Landscape-aware performance prediction for evolutionary multi-objective optimization

Arnaud Liefooghe, Fabio Daolio, Sébastien Verel, Bilel Derbel, Hernan
Aguirre, Kiyoshi Tanaka

HAL Id: hal-02294201

https://hal.science/hal-02294201

Submitted on 2 Dec 2020

Landscape-Aware Performance Prediction for Evolutionary Multi-objective Optimization

Arnaud Liefooghe, Fabio Daolio, Sébastien Verel, Bilel Derbel, Hernán Aguirre, Kiyoshi Tanaka

Abstract—We expose and contrast the impact of landscape characteristics on the performance of search heuristics for black-box multi-objective combinatorial optimization problems. A sound and concise summary of features characterizing the structure of an arbitrary problem instance is identified and related to the expected performance of global and local dominance-based multi-objective optimization algorithms. We provide a critical review of existing features tailored to multi-objective combinatorial optimization problems, and we propose additional ones that do not require any global knowledge from the landscape, making them suitable for large-size problem instances. Their intercorrelation and their association with algorithm performance are also analyzed. This allows us to assess the individual and the joint effect of problem features on algorithm performance, and to highlight the main difficulties encountered by such search heuristics. By providing effective tools for multi-objective landscape analysis, we highlight that multiple features are required to capture problem difficulty, and we provide further insights into the importance of ruggedness and multimodality to characterize multi-objective combinatorial landscapes.

Index Terms—Evolutionary multi-objective optimization, black-box combinatorial optimization, problem difficulty and landscape analysis, feature-based performance prediction.

I. INTRODUCTION

E VOLUTIONARY Multi-objective Optimization (EMO) algorithms and other multi-objective randomized search heuristics have to face a lot of difficulties. Apart from the typical issues encountered in single-objective optimization, multi-objective landscapes exhibit additional challenges such as the identification of multiple (Pareto) optimal solutions, corresponding to different tradeoffs between the objectives. A large number of EMO algorithms has been designed in the last thirty years, ranging from dominance-based approaches to scalarization- and indicator-based refinements; see e.g. [1], [2], [3]. Despite the significant progress in recent years, we argue that most EMO algorithms continue to be designed on the basis of intuition, that they require a significant effort to be tailored to a particular optimization scenario, and that there is a lack of understanding about what makes them efficient or not when solving a particular problem instance.

In single-objective optimization, however, landscape analysis has emerged as a valuable set of tools to characterize

A. Liefooghe and B. Derbel are with Univ. Lille, CNRS, Centrale Lille, UMR 9189 – CRIStAL, Inria Lille–Nord Europe, F-59000 Lille, France (e-mails: arnaud.liefooghe@univ-lille.fr, bilel.derbel@univ-lille.fr).

F. Daolio is with ASOS.com, London, UK (e-mail: fabio.daolio@asos.com).

S. Verel is with Univ. Littoral Côte d'Opale, LISIC, F-62100 Calais, France (e-mail: verel@lisic.univ-littoral.fr).

H. Aguirre and K. Tanaka are with Shinshu University, Faculty of Engineering, Nagano, Japan (e-mails: ahernan@shinshu-u.ac.jp, ktanaka@shinshu-u.ac.jp).

problem difficulty [4]. Contrary to problem-specific structural properties such as the average vertex degree in the minimum vertex cover problem [5] or the maximum cost between two cities in the traveling salesman problem [6], landscape analysis aims at designing general-purpose features that do *not* depend on a specific problem class or problem domain. Instead, it tries to characterize the topology of black-box problems in the eye of the challenges that stochastic local search algorithms have to face when tackling them. Of particular interest is the number and distribution of local optima in the landscape, i.e. multimodality and ruggedness [7], [8], [9], [10], [11]. These features are empirically related to instance hardness and algorithm efficiency, and provide significant insights into the interplay between the problem structure and the behavior of search algorithms and their working components. Pioneering works on multi-objective landscape analysis include Knowles and Corne [12], Paquete and Stützle [13], [14], Garrett and Dasgupta [15], [16], as well as the authors own previous works [17], [18], [19], [20]. We build upon those by considering general-purpose problem features defined therein. In addition, we derive new landscape features that our analysis reveals as highly impactful for multi-objective search.

The purpose of designing features to characterize search difficulty is two-fold: (i) gathering a fundamental understanding of optimization problems and algorithms, eventually leading to a better algorithm design, and (ii) automatically predicting performance or selecting algorithm based on relevant features. More particularly, feature-based performance prediction consists of modeling the expected runtime or solution quality of a given algorithm applied to a problem instance exhibiting particular features. This, in turn, might also lead to algorithm selection [21], [22] and configuration [23], [24], where the best-performing algorithm or configuration is to be selected from a set of competitors. This issue is not specific to optimization, and is known as the algorithm selection problem [25]. A statistical or machine learning regression or classification model is constructed by means of extensive experiments on a training set of instances, and this model is later used to predict the performance or to select between algorithms for previously-unseen instances. This research area has received a growing attention in recent years, mainly by relying on features that require a specific domain knowledge from the target combinatorial optimization problem; see e.g. [6], [26], [27]. Few exceptions can be found in [28], [29], [30], [31], where the selection among an algorithm portfolio is performed using general-purpose landscape features related to local optimality, ruggedness and multimodality. Research in this line can also be found for black-box continuous

single-objective optimization [32], [33], [34]. However, to our knowledge, black-box landscape features have never been used for performance prediction or algorithm recommendation in the context of multi-objective combinatorial optimization. Although the statistical and machine learning models used in the single-objective case can be applied, multi-objective landscape features need to be carefully designed and analyzed, since existing single-objective features are not relevant for multi-objective optimization.

This is precisely the purpose of the current study. Particularly, we first review and extend general-purpose features to characterize the different facets of difficulty encountered in multi-objective combinatorial optimization. Features include problem descriptors, as the solution and objective space dimensions, *global* measures, that require the knowledge of all or part of the solution space, and *local* measures, that are computed from an affordable sample of solutions. Then, we analyze features interrelation as well as their impact on the performance of two canonical EMO algorithms, namely the global simple evolutionary multi-objective optimizer [35] and the Pareto local search [13]. We selected a *global* and a *local* simple elitist dominance-based EMO algorithms, respectively, for the sake of clarifying the understanding of core EMO algorithm components. Experiments are conducted on a family of multimodal pseudo-boolean optimization problems known as multi-objective `nk`-landscapes with objective correlation [18]. By paying a particular attention to the computational cost induced by these features, we finally analyze their ability to predict algorithm performance and to select among a small algorithm portfolio. A sound statistical analysis allows us to highlight the main difficulties that dominance-based EMO algorithms have to face, as well as the main differences induced by global and local EMO search approaches.

The paper is organized as follows. In Section II, we present the main concepts from multi-objective combinatorial optimization, and we introduce the multi-objective algorithms and multi-objective `nk`-landscapes considered in the paper. In Section III, we identify a substantial number of existing and original features that characterize black-box multi-objective landscapes. In Section IV, based on a set of small-size multi-objective `nk`-landscapes, we analyze the correlation among features, and we measure their ability to predict algorithm performance as well as their impact on search efficiency. In Section V, we extend our analysis to large-size instances by focusing on local features, i.e. features that can be computed efficiently. In Section VI, we experiment with feature-based algorithm selection for both multi-objective `nk`-landscapes and quadratic assignment problems. In the last section, we conclude the paper and discuss further research.

II. MULTI-OBJECTIVE COMBINATORIAL OPTIMIZATION

A. Definitions

Let us consider a black-box objective function vector $f \colon X \mapsto Z$ to be maximized. Each solution from the solution space $x \in X$ maps to a vector in the objective space $z \in Z$, with $Z \subseteq \mathbb{R}^m$, such that $z = f(x)$. In multi-objective combinatorial optimization, the solution space X is a discrete set. We here assume that $X := \{0, 1\}^n$, where n is the problem size, i.e. the number of binary (zero-one) variables. Given two objective vectors $z, z' \in Z$, z is dominated by z' iff for all $i \in \{1, \ldots, m\}$ $z_i \leqslant z'_i$, and there is a $j \in \{1, \ldots, m\}$ such that $z_j < z'_j$. Similarly, given two solutions $x, x' \in X$, x is dominated by x' iff $f(x)$ is dominated by $f(x')$. An objective vector $z^\star \in Z$ is non-dominated if there does not exist any $z \in Z$ such that $z^\star$ is dominated by z. A solution $x^\star \in X$ is Pareto optimal (PO), or non-dominated, if $f(x)$ is non-dominated. The set of PO solutions is the Pareto set (PS); its mapping in the objective space is the Pareto front (PF). One of the main challenges in multi-objective optimization is to identify the PS, or a good approximation of it for large-size and difficult black-box problems.

B. Algorithms

We consider two randomized multi-objective search heuristics: (i) the Global Simple Evolutionary Multi-objective Optimizer (G-SEMO) [35], a simple elitist steady-state *global* EMO algorithm; and (ii) Pareto local search (PLS) [13], a population-based multi-objective *local* search. Both algorithms maintain an *unbounded* archive A of mutually non-dominated solutions. This archive is initialized with one random solution from the solution space. At each iteration, one solution is selected at random from the archive $x \in A$. In G-SEMO, each binary variable from x is independently flipped with a rate of $1/n$ in order to produce an offspring solution x'. The archive is then updated by keeping the non-dominated solutions from $A \cup \{x'\}$. In PLS, the solutions located in the neighborhood of x are evaluated. Let $\mathcal{N}(x)$ be the set of solutions located at a Hamming distance 1. The non-dominated solutions from $A \cup \mathcal{N}(x)$ are stored in the archive, and the current solution x is tagged as *visited* in order to avoid a useless revaluation of its neighborhood. This process is iterated until a stopping condition is satisfied. While G-SEMO does not have any explicit stopping rule, PLS naturally stops once all solutions from the archive are tagged as *visited*. For this reason, we consider a simple iterated version of PLS (I-PLS), that restarts from a solution randomly chosen from the archive and perturbed by stochastic mutation [36]. While PLS is based on the exploration of the whole 1-bit-flip neighborhood from x, G-SEMO rather uses an ergodic operator, i.e. an independent bit-flip mutation. Hence, every iteration has a non-zero probability of reaching any solution from the solution space. This makes G-SEMO a *global* optimizer, in contrast with a *local* optimizer as PLS.

