Decomposition-based multi-objective landscape features and automated algorithm selection

Raphaël Cosson, Bilel Derbel, Arnaud Liefooghe, Hernán Aguirre, Kiyoshi Tanaka, Qingfu Zhang

Decomposition-based Multi-objective Landscape Features and Automated Algorithm Selection

Raphaël Cosson[1], Bilel Derbel[1], Arnaud Liefooghe[1], Hernán Aguirre[2], Kiyoshi Tanaka[2], and Qingfu Zhang[3]

[1] Univ. Lille, CNRS, Inria, Centrale Lille, UMR 9189 CRIStAL, F-59000 Lille, France
{raphael.cosson.etu,bilel.derbel,arnaud.liefooghe}@univ-lille.fr
[2] Shinshu University, Faculty of Engineering, Japan
{ahernan,ktanaka}@shinshu-u.ac.jp
[3] City University of Hong Kong, Kowloon Tong, Hong Kong
qingfu.zhang@cityu.edu.hk

Abstract. Landscape analysis is of fundamental interest for improving our understanding on the behavior of evolutionary search, and for developing general-purpose automated solvers based on techniques from statistics and machine learning. In this paper, we push a step towards the development of a landscape-aware approach by proposing a set of landscape features for multi-objective combinatorial optimization, by decomposing the original multi-objective problem into a set of single-objective sub-problems. Based on a comprehensive set of bi-objective ρmnk-landscapes and three variants of the state-of-the-art Moea/d algorithm, we study the association between the proposed features, the global properties of the considered landscapes, and algorithm performance. We also show that decomposition-based features can be integrated into an automated approach for predicting algorithm performance and selecting the most accurate one on blind instances. In particular, our study reveals that such a landscape-aware approach is substantially better than the single best solver computed over the three considered Moea/d variants.

1 Introduction

Context. Evolutionary algorithms have been proven extremely effective for solving a broad range of optimization problems. In the last decades, the community has gained a deep understanding on the key components underlying the design of a successful evolutionary approach for a given problem. However, one of the main challenge remains to *automate* the process of choosing the most suitable algorithm or configuration for the instance under consideration. In fact, it is well known that the structural properties of an optimization problem highly impact the dynamics and performance of search algorithms, leading to the requirement of adopting a *landscape-aware algorithm selection and configuration* methodology for the success of evolutionary problem solving. On the one hand, landscape analysis [15] provides a principled approach for studying and analyzing the relation between the underlying search space structure and algorithm behavior. On the other hand, machine learning techniques can be leveraged to perform sophisticated tasks, such as predicting algorithm performance, identifying the best algorithm configuration, or selecting the best algorithm [7]. As a byproduct, and since the pioneer work of Rice [14], landscape-aware algorithm configuration has emerged as an appealing approach for increasing the effectiveness and efficiency of evolutionary algorithms. In this paper, we contribute to the development of such an approach when specifically dealing