. 1165–1188, 2001.
[23] D. Groppe, T. Urbach, and M. Kutas, "Mass univariate analysis of event-related brain potentials/fields I: A critical tutorial review," *Psychophysiology*, vol. 48, no. 12, pp. 1711–1725, 2011.
[24] D. Hinkle, W. Wiersma, and S. Jurs, *Applied Statistics for the Behavioral Sciences*. Boston, MA, USA: Houghton Mifflin, 2003.
[25] C. Müller and I. Sbalzarini, "Global characterization of the CEC 2005 fitness landscapes using fitness-distance analysis," in *Applications of Evolutionary Computation*, LNCS 6624. Berlin/Heidelberg, Germany: Springer, 2011, pp. 294–303.
[26] M. Graff and R. Poli, "Practical performance models of algorithms in evolutionary program induction and other domains," *Artif. Intell.*, vol. 174, no. 15, pp. 1254–1276, Oct. 2010.
[27] B. Efron and R. Tibshirani, *An Introduction to the Bootstrap*. London, U.K.: Chapman & Hall, 1993.
[28] J. Rice, "The algorithm selection problem," in *Advances in Computers*, vol. 15. Amsterdam, The Netherlands: Elsevier, 1976, pp. 65–118.
[29] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*. Berlin/Heidelberg, Germany: Springer, 2009.
[30] D. Wolpert and W. Macready, "No free lunch theorems for optimization," *IEEE Trans. Evol. Comput.*, vol. 1, no. 1, pp. 67–82, Apr. 1997.
[31] R. Morgan and M. Gallagher, "Sampling techniques and distance metrics in high dimensional continuous landscape analysis: Limitations and improvements," *IEEE Trans. Evol. Comput.*, vol. 18, no. 3, pp. 456–461, Jun. 2014.
[32] O. Mersmann, M. Preuss, and H. Trautmann, "Benchmarking evolutionary algorithms: Towards exploratory landscape analysis," in *Proc. PPSN XI*, LNCS 6238. 2010, pp. 73–82.
[33] D. Stowell and M. Plumbley, "Fast multidimensional entropy estimation by k-d partitioning," *IEEE Signal Process. Lett.*, vol. 16, no. 6, pp. 537–540, Jun. 2009.

Mario A. Muñoz received the B.Eng. and M.Eng. degrees in electronics engineering from Universidad del Valle, Cali, Colombia, in 2005 and 2008, respectively, and completed in 2014 the Ph.D. degree with The University of Melbourne, Melbourne, Australia, focusing on the fitness landscape analysis of continuous optimization problems and its application in empirical models of algorithm performance.

His research interests include randomized heuristics for optimization, intelligent control systems, and applications of machine learning in biomechanics.

Michael Kirley received the B.Ed. degree from Deakin University, Waurn Ponds, Australia, in 1988 and the Ph.D. degree from Charles Sturt University, Melbourne, Australia, in 2003.

He is a Senior Lecturer with the Department of Computing and Information Systems, The University of Melbourne, Melbourne, Australia. He has published over 100 papers. His research interests include theory and application of evolutionary computation, evolutionary game theory, multiagent systems, and complex systems science.

Saman K. Halgamuge (SM'13) received the B.Sc.Eng. degree in electronic and telecommunication engineering from University of Moratuwa, Moratuwa, Sri Lanka, in 1984 and the Dr.-Ing and Dipl.-Ing degrees in electrical engineering from TU Darmstadt, Darmstadt, Germany, in 1995 and 1990, respectively.

He is a Professor with the Department of Mechanical Engineering and the Associate Dean (International) of the Melbourne School of Engineering, The University of Melbourne, Melbourne, Australia. He has published over 250 papers. His research interests include big data, mechatronics, bioinformatics, bio-inspired computing, and optimization algorithms.