C. Pseudo-Boolean Multi-objective Benchmark Instances

We consider ρ`mnk`-landscapes [18] as a problem-independent model used for constructing multi-objective multimodal landscapes with objective correlation. They extend single-objective `nk`-landscapes [8] and multi-objective `nk`-landscapes with independent objectives [17]. Candidate solutions are binary strings of size n. The objective function vector $f = (f_1, \ldots, f_i, \ldots, f_m)$ is defined as $f \colon \{0, 1\}^n \mapsto [0, 1]^m$ such that each objective f_i is to be maximized. As in the single-objective case, the objective value $f_i(x)$ of a

solution $x = (x_1, \ldots, x_j, \ldots, x_n)$ is an average value of the individual contributions associated with each variable x_j. Given objective f_i, $i \in \{1, \ldots, m\}$, and each variable x_j, $j \in \{1, \ldots, n\}$, a component function $f_{ij} \colon \{0, 1\}^{k+1} \mapsto [0, 1]$ assigns a real-valued contribution for every combination of x_j and its k *epistatic interactions* $\{x_{j_1}, \ldots, x_{j_k}\}$. These f_{ij}-values are uniformly distributed in $[0, 1]$. Thus, the individual contribution of a variable x_j depends on its value and on the values of $k < n$ variables $\{x_{j_1}, \ldots, x_{j_k}\}$ other than x_j. The problem can be formalized as follows:

$$\max \quad f_i(x) = \frac{1}{n} \sum_{j=1}^{n} f_{ij}(x_j, x_{j_1}, \ldots, x_{j_k}) \quad i \in \{1, \ldots, m\}$$
$$\text{s.t.} \quad x_j \in \{0, 1\} \qquad\qquad\qquad\quad j \in \{1, \ldots, n\}$$

In this work, the epistatic interactions, i.e. the k variables that influence the contribution of x_j, are set uniformly at random among the $(n - 1)$ variables other than x_j, following the random neighborhood model from [8]. By increasing the number of epistatic interactions k from 0 to $(n - 1)$, problem instances can be gradually tuned from smooth to rugged. In ρmnk-landscapes, f_{ij}-values additionally follow a multivariate uniform distribution of dimension m, defined by an $m \times m$ positive-definite symmetric covariance matrix (c_{pq}) such that $c_{pp} = 1$ and $c_{pq} = \rho$ for all $p, q \in \{1, \ldots, m\}$ with $p \neq q$, where $\rho > \frac{-1}{m-1}$ defines the correlation among the objectives; see [18] for details. The positive (respectively, negative) objective correlation ρ decreases (respectively, increases) the degree of conflict between the different objective function values. Notice that the correlation coefficient ρ is the same between all pairs of objectives, and the same epistatic degree k and epistatic interactions are set for all the objectives.

III. Characterizing Multi-objective Landscapes

In this section, we present the set of multi-objective landscape features considered in our analysis. We start with *global* features, that, in order to be computed, require the knowledge of all solutions and/or Pareto optima. This makes them impractical for performance prediction and algorithm selection. However, we decided to include them in order to measure and understand their impact on search performance. Next, we introduce a number of *local* features, which are based on a reasonable subset of solutions sampled during random and adaptive walks, making them affordable in practice for performance prediction. The whole set of features is listed in Table I, together with the parameters of ρmnk-landscapes described above, i.e. ρ, m, n, and k.

As in single-objective optimization, we define a multi-objective landscape as a triplet $(X, \mathcal{N}, f)$, where X is a solution space, $\mathcal{N} \colon X \mapsto 2^X$ is a neighborhood relation, and $f \colon X \mapsto Z$ is a (black-box) objective function vector.

A. Global Features

Let us start with the subset of global features illustrated in Fig.1. In multi-objective combinatorial optimization, the number of Pareto optimal (PO) solutions is considered as an important aspect of difficulty. Generally speaking, the larger the PS, the smaller the chance to identify all PO

solutions in an efficient manner. In that sense, most multi-objective combinatorial optimization problems are known to be *intractable*, i.e. the number of PO solutions typically grows exponentially with the problem size [37]. As such, the proportion of PO solutions in the solution space (#po) is one of the most obvious facet to characterize problem difficulty; see e.g. [12], [15], [17]. For ρmnk-landscapes, #po is known to grow exponentially with the number of objectives and with their degree of conflict [18]. PO solutions can further be classified into two categories: supported and non-supported solutions. A supported solution is an optimal solution of a weighted-sum aggregation of the objectives, and is mapping to an objective vector that is located on the convex hull of the PF [37]. Although dominance-based approaches are considered in our analysis, the proportion of supported solutions in the PS (#supp) might impact the general convexity of the PF, as illustrated by previous studies on multi-objective landscape analysis [12]. Similarly, the hypervolume (hv) covered by the (exact) PF is shown to relate to the expected performance of EMO algorithms [17]. The hypervolume is one of the few recommended indicators for comparing solution-sets in EMO. It gives the portion of the objective space that is dominated by a solution-set [38].

Other relevant characteristics from the PS deal with the distance and connectedness between PO solutions. Knowles and Corne [12] study the maximal distance, in the solution space, between any pair of PO solutions (podist_max). They denote this as the *diameter* of the PS. For ρmnk-landscapes, the distance measure is taken as the Hamming distance between binary strings, which is directly related to the bit-flip neighborhood operator. Similarly, the average distance between PO solutions (podist_avg) can also be taken into account [19]. Another measure capturing the dispersion of solutions is the entropy of the PS [12], here measured as the entropy of (binary) variables from PO solutions (po_ent). Extensions of the fitness-distance correlation, a widely acknowledged landscape measure [39], to multi-objective optimization is discussed in [12], [15]. We here consider the correlation between the (Hamming) distance between PO solutions and their Manhattan distance in the objective space [12]. Another important property of the PS topology is *connectedness* [40], [41]. The PS is connected if all PO solutions are connected with respect to a given neighborhood structure. This makes it possible for local search to identify the whole PS by starting with one PO solution. Let us define a graph such that each node corresponds to a PO solution, and there is an edge between two nodes if the corresponding solutions are neighbors in the landscape. Arguing that the degree of connectedness impacts the performance of multi-objective local search [14], [42], we here consider the following landscape features, illustrated in Fig. 2: the proportion of connected components in this Pareto graph (#cc) [14], the proportion of isolated nodes (#sing) [14], the proportional size of the largest connected component (#lcc) [18], [42], as well as the average distance between pairs of nodes (lcc_dist) and the proportion of hypervolume covered by the largest connected component from the Pareto graph (lcc_hv).

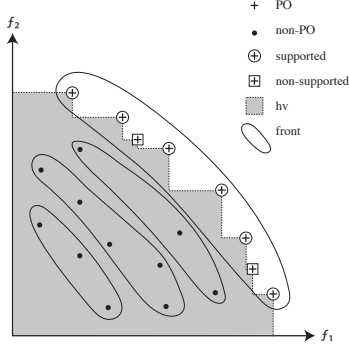The characteristics of the PS and the PF, however, are

Fig. 1. Illustration of global features extracted from the PS, the PF, and the solution space.
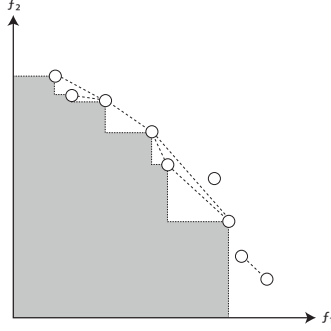

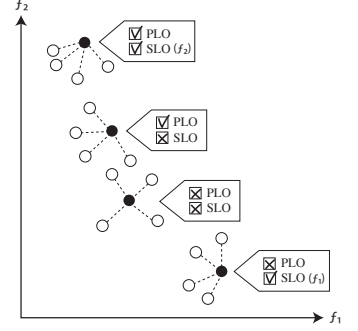Fig. 2. Illustration of global features extracted from the connectedness properties of the PS.

Fig. 3. Illustration of Pareto local optima (PLO) and single-objective local optima (SLO).

not the sole factors that impact the performance of EMO algorithms. In [16], [17], the authors analyze how the landscape affects the number of non-dominated fronts, and how this relates to search performance. As illustrated in Fig. 1, the whole set of solutions from the search space is divided into different layers of mutually non-dominated solutions, following the principles of non-dominated sorting used, e.g., in NSGA-II [1]. To cater for this, we measure both the proportion of non-dominated fronts in the solution space (#fronts) [16], [17], and the entropy of the non-dominated front's size distribution (front_ent). Finally, one of the main landscape features in single-objective optimization is the number of local optima [8]. Although multimodality is still largely overlooked in the multi-objective optimization literature, where the number of objectives is seen as the main source of difficulty, few recent studies have revealed its impact on multi-objective search performance [13], [18], [20], [11]. Following [13], we define a Pareto local optimum (PLO) as a solution $x \in X$ for which there does not exist any neighboring solution $x' \in \mathcal{N}(x)$ such that x is dominated by x', and we measure the proportion of PLO in the solution space (#plo). Additionally, we also consider the average number of single-objective local optima (SLO) with respect to each separate objective function, proportional to the size of the solution space (#slo_avg). In other words, #slo_avg corresponds to the proportion of local optima per objective, all m values (i.e. one per objective) being averaged. The definitions of PLO and SLO are illustrated in Fig. 3. We expect #slo_avg to increase with the number of variable interactions k, as in single-objective nk-landscapes [8]. However, we conjecture that #plo is not only affected by k, but also by the number of objectives and their degree of conflict. Both features might then capture different facets of multi-objective multimodality.

B. Local Features

Unfortunately, computing the global features introduced above requires the solution space, or the PS, to be exhaustively enumerated, which makes them impractical. Therefore, we consider local features, computed from the neighborhood of a sample of solutions, which makes them relevant for

performance prediction. In the following, we introduce two sampling strategies and a number of landscape measures. We simply consider a local feature as a combination of both.

1) Sampling: In single-objective landscape analysis, sampling is often performed by means of a *walk* over the landscape. A walk is an ordered sequence of solutions $(x_0, x_1, \ldots, x_\ell)$ such that $x_0 \in X$, and $x_t \in \mathcal{N}(x_{t-1})$ for all $t \in \{1, \ldots, \ell\}$ [7], [8].

During a *random walk*, there is no particular criterion to pick the neighboring solution at each step; i.e. a random neighbor is selected. In the single-objective case, the first autocorrelation coefficient of (scalar) fitness values encountered during the random walk characterizes the *ruggedness* of the landscape [7], [43]: the larger this coefficient, the smoother the landscape. To accommodate the multi-objective nature of the landscape, different autocorrelation measures will be discussed below. In the case of a random walk, the length of the walk ℓ is a parameter that must be provided beforehand. The longer the length, the better the estimation.

By contrast, during an *adaptive walk*, an improving neighbor is selected at each step, as a conventional hill climber would do. In this case, the length ℓ is the number of steps performed until the walk falls into a local optimum. This length is used as an estimator of the diameter of local optima's basins of attraction: assuming isotropy in the search space, the longer the length, the larger the basins size, hence the lower the number of local optima [8]. Multiple adaptive walks are typically performed to improve the estimation.

A random walk does not require any adaptation to the multi-objective case, except for the measure used to estimate the correlation coefficient, detailed next. As for the adaptive walk, we consider a very basic single solution-based multi-objective Pareto hill climber (PHC) [18]. The PHC is initialized with a random solution. At each iteration, the current solution is replaced by a random dominating neighbor until it falls into a PLO. The considered random and adaptive walks are illustrated in Figs. 4 and 5, respectively.

2) Measures: Given an ordered sequence of solutions collected along a walk, we consider the following measures. For each solution from the sample, we explore its neighborhood, and we measure the proportion of dominated (#inf),
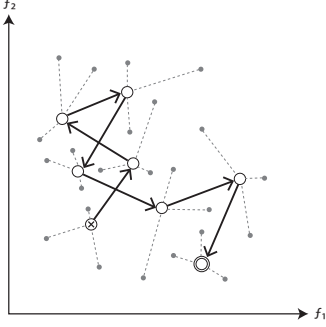
Fig. 4. Illustration of a multi-objective random walk in the objective space. In this example, the walk length is set to $\ell = 7$.
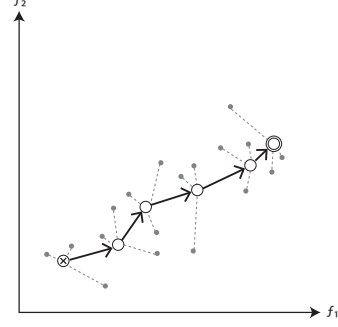

Fig. 5. Illustration of a multi-objective adaptive walk (PHC) in the objective space. In this example, the walk performs $\ell = 5$ until it falls into a PLO.
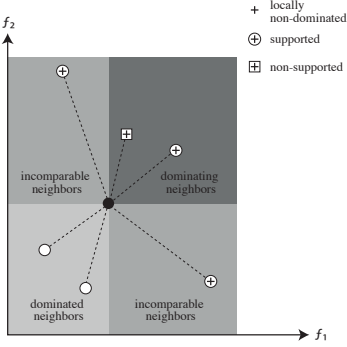

Fig. 6. Illustration of local dominance measures collected along random and adaptive walks.
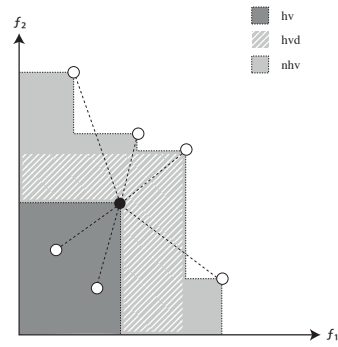

Fig. 7. Illustration of local hypervolume measures collected along random and adaptive walks.

dominating (`#sup`), and incomparable (`#inc`) neighbors, as illustrated in Fig. 6. We also consider the proportion of non-dominated solutions in its neighborhood (`#lnd`), as well as the proportion of supported solutions therein (`#lsupp`). In Fig. 7, we illustrate some measures based on hypervolume: the average hypervolume covered by each neighbor (`hv`), the average difference between the hypervolume covered by each neighbor and the one covered by the current solution (`hvd`), and the hypervolume covered by the whole neighborhood (`nhv`). The notions of dominance and hypervolume improvement that can be reached by a solution's neighborhood can be seen as measures of evolvability [44] for multi-objective optimization.

For samples collected by means of a random walk, we compute both the average value as well as the first autocorrelation coefficient of the measures reported above. Let us consider, for instance, the `hv` measure. When there is a strong correlation between the hypervolume of neighboring solutions observed at two consecutive steps of the random walk, we argue that it tends to be easier to improve locally by means of neighborhood exploration. On the contrary, when there is no correlation between the hypervolume of neighboring solutions, it is likely harder to improve locally. As such, the corresponding feature might characterize a facet of difficulty for multi-objective landscapes. We also use the random walk

sample to estimate the degree of correlation between the objectives (`f_cor_rws`). The latter is expected to estimate ρ for ρmnk-landscapes. For adaptive walks, we simply compute average values for each measure, as well as walks length (`length_aws`). In [18], `length_aws` is shown to be a good estimator for `#plo`.

IV. LANDSCAPE FEATURES VS. ALGORITHM PERFORMANCE ON SMALL INSTANCES

A. Experimental Setup

We consider small-size ρmnk-landscapes with a problem size $n \in \{10, 11, 12, 13, 14, 15, 16\}$ in order to enumerate the solution space exhaustively, as required by global features; a number of variable interactions (epistatic degree) $k \in \{0, 1, 2, 3, 4, 5, 6, 7, 8\}$, from linear to highly rugged landscapes; a number of objectives $m \in \{2, 3, 4, 5\}$, from bi-, to multi- and many-objective instances; and an objective correlation $\rho \in \{-0.8, -0.6, -0.4, -0.2, 0.0, 0.2, 0.4, 0.6, 0.8, 1.0\}$ such that $\rho > -1/(m-1)$. We generate 30 landscapes independently at random for each combination of instance settings. This represents a dataset of $60\,480$ small-size problem instances in total, exhibiting a large span of landscape characteristics. For local features, we perform one random walk of length $\ell = 1\,000$, and 100 independent adaptive walks,

BENCHMARK **PARAMETERS** (4)		
n	number of (binary) variables	
k_n	proportional number of variable interactions (epistatic links) : k/n	
m	number of objectives	
ρ	correlation between the objective values	

GLOBAL FEATURES FROM **FULL ENUMERATION** (16)		
#po	proportion of Pareto optimal (PO) solutions	[12], [15]
#supp	proportion of supported PO solutions	[12]
hv	hypervolume-value of the (exact) Pareto front	[17]
podist_avg	average Hamming distance between PO solutions	[19]
podist_max	maximal Hamming distance between PO solutions (diameter of the Pareto set)	[12]
po_ent	entropy of binary variables from Pareto optimal solutions	[12]
fdc	fitness-distance correlation in the Pareto set (Hamming dist. in sol. space vs. Manhattan dist. in obj. space)	[12], [15]
#cc	proportion of connected components in the Pareto graph	[14]
#sing	proportion of isolated Pareto optimal solutions (singletons) in the Pareto graph	[14]
#lcc	proportional size of the largest connected component in the Pareto graph	[18]
lcc_dist	average Hamming distance between solutions from the largest connected component	here
lcc_hv	proportion of hypervolume covered by the largest connected component	here
#fronts	proportion of non-dominated fronts	[16], [17]
front_ent	entropy of the non-dominated front's size distribution	here
#plo	proportion of Pareto local optimal (PLO) solutions	[13]
#slo_avg	average proportion of single-objective local optimal solutions per objective	here

LOCAL FEATURES FROM RANDOM WALK **SAMPLING** (17)		
#inf_avg_rws	average proportion of neighbors dominated by the current solution	here
#inf_r1_rws	first autocorrelation coefficient of the proportion of neighbors dominated by the current solution	here
#sup_avg_rws	average proportion of neighbors dominating the current solution	here
#sup_r1_rws	first autocorrelation coefficient of the proportion of neighbors dominating the current solution	here
#inc_avg_rws	average proportion of neighbors incomparable to the current solution	here
#inc_r1_rws	first autocorrelation coefficient of the proportion of neighbors incomparable to the current solution	here
#lnd_avg_rws	average proportion of locally non-dominated solutions in the neighborhood	here
#lnd_r1_rws	first autocorrelation coefficient of the proportion of locally non-dominated solutions in the neighborhood	here
#lsupp_avg_rws	average proportion of supported locally non-dominated solutions in the neighborhood	here
#lsupp_r1_rws	first autocorrelation coefficient of the proportion of supported locally non-dominated solutions in the neighborhood	here
hv_avg_rws	average (single) solution's hypervolume-value	here
hv_r1_rws	first autocorrelation coefficient of (single) solution's hypervolume-values	[19]
hvd_avg_rws	average (single) solution's hypervolume difference-value	here
hvd_r1_rws	first autocorrelation coefficient of (single) solution's hypervolume difference-values	[19]
nhv_avg_rws	average neighborhood's hypervolume-value	here
nhv_r1_rws	first autocorrelation coefficient of neighborhood's hypervolume-value	here
f_cor_rws	estimated correlation between the objective values	here

LOCAL FEATURES FROM ADAPTIVE WALK **SAMPLING** (9)		
#inf_avg_aws	average proportion of neighbors dominated by the current solution	here
#sup_avg_aws	average proportion of neighbors dominating the current solution	here
#inc_avg_aws	average proportion of neighbors incomparable to the current solution	here
#lnd_avg_aws	average proportion of locally non-dominated solutions in the neighborhood	here
#lsupp_avg_aws	average proportion of supported locally non-dominated solutions in the neighborhood	here
hv_avg_aws	average (single) solution's hypervolume-value	here
hvd_avg_aws	average (single) solution's hypervolume difference-value	here
nhv_avg_aws	average neighborhood's hypervolume-value	here
length_aws	average length of adaptive walks	[18]

per instance. As in single-objective landscape analysis [8], multiple adaptive walks are performed to account for the stochasticity observed in their length, whereas a single long random walk is performed to obtain a large sample to better estimate the autocorrelation coefficients. For features based on hypervolume, given that all ρmnk-landscape's objectives have a similar range and take their values in $[0, 1]$, we set the reference point to the origin. In terms of algorithms, we perform 30 independent runs of both G-SEMO and I-PLS on each instance. We are interested in the approximation quality found by each algorithm after reaching a maximum budget, here defined as a number of calls to the evaluation function. The stopping condition is set to a fixed budget of 10% of the solution space size, i.e. $\lceil 0.1 \cdot |X| \rceil = \lceil 0.1 \cdot 2^n \rceil$ calls of the evaluation function. This represents a budget of 103 evaluations for n = 10, up to 6 554 evaluations for n = 16.
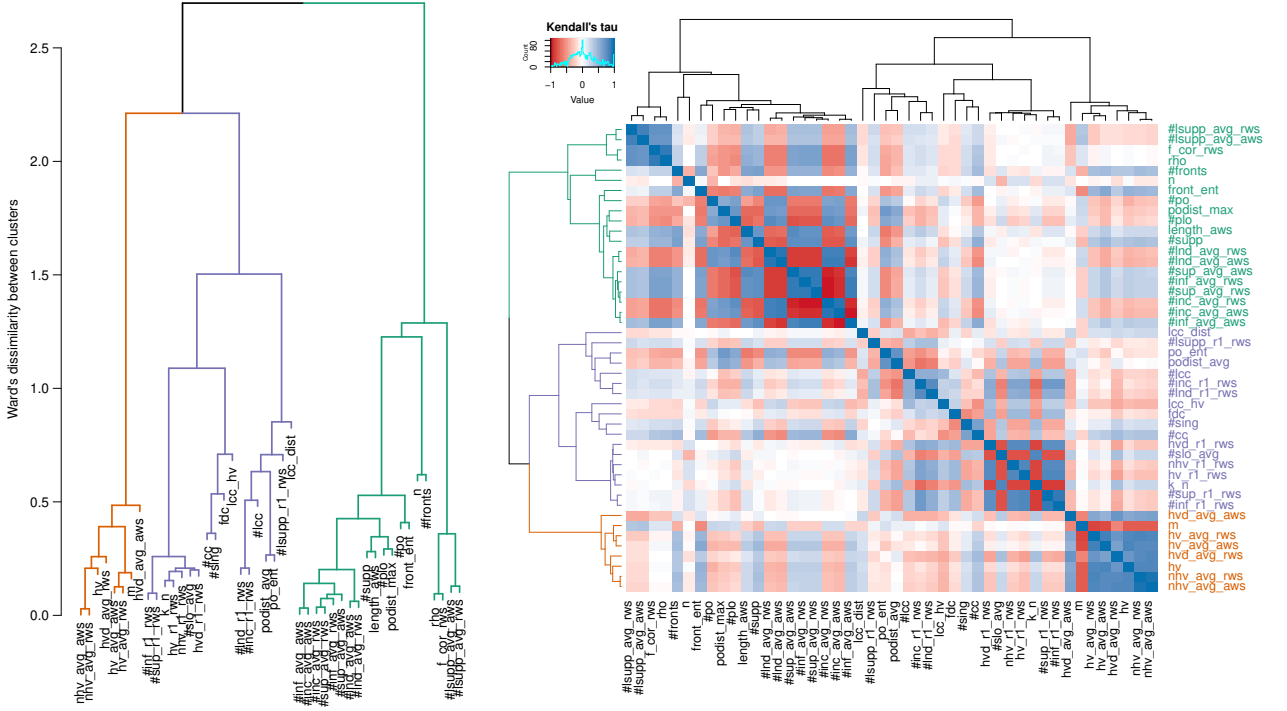
Fig. 8. Features clustering (left) and features association (right) computed over the whole set of small instances. The strength of monotonic association between each pair of features is measured by the Kendall coefficient τ, which is a rank-based non-linear correlation measure. The distance between each pair of features is defined as $1 - |\tau|$. Ward's hierarchical clustering [45] is performed according to such distance (left figure, y-axis): the lower the split, the closer the features. Clustering is also used to reorder rows and columns of the symmetrical correlation matrix [46] (right figure, see color legend in the middle): the darker the color, the higher the strength of association between the corresponding features. By cutting the clustering tree, we can group together the features that are more associated with each one of the benchmark parameters $\{\rho, \text{m}, \text{n}, \text{k\_n}\}$; see branches and row label colors: green for ρ, violet for k_n, and orange for m.

Performance quality is measured in terms of the multiplicative epsilon indicator [38], that is the epsilon approximation ratio to the exact PF.

B. Correlation between Landscape Features

Fig. 8 reports the correlation matrix and a hierarchical clustering of all features, as measured on the complete dataset of small-size instances. This highlights the similarities between features and their main association with either benchmark parameters: it is worth noticing that each cluster contains a benchmark parameter, as well as both global and local features.

1) Cluster associated with ruggedness (violet): All of the eight landscape features from the first autocorrelation coefficient of random walks measures strongly correlate with the proportional number of variable interactions (epistatic links) of ρmnk-landscapes (k_n = k/n). Intuitively, those features are related to the ruggedness of the multi-objective landscape, which generalizes known results from single-objective landscape analysis [7]: the ruggedness of the landscape increases with k_n. As in single-objective optimization, the average number of local optima per objective #slo_avg also correlates to k_n. All the features related with connectedness (#cc, #sing, #lcc, lcc_dist, lcc_hv) belong to this same cluster, together with other features related to the distance be-

tween PO solutions (podist_avg, po_ent, fdc), although the correlation with k_n is lower in this case.

2) Cluster associated with the number of objectives (orange): The features related to hypervolume that do not belong to the previous cluster (associated with ruggedness) are all negatively correlated to the number of objectives (m). Interestingly, features based on average hypervolume measures (hv, hvd, nhv) are closely related to one another, for samples from both random and adaptive walks. This means that the landscape evolvability, in terms of hypervolume, decreases with the objective space dimension, and so does the PF hypervolume.

3) Cluster associated with objective correlation (green): This last cluster contains the highest number of features, all related to the correlation between the objective values (ρ). Note that f_cor_rws is shown to highly correlate with ρ, and can thus be used as an estimator for black-box instances, for which ρ is typically unknown. Objective correlation seems to impact both the shape and the cardinality of the PF (#po, #supp, podist_max). Similarly, local features based on dominance (#inf, #inc, #sup) are close to one another, both for random and adaptive walks. More interestingly, the proportion of Pareto local optima (#plo) and its estimator length_aws both belong to this cluster. Although #slo_avg belongs to the first cluster associated with ruggedness (see above), #plo seems to increase with the degree of conflicts between the

TABLE II
10-FOLD CROSS-VALIDATED REGRESSION ACCURACY OBTAINED ON SMALL INSTANCES FOR DIFFERENT INPUT FEATURES.

Algo.	Set of features	MAE		MSE		R^2		adjusted R^2		Rank
		avg	std	avg	std	avg	std	avg	std	
G-SEMO	all features	**0.007781**	0.000055	**0.000118**	0.000002	0.951609	0.001463	0.951238	0.001474	1
	global features	0.008411	0.000064	0.000142	0.000003	0.943046	0.001665	0.942876	0.001670	2
	local features	0.009113	0.000072	0.000161	0.000003	0.932975	0.001555	0.932663	0.001562	3
	local features (random walk)	0.009284	0.000081	0.000167	0.000003	0.930728	0.001605	0.930510	0.001610	4
	local features (adaptive walk)	0.010241	0.000106	0.000195	0.000004	0.917563	0.002260	0.917399	0.002264	5
	$\{\rho, \mathtt{m}, \mathtt{n}, \mathtt{k\_n}\}$	0.010609	0.000110	0.000215	0.000004	0.911350	0.002436	0.911292	0.002372	6
	$\{\mathtt{m}, \mathtt{n}\}$	0.032150	0.000309	0.001545	0.000025	0.340715	0.011217	0.340497	0.011220	7
I-PLS	all features	**0.008043**	0.000052	**0.000127**	0.000002	0.944367	0.001429	0.943940	0.001440	1
	global features	0.008613	0.000054	0.000149	0.000002	0.936046	0.001479	0.935856	0.001484	2
	local features	0.009297	0.000081	0.000167	0.000003	0.925610	0.001900	0.925264	0.001909	3
	local features (random walk)	0.009485	0.000089	0.000173	0.000004	0.923032	0.001863	0.922789	0.001869	4
	local features (adaptive walk)	0.010336	0.000098	0.000198	0.000004	0.910670	0.002455	0.910493	0.002459	5
	$\{\rho, \mathtt{m}, \mathtt{n}, \mathtt{k\_n}\}$	0.010817	0.000122	0.000223	0.000005	0.901888	0.002803	0.901823	0.002882	6
	$\{\mathtt{m}, \mathtt{n}\}$	0.030523	0.000286	0.001423	0.000023	0.351707	0.009822	0.351493	0.009826	7

objectives. Indeed, the objective correlation directly impacts the probability of dominance: the larger ρ, the smaller the chance to have a dominated or dominating neighbor, and the larger the chance to have an incomparable one, which directly impacts the number of PLO. The problem size $\mathtt{n}$ is also contained in this cluster, although it is only slightly correlated to other features, except for the proportional number of fronts (#fronts).

C. Feature-based Performance Prediction

To investigate the association between instance features and empirical problem hardness, we build a regression model that predicts search performance based on different subsets of input features. More precisely, we predict the multiplicative epsilon indicator value reached by G-SEMO and I-PLS based on: all features, global features, local features, local features based on random walk, local features based on adaptive walk, benchmark parameters, and problem parameters available in a black-box scenario. Given the non-linearity observed in the data, we chose a tree-based regression model: an ensemble of extremely randomized trees [47]. It is a variant of the popular random forest model [48] that differs in the way individual trees are built. While splitting a node, we do not only randomize the choice of input variable, but also the cut-point. Moreover, each tree uses the entire training data, rather than bootstrap replicas. In our experiments, we employ ensembles of 500 unpruned regression trees [49]. The prediction target is the approximation ratio to the exact PF, measured every tenth of the total evaluations budget. That is, we model the search convergence curve with a multi-output regression. The mean square error (MSE), mean absolute error (MAE), coefficient of determination (R^2) and adjusted R^2 of the regression model for different sets of predictors are reported in Table II. A score is the average score over the multiple outputs of a model. The closer MSE and MAE are to 0.0, the better. Conversely, R^2 reaches 1.0 when the predictions are perfect, and would be 0.0 for a constant model that always predicts the global average of the target value, irrespective of the input features. For each measure of accuracy, we report the average value on test and its standard deviation over a 10-fold cross-validation.

A general observation is that the MAE and the MSE are in accordance with each other, as shown by the relative ranking of each subset of features. The rank reflects any significant statistical difference on MAE and MSE over the holdouts of each cross-validation iteration, with respect to a Mann-Whitney statistical test at a significance level of 0.05 with Bonferroni correction for multiple comparisons [50]. In addition, when comparing G-SEMO and I-PLS, we observe almost no difference in the models accuracy. When analyzing the impact of the different subset of features, we can observe a poor performance when using solely $\mathtt{m}$ and $\mathtt{n}$ as input variables. This means that the problem input provided in a black-box scenario, i.e. the solution and objective space dimensions, is not sufficient to explain the performance of G-SEMO and I-PLS. Once we take into account the objective correlation ρ, and more importantly the proportional number of variable interactions $\mathtt{k\_n}$, we observe a significant increase in the model accuracy. For both algorithms, the R^2 exceeds 0.9. In a sense, more than 90% of the variance of search performance between instances is explained by the ρmnk-landscape parameters. This is not a surprise since these four parameters define the way ρmnk-landscapes are constructed; see Section II-C. However, let us remind that ρ and $\mathtt{k\_n}$ are *not* known in practice when solving a black-box problem instance. More interestingly, however, we see that the proposed local features, based on sampling, allow the model to obtain a better prediction accuracy than benchmark parameters. We attribute this to the fact that they are able to capture the variations between instances with the same parameters; i.e. the randomness in the construction of ρmnk-landscapes. This is particularly true for local features based on random walk, which contain more insightful information for search performance than the ones based on adaptive walk. Indeed, the regression accuracy obtained with the former subset of local features is almost as good as the combination of both. At last, we observe that global features, based on the enumeration of the solution space, obtain a better ranking, although the addition of local features seems to increase the predictive power of the regression model even more, as illustrated by the results obtained by the models using all features.
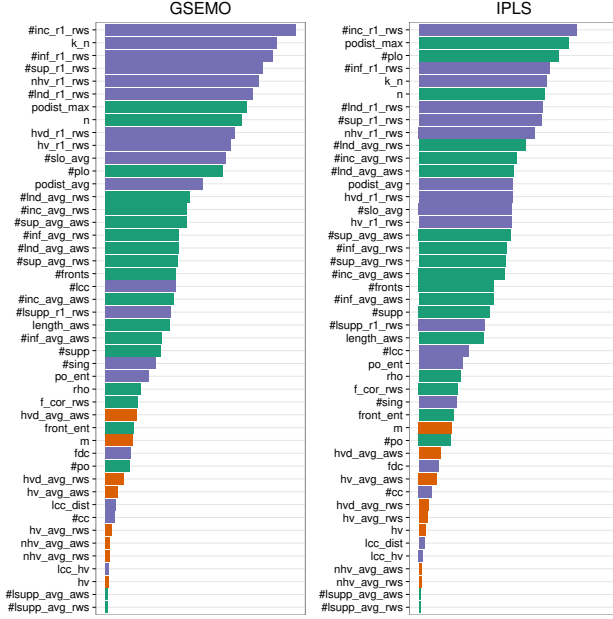
Fig. 9. Relative importance of features (mean decrease in node impurity) from totally-randomized regression trees for small instances.

D. Importance of Features for Search Performance

Tree-based predictive models also allow for the identification of which input features are the most important to make accurate predictions, which provides insight into the process being modeled [51], [52]. In particular, we consider the measure of importance that relates to the decrease in node impurity after each split on a given predictor; the larger the decrease, the more important the predictor. Note that, in the regression case, node impurity is measured by variance. We derive our estimates from a large ensemble of $50\,000$ shallow and totally-randomized regression trees. Choosing the input variable to split on totally at random prevents correlated variables to mask one another, which would result in underestimating their relative importance [52]. Then, by using small trees, we strive to minimize the effect of having a finite sample set, which introduces noise in the node impurities as trees grow. The relative importance of features thus extracted, is depicted in Fig. 9. For a given algorithm, features are sorted in decreasing order of importance, from top to bottom. Although the regression accuracy is similar for both algorithms, the most important features are different for G-SEMO and I-PLS.

For G-SEMO, the six most important features are all related to the ruggedness of the landscape (in violet). Apart from the proportional number of variable interactions k_n in ρmnk-landscapes, the others correspond to the first autocorrelation coefficient of the proportional number of dominated (#inf), dominating (#sup), and incomparable (#inc) neighbors, the proportional number of non-dominated solutions in the neighborhood (#lnd), and the hypervolume covered by the neighborhood (nhv) encountered along a random walk. Next in the ranking are those associated with objective correlation and dominance (in green), such as the diameter of

the PS (podist_max), which also correspond to the most important global feature. For I-PLS, features related to the ruggedness (in violet) and to the objective correlation (in green) seem equally important, and the features listed above also appear to be impactful. Most notably, the proportion of PLO (#plo) seems of high importance; it appears in the 3rd place for I-PLS and only in the 12th place for G-SEMO. By contrast, the features associated with the number of objectives (in orange) are of low importance for the two algorithms. Interestingly, for both G-SEMO and I-PLS, the most important benchmark parameter is the proportional number of variable interactions k_n, followed by the problem size n, the objective correlation ρ, and finally the number of objectives m.

V. Scaling to Large Instances

In this section, we extend our analysis to large-size instances. Since global features cannot be computed anymore, we investigate the ability of *local* features to explain algorithm performance for large dimensions.

A. Experimental Setup

We generate large-size ρmnk-landscapes by means of a design of experiments based on random latin hyper-cube sampling. We consider problem sizes in the range $n \in \{64, \ldots, 256\}$, numbers of variable interactions $k \in \{0, \ldots, 8\}$, numbers of objectives $m \in \{2, \ldots, 5\}$, and objective correlations $\rho \in \,]-1, 1]$ such that $\rho > -1/(m-1)$. A total of $1\,000$ problem instances is considered, a single instance is generated for each parameter setting provided by the design of experiments. We consider all local features as well as benchmark parameters, and the same two multi-objective algorithms. We perform 30 independent runs per instance and per algorithm, with a fixed budget of $100\,000$ calls to the evaluation function. The performance quality is measured in terms of the multiplicative epsilon indicator to the best-known non-dominated set.

B. Correlation between Landscape Features

As performed in the previous section for small instances, we report the correlation between each pair of features and the corresponding clustering in Fig. 10. Similar to our previous results, we obtain three clusters, each one being associated with one benchmark parameter.

The local features related to the first autocorrelation co-efficient measured on random walks remain in the cluster associated to ruggedness (in violet), as it was with small instances. In addition, both features measuring the average difference between the hypervolume covered by each neighbor and the one covered by the current solution (hvd) moved to this cluster, but their correlation with the other features in the cluster is low. Similar observations can be made for the problem size n. We attribute this to the design of experiments of this new dataset for large instances. Once again, the features related to dominance (in green) are all very close to one another. They relate very much to the objective correlation (ρ) and to the proportion of PLO (#plo). As with small instances,
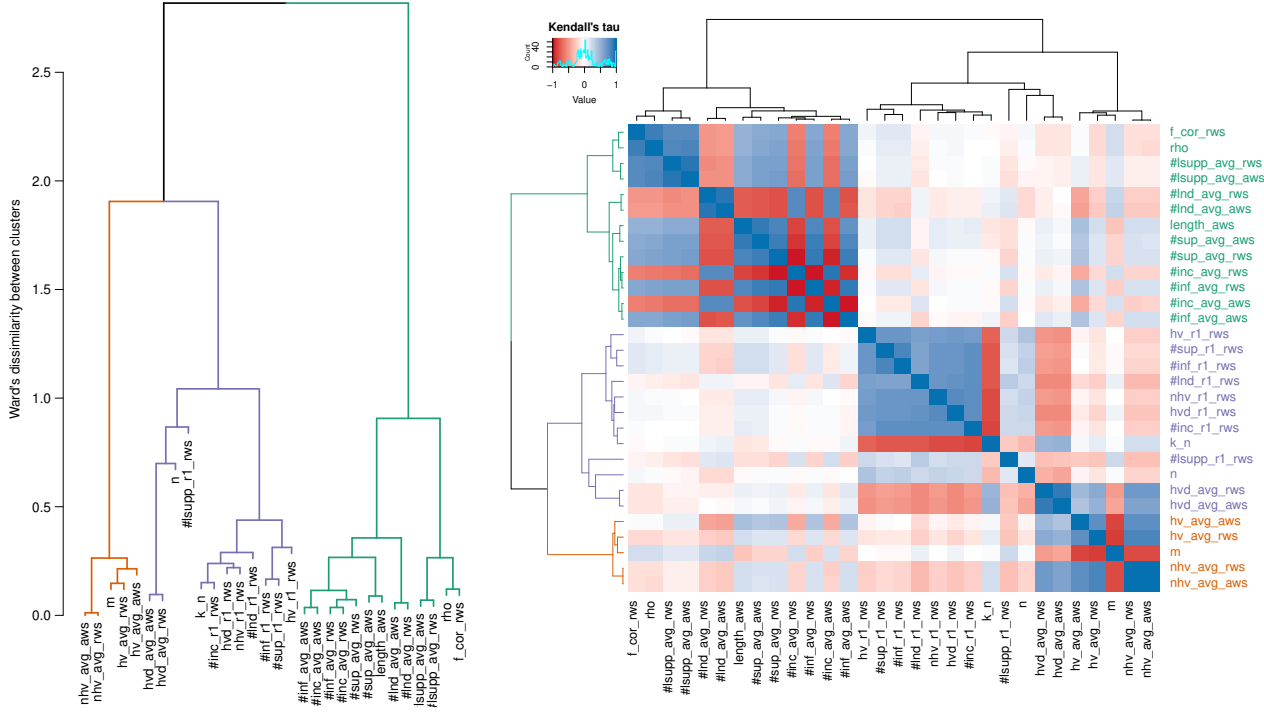
Fig. 10. Features clustering (left) and features association (right) computed over the whole set of large instances; see Fig. 8 for details.

features related to hypervolume are correlated with the number of objectives (m, in orange). Overall, there are no major changes with respect to the previous dataset, which validates our study on small instances.

C. Feature-based Performance Prediction

The prediction accuracy of regression models predicting search performance for different subsets of input variables is reported in Table III. Overall, the fitting quality is lower than for small instances. We attribute this to the smaller number of observations contained in the dataset for large instances ($1\,000$, against $60\,480$ for small instances). Once again, the results for G-SEMO and I-PLS are quite similar. As before, the objective correlation ρ and the proportional number of variable interactions k_n, which are unknown in a black-box scenario, are essential to understand search performance and to reach a good prediction accuracy. Surprisingly, the model using solely the solution space and objective space dimensions, n and m, has a negative R^2, and performs worse than a model that always predict the average performance value. Actually, a visual inspection (not reported) allows one to note that observed and predicted values can be far from each other, in particular for instances where algorithms are efficient. All other models obtain an R^2 larger than 0.8. This means that more than 80% of the variance in the algorithms performance is explained by local features. The set of all local features has a similar predictive power than (known and unknown) benchmark parameters. Let us remind that in this dataset, a single instance is generated per instance setting, so that there

is no variance between instances with the same parameters. As with small instances, the local features based on random walks have a higher predictive power than those based on adaptive walks, although the combination of both is always more accurate. Ultimately, local features allow the regression model to obtain a satisfying prediction accuracy. We analyze the importance of local features below, and then we study their relevance in the context of algorithm selection.

D. Importance of Features for Search Performance

The importance of features for both algorithms is reported in Fig. 11. For G-SEMO, features related to ruggedness (in violet) are more relevant than others, followed by features related to objective correlation, such as the estimator of the proportion of PLO (length_aws), and to a smaller extent, to features that are associated with the number of objectives (m). Interestingly, for I-PLS, features related to dominance and objective correlation (in green) are clearly much more informative. Indeed, the average rank of those features is 7 for I-PLS, whereas it is only 19.08 for G-SEMO. Conversely, the average rank of features related to ruggedness (in violet) is 10.33 for G-SEMO, against 23.67 for I-PLS. For both algorithms, the average rank of features related to the number of objectives (in orange) is about the same, and the second most important one (16.6 for G-SEMO, against 18 for I-PLS). This highlights that problem features impact local and global dominance-based multi-objective algorithms differently.

TABLE III
RANDOM SUBSAMPLING CROSS-VALIDATED REGRESSION ACCURACY OBTAINED ON LARGE INSTANCES (50 ITERATIONS, 90/10 SPLIT).

| Algo.	Set of features	MAE avg	MAE std	MSE avg	MSE std	R^2 avg	R^2 std	adjusted R^2 avg	adjusted R^2 std	Rank
G-SEMO	all features	**0.003049**	0.000285	**0.000017**	0.000004	0.891227	0.024584	0.843934	0.035273	1
	local features	**0.003152**	0.000295	**0.000018**	0.000004	0.883909	0.026863	0.838126	0.037457	1
	local features (random walk)	**0.003220**	0.000314	0.000019	0.000004	0.878212	0.028956	0.849287	0.035833	1.5
	local features (adaptive walk)	0.003525	0.000329	0.000023	0.000006	0.854199	0.032339	0.834089	0.036799	5
	$\{\rho, \mathrm{m}, \mathrm{n}, \mathrm{k\_n}\}$	**0.003084**	0.000270	**0.000017**	0.000003	0.892947	0.020658	0.888440	0.021528	1
	$\{\mathrm{m}, \mathrm{n}\}$	0.010813	0.000830	0.000206	0.000030	-0.303336	0.188046	-0.330209	0.191923	6
I-PLS	all features	**0.004290**	0.000430	**0.000034**	0.000008	0.886568	0.026980	0.837249	0.038710	1
	local features	**0.004359**	0.000423	**0.000035**	0.000008	0.883323	0.027274	0.837309	0.038030	1
	local features (random walk)	**0.004449**	0.000394	**0.000036**	0.000008	0.879936	0.026335	0.851421	0.032589	1
	local features (adaptive walk)	0.004663	0.000403	0.000039	0.000008	0.871011	0.025903	0.853219	0.029476	3.5
	$\{\rho, \mathrm{m}, \mathrm{n}, \mathrm{k\_n}\}$	**0.004353**	0.000320	**0.000033**	0.000006	0.889872	0.024505	0.885235	0.025537	1
	$\{\mathrm{m}, \mathrm{n}\}$	0.016959	0.001473	0.000472	0.000077	-0.568495	0.228629	-0.600836	0.233343	6
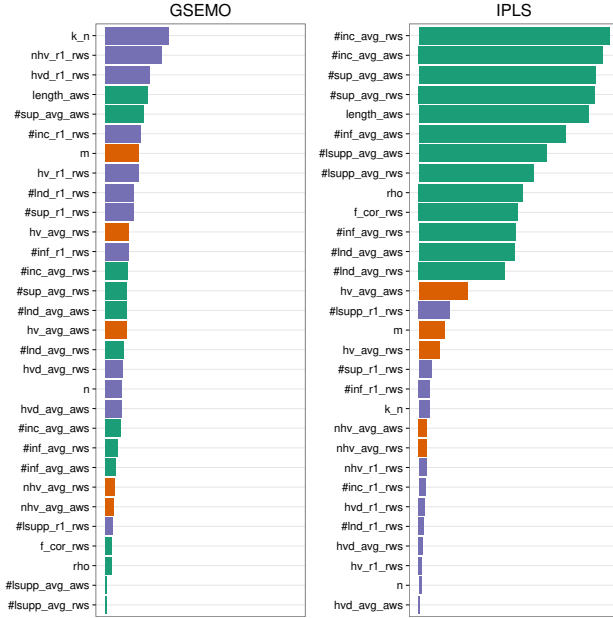
Fig. 11. Relative importance of features (mean decrease in node impurity) from totally-randomized regression trees for large instances.

VI. TOWARDS FEATURE-BASED ALGORITHM PORTFOLIO

We conclude our analysis with a feature-based algorithm selection using a portfolio of three EMO algorithms, namely NSGA-II [1] IBEA [2], and MOEA/D [3]. They were chosen as representatives of the state-of-the-art in the field, covering dominance-, indicator- and scalarization-based approaches, respectively. We rely on an out-of-the-box implementation with default parameters, as provided in jMetal 4.5 [53]. We first consider the dataset of large ρmnk-landscapes, and then a scenario for multi-objective quadratic assignment problems.

A. ρmnk-Landscapes

For ρmnk-landscapes, all three algorithms use a population of size 100, a 1-point crossover with a rate of 0.9, and a bit-flip mutation with a rate of $1/\mathrm{n}$, under a fixed budget of

$1\,000\,000$ evaluations. Notice that the dataset contains 999 observations: one instance was discarded as there was no distinction between the algorithms. In order to predict the best-performing algorithm for solving a given instance, we build an ensemble of 500 extremely randomized *classification* trees, in contrast to the regression models discussed so far. The output class is simply whether (i) NSGA-II, (ii) IBEA, or (iii) MOEA/D performs better, on average, for a given instance, in terms of hypervolume. The classification accuracy, measured in terms of a cross-validated error rate, is reported in Table IV. In fact, we report two error rates. In the error rate of best average performance, an error is taken into account if the predicted algorithm differs from the best performing algorithm on average. Complementarily, in the error rate of best statistical rank, an error is taken into account only if the predicted algorithm is significantly outperformed by any other according to a Mann-Whitney statistical test at a significance level of 0.05 with Bonferroni correction.

Overall, the feature-based classification models are able to reach an error rate below 0.131 for the best average performance and below 0.016 for the best statistical rank. As such, one of the significantly best-performing algorithms is predicted in more than 98.4% of the cases. That is significantly more accurate than the basic approach based on the solution and objective space dimensions (n and m), which has an error rate of about 41.3% for the best average performance, and of 19.7% for the best statistical rank. Notice that a naive approach that always chooses the best algorithm on average (NSGA-II) has an error rate of more than 50%, while always selecting the algorithm with the best statistical rank (MOEA/D) would result in more than 12% of errors. We did not find any statistical difference between all other classification models, although the model with the lowest error always uses local features. Note that models built on features from random walks only are almost as good as any other model: this might provide a viable option to reduce the computational cost of the portfolio without altering much the prediction accuracy.

For the sake of providing a model that is easier to interpret, we construct another classifier based on a simple decision tree [54], [55], as illustrated in Fig. 12. Even with such a simple decision tree of depth three, the proposed features

TABLE IV
RANDOM SUBSAMPLING CROSS-VALIDATED CLASSIFICATION ACCURACY OBTAINED ON LARGE ρMNK-LANDSCAPES (50 ITERATIONS, 90/10 SPLIT).

| Set of features	Error rate of best average performance		Rank	Error rate of best statistical rank		Rank
	mean	std		mean	std	
all features	**0.122222**	0.031033	1	**0.012727**	0.014110	1
local features	**0.123030**	0.030521	1	**0.013737**	0.014103	1
local features (random walk)	**0.118788**	0.029187	1	**0.013333**	0.012149	1
local features (adaptive walk)	**0.130303**	0.029308	1	**0.015354**	0.014026	1
$\{\rho, \mathtt{m}, \mathtt{n}, \mathtt{k\_n}\}$	**0.125859**	0.028875	1	**0.014141**	0.013382	1
$\{\mathtt{m}, \mathtt{n}\}$	0.413333	0.045533	6	0.197374	0.043778	6
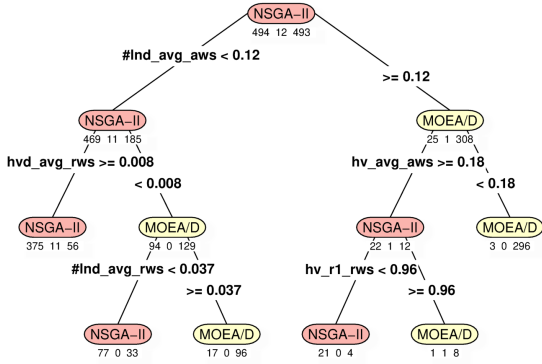
Fig. 12. CART decision tree for algorithm selection on ρmnk-landscapes. The nodes report the number of instances where NSGA-II, IBEA, and MOEA/D performs better in average, from left to right, respectively.

are able to distinguish between the algorithms with a cross-validated error rate on best average performance of 12.61%. The root of the decision tree is a feature related to the objective correlation (#lnd_avg_aws), measured in terms of the proportion of locally non-dominated neighbors encountered along an adaptive walk. When there are few non-dominated solutions in the neighborhood, NSGA-II has more chances of being selected. This typically happens when the objectives are correlated. Indeed, on the left-hand side of the tree, NSGA-II outperforms IBEA and MOEA/D on 469 instances, whereas it is outperformed only 196 times. On the contrary, when there are more non-dominated solutions, MOEA/D shall be selected, as it performs better on 308 instances, against 26 for the other algorithms. In order to reduce the error rate on the left-hand side of the decision tree, two features are considered (hvd_avg_rws and #lnd_avg_rws), both related to ruggedness. Roughly speaking, MOEA/D shall be preferred over NSGA-II for correlated objectives only when the landscape is relatively smooth. Overall, this emphasizes that a single feature is not enough to distinguish between the different algorithms, and that multiple features, in this case related to ruggedness and objective correlation, are required to design an accurate portfolio approach. This simple example illustrates the potential of algorithm selection based on multi-objective landscape features for large dimensions.

B. Multi-objective QAP

The second considered algorithm portfolio scenario deals with another problem class: the multi-objective quadratic assignment problem (mQAP) [12]. The mQAP differs from ρmnk-landscapes in multiple aspects, notably the solution

representation, which is based on permutations and not binary strings. We generate $1\,000$ mQAP instances based on [12], and following a design of experiments based on random latin hypercube sampling. We consider problem sizes in the range $\mathtt{n} \in \{30, \ldots, 100\}$, numbers of objectives $\mathtt{m} \in \{2, \ldots, 5\}$, objective correlations $\rho \in\,]-1, 1]$, and two instance types: uniform and real-like instances [12]. Bear in mind that the instance type and the objective correlation are *unknown* in practice for black-box mQAP instances. Local features are computed based on a single adaptive walk, and a single random walk of length $\ell = 200$. Moreover, in order to cope with the quadratic nature of the swap neighborhood considered for mQAP, we simply sample 200 neighbors at random instead of performing a full neighborhood exploration at each step. In terms of algorithms, we still consider NSGA-II, IBEA and MOEA/D, with a population of size 100, a swap mutation with a rate of 0.2, and a 2-point crossover with a rate of 0.95. The crossover operator simply copies the segment in-between two randomly-chosen points from one parent, and fill the missing values following the order from the other parent [53]. All the algorithms stop after $1\,000\,000$ evaluations.

Table V reports the error rates obtained by the classification model based on different subsets of features. Similar to ρmnk-landscapes, apart from the simple model based on $\mathtt{m}$ and $\mathtt{n}$, the error rate is always below 17.1% in terms of best average performance, and below 1.1% in terms of best statistical rank. In other words, a feature-based classification model is able to choose an algorithm that is not significantly outperformed by any other in almost 99% of the cases. This shows that the information contained in the proposed local features generalizes to other problem classes. Notice that IBEA is the best-performing algorithm overall for mQAP, being outperformed on 51.8% of instances only in terms of average performance, and on 31.4% of instances in terms of statistical difference. Once again, we did not observe any significant difference between all classification models, but the one based on $\mathtt{m}$ and $\mathtt{n}$. It is interesting to notice that, in average, the total cost of local features represents less than 5% of the budget allocated to the search process in this case. A simple decision tree for mQAP is provided in Fig. 13. In this case, the most discriminant feature to distinguish between the algorithms is length_aws, that is the length of the adaptive walk. As such, IBEA is recommended when the length is large, that is, when there are few PLO. On the contrary, MOEA/D is recommended when there are more PLO. This emphasizes the high importance of multimodality for mQAP as well, over the number of objectives and their degree of conflict.

TABLE V
RANDOM SUBSAMPLING CROSS-VALIDATED CLASSIFICATION ACCURACY OBTAINED ON MQAP (50 ITERATIONS, 90/10 SPLIT).

| Set of features	Error rate of best average performance mean	std	Rank	Error rate of best statistical rank mean	std	Rank
all features	**0.1558**	0.038124	1	**0.0076**	0.008704	1
local features	**0.1560**	0.038012	1	**0.0070**	0.010152	1
local features (random walk)	**0.1632**	0.036558	1	**0.0104**	0.011773	1
local features (adaptive walk)	**0.1704**	0.033195	1	**0.0066**	0.009392	1
$\{\texttt{type}, \texttt{m}, \texttt{n}, \rho\}$	**0.1686**	0.035226	1	**0.0100**	0.010880	1
$\{\texttt{m}, \texttt{n}\}$	0.3132	0.044741	6	0.1156	0.031178	6
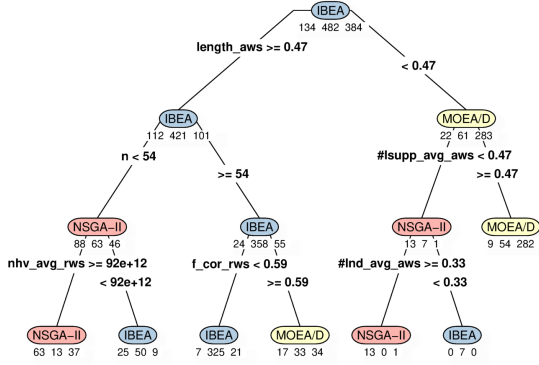
Fig. 13. CART decision tree for algorithm selection on mQAP. The nodes report the number of instances where NSGA-II, IBEA, and MOEA/D performs better in average, from left to right, respectively.

VII. CONCLUSIONS

In this paper, we investigated the potential of landscape features to explain and predict the performance of EMO algorithms for black-box multi-objective combinatorial optimization. We reviewed the state-of-the-art of multi-objective landscape analysis, and we proposed new *general-purpose* features characterizing the landscape, which are affordable for high-dimensional problems due to their local nature. By analyzing their association and relevance to search performance, we highlighted the insightful information they are able to capture regarding problem difficulty. In the context of performance prediction, our data-driven analysis revealed the crucial importance of considering *multiple* features to reach a good prediction accuracy. From a benchmarking point-of-view, we showed that not only the number of objectives, but also their degree of conflict, are jointly important to model search performance. Even more notably, ruggedness and multimodality, which are often overlooked in the EMO literature, constitute crucial dimensions that complements the portrait of multi-objective landscapes. By extending results from single-objective landscape analysis, we were able to design affordable features to characterize ruggedness and multimodality in multi-objective optimization. Interestingly, relevant features are not the same for the considered algorithms, which allows us to understand what makes an instance more difficult to solve for a given algorithm. Ultimately, two algorithm selection scenarios with a portfolio of three algorithms allowed us to emphasize that, by leveraging the proposed landscape features, one can accurately select the most appropriate algorithm for different large-size problem classes and instances.

Extending and analyzing our feature-based performance prediction and algorithm selection methodology by considering more practical scenarios would allow us to increase our understanding of the landscape structure exhibited by black-box multi-objective combinatorial optimization problems, particularly when an instance generator is not available. Consequently, we plan to consider other multi-objective problem and algorithms classes, to study how algorithm component choices impact search performance over different multi-objective landscapes, and how this could help improving the design of EMO algorithms. Another challenge for feature-based algorithm selection is to investigate the tradeoff between feature cost and accuracy.

REFERENCES

[1] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm : NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.

[2] E. Zitzler and S. Künzli, "Indicator-based selection in multiobjective search," in *Conference on Parallel Problem Solving from Nature (PPSN VIII)*, ser. Lecture Notes in Computer Science, vol. 3242. Birmingham, UK: Springer, 2004, pp. 832–842.

[3] Q. Zhang and H. Li, "MOEA/D: A multiobjective evolutionary algorithm based on decomposition," *IEEE Transactions on Evolutionary Computation*, vol. 11, no. 6, pp. 712–731, 2007.

[4] H. Richter and A. Engelbrecht, Eds., *Recent Advances in the Theory and Application of Fitness Landscapes*, ser. Emergence, Complexity and Computation. Springer, 2014.

[5] M. Wagner, T. Friedrich, and M. Lindauer, "Improving local search in a minimum vertex cover solver for classes of networks," in *IEEE Congress on Evolutionary Computation (CEC 2017)*, San Sebastián, Spain, 2017, pp. 1704–1711.

[6] O. Mersmann, B. Bischl, H. Trautmann, M. Wagner, J. Bossek, and F. Neumann, "A novel feature-based approach to characterize algorithm performance for the traveling salesperson problem," *Annals of Mathematics and Artificial Intelligence*, pp. 1–32, 2013.

[7] E. D. Weinberger, "Correlated and uncorrelatated fitness landscapes and how to tell the difference," *Biological Cybernetics*, vol. 63, no. 5, pp. 325–336, 1990.

[8] S. A. Kauffman, *The Origins of Order*. Oxford University Press, 1993.

[9] P. Merz, "Advanced fitness landscape analysis and the performance of memetic algorithms," *Evolutionary Computation*, vol. 12, no. 3, pp. 303–325, 2004.

[10] M. Preuss, *Multimodal Optimization by Means of Evolutionary Algorithms*, ser. Natural Computing Series. Springer, 2015.

[11] P. Kerschke, H. Wang, M. Preuss, C. Grimme, A. Deutz, H. Trautmann, and M. Emmerich, "Towards analyzing multimodality of continuous multiobjective landscapes," in *Parallel Problem Solving from Nature (PPSN XIV)*, ser. Lecture Notes in Computer Science, vol. 9921. Edinburgh, UK: Springer, 2016, pp. 962–972.

[12] J. Knowles and D. Corne, "Instance generators and test suites for the multiobjective quadratic assignment problem," in *Evolutionary Multi-Criterion Optimization (EMO 2003)*, ser. Lecture Notes in Computer Science, vol. 2632. Faro, Portugal: Springer, 2003, pp. 295–310.

[13] L. Paquete, T. Schiavinotto, and T. Stützle, "On local optima in multiobjective combinatorial optimization problems," *Annals of Operations Research*, vol. 156, no. 1, pp. 83–97, 2007.

[14] L. Paquete and T. Stützle, "Clusters of non-dominated solutions in multiobjective combinatorial optimization: An experimental analysis," in *Multiobjective Programming and Goal Programming: Theoretical Results and Practical Applications*, ser. Lecture Notes in Economics and Mathematical Systems, vol. 618. Springer, 2009, pp. 69–77.

[15] D. Garrett and D. Dasgupta, "Multiobjective landscape analysis and the generalized assignment problem," in *Learning and Intelligent OptimizatioN (LION 2)*, ser. Lecture Notes in Computer Science, vol. 5313. Trento, Italy: Springer, 2007, pp. 110–124.

[16] ——, "Plateau connection structure and multiobjective metaheuristic performance," in *IEEE Congress on Evolutionary Computation (CEC 2009)*, 2009, pp. 1281–1288.

[17] H. Aguirre and K. Tanaka, "Working principles, behavior, and performance of MOEAs on MNK-landscapes," *European Journal of Operational Research*, vol. 181, no. 3, pp. 1670–1690, 2007.

[18] S. Verel, A. Liefooghe, L. Jourdan, and C. Dhaenens, "On the structure of multiobjective combinatorial search space: MNK-landscapes with correlated objectives," *European Journal of Operational Research*, vol. 227, no. 2, pp. 331–342, 2013.

[19] A. Liefooghe, S. Verel, H. Aguirre, and K. Tanaka, "What makes an instance difficult for black-box 0–1 evolutionary multiobjective optimizers?" in *Artificial Evolution (EA 2013)*, ser. Lecture Notes in Computer Science, vol. 8752, Bordeaux, France, 2013, pp. 3–15.

[20] F. Daolio, A. Liefooghe, S. Verel, H. Aguirre, and K. Tanaka, "Problem features versus algorithm performance on rugged multiobjective combinatorial fitness landscapes," *Evolutionary Computation*, vol. 25, no. 4, pp. 555–585, 2017.

[21] K. Smith-Miles, "Towards insightful algorithm selection for optimisation using meta-learning concepts," in *International Joint Conference on Neural Networks (IJCNN 2008)*. Hong Kong: IEEE, 2008, pp. 4118–4124.

[22] P. Kerschke, H. Hoos, F. Neumann, and H. Trautmann, "Automated algorithm selection: Survey and perspectives," *Evolutionary Computation*, vol. 27, no. 1, pp. 3–45, 2019.

[23] S. Kadioglu, Y. Malitsky, M. Sellmann, and K. Tierney, "ISAC – instance-specific algorithm configuration," in *19th European Conference on Artificial Intelligence (ECAI 2010)*, ser. Frontiers in Artificial Intelligence and Applications, vol. 215. Lisbon, Portugal: IOS Press, 2010, pp. 751–756.

[24] N. Belkhir, J. Dréo, P. Savéant, and M. Schoenauer, "Per instance algorithm configuration of CMA-ES with limited budget," in *Genetic and Evolutionary Computation Conference (GECCO 2017)*. Berlin, Germany: ACM, 2017, pp. 681–688.

[25] J. R. Rice, "The algorithm selection problem," *Advances in Computers*, vol. 15, pp. 65–118, 1976.

[26] K. Smith-Miles, D. Baatar, B. Wreford, and R. Lewis, "Towards objective measures of algorithm performance across instance space," *Computers & Operations Research*, vol. 45, pp. 12–24, 2014.

[27] F. Hutter, L. Xu, H. Hoos, and K. Leyton-Brown, "Algorithm runtime prediction: Methods & evaluation," *Artificial Intelligence*, vol. 206, pp. 79–111, 2014.

[28] F. Hutter, H. Hoos, and K. Leyton-Brown, "Sequential model-based optimization for general algorithm configuration," in *Learning and Intelligent OptimizatioN (LION 5)*, ser. Lecture Notes in Computer Science, vol. 6683. Rome, Italy: Springer, 2011, pp. 507–523.

[29] F. Daolio, S. Verel, G. Ochoa, and M. Tomassini, "Local optima networks and the performance of iterated local search," in *Genetic and Evolutionary Computation Conference (GECCO 2012)*. Philadelphia, USA: ACM, 2012, pp. 369–376.

[30] A. Liefooghe, B. Derbel, S. Verel, H. Aguirre, and K. Tanaka, "Towards landscape-aware automatic algorithm configuration: Preliminary experiments on neutral and rugged landscapes," in *Evolutionary Computation in Combinatorial Optimization (EvoCOP 2017)*, ser. Lecture Notes in Computer Science, vol. 10197. Amsterdam, The Netherlands: Springer, 2017, pp. 215–232.

[31] A. Beham, S. Wagner, and M. Affenzeller, "Algorithm selection on generalized quadratic assignment problem landscapes," in *Genetic and Evolutionary Computation Conference (GECCO 2018)*. Kyoto, Japan: ACM, 2018, pp. 253–260.

[32] B. Bischl, O. Mersmann, H. Trautmann, and M. Preuss, "Algorithm selection based on exploratory landscape analysis and cost-sensitive learning," in *Genetic and Evolutionary Computation Conference (GECCO 2012)*. Philadelphia, USA: ACM, 2012, pp. 313–320.

[33] K. Malan and A. Engelbrecht, "Fitness landscape analysis for meta-heuristic performance prediction," in *Recent Advances in the Theory and Application of Fitness Landscapes*. Springer, 2014, pp. 103–132.

[34] P. Kerschke and H. Trautmann, "Automated algorithm selection on continuous black-box problems by combining exploratory landscape analysis and machine learning," *Evolutionary Computation*, vol. 27, no. 1, pp. 99–127, 2019.

[35] M. Laumanns, L. Thiele, and E. Zitzler, "Running time analysis of evolutionary algorithms on a simplified multiobjective knapsack problem," *Natural Computing*, vol. 3, no. 1, pp. 37–51, 2004.

[36] M. M. Drugan and D. Thierens, "Path-guided mutation for stochastic pareto local search algorithms," in *Parallel Problem Solving from Nature (PPSN XI)*, ser. Lecture Notes in Computer Science, vol. 6238. Kraków, Poland: Springer, 2010, pp. 485–495.

[37] M. Ehrgott, *Multicriteria optimization*, 2nd ed. Springer, 2005.

[38] E. Zitzler, L. Thiele, M. Laumanns, C. M. Fonseca, and V. Grunert da Fonseca, "Performance assessment of multiobjective optimizers: An analysis and review," *IEEE Transactions on Evolutionary Computation*, vol. 7, no. 2, pp. 117–132, 2003.

[39] T. Jones and S. Forrest, "Fitness distance correlation as a measure of problem difficulty for genetic algorithms," in *6th International Conference on Genetic Algorithms (ICGA 1995)*, San Francisco, CA, USA, 1995, pp. 184–192.

[40] M. Ehrgott and K. Klamroth, "Connectedness of efficient solutions in multiple criteria combinatorial optimization," *European Journal of Operational Research*, vol. 97, no. 1, pp. 159–166, 1997.

[41] J. Gorski, K. Klamroth, and S. Ruzika, "Connectedness of efficient solutions in multiple objective combinatorial optimization," *Journal of Optimization Theory and Applications*, vol. 150, no. 3, pp. 475–497, 2011.

[42] A. Liefooghe, L. Paquete, and J. R. Figueira, "On local search for bi-objective knapsack problems," *Evolutionary Computation*, vol. 21, no. 1, pp. 179–196, 2013.

[43] I. Moser, M. Gheorghita, and A. Aleti, "Identifying features of fitness landscapes and relating them to problem difficulty," *Evolutionary computation*, vol. 25, no. 3, pp. 407–437, 2017.

[44] T. Smith, P. Husbands, and M. O'Shea, "Fitness landscapes and evolvability," *Evolutionary Computation*, vol. 10, no. 1, pp. 1–34, 2002.

[45] F. Murtagh and P. Legendre, "Ward's hierarchical agglomerative clustering method: which algorithms implement ward's criterion?" *Journal of Classification*, vol. 31, no. 3, pp. 274–295, 2014.

[46] G. R. Warnes, B. Bolker, L. Bonebakker, R. Gentleman, W. H. A. Liaw, T. Lumley, L. Maechler, A. Magnusson, S. Moeller, M. Schwartz, and B. Venables, *gplots: Various R Programming Tools for Plotting Data*, 2016, R package version 3.0.1.

[47] P. Geurts, D. Ernst, and L. Wehenkel, "Extremely randomized trees," *Machine Learning*, vol. 63, no. 1, pp. 3–42, 2006.

[48] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[49] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[50] W. J. Conover, *Practical Nonparametric Statistics*. John Wiley, 1999.

[51] G. Louppe, L. Wehenkel, A. Sutera, and P. Geurts, "Understanding variable importances in forests of randomized trees," in *Advances in Neural Information Processing Systems 26*. Curran Associates, Inc., 2013, pp. 431–439.

[52] G. Louppe, "Understanding random forests: From theory to practice," Ph.D. dissertation, Université de Liège, Liège, Belgique, 2014.

[53] J. J. Durillo and A. J. Nebro, "jMetal: A java framework for multi-objective optimization," *Advances in Engineering Software*, vol. 42, pp. 760–771, 2011.

[54] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, *Classification and regression trees*. Wadsworth & Brooks, 1984.

[55] T. Therneau and B. Atkinson, *rpart: Recursive Partitioning and Regression Trees*, 2018, R package version 4.1-13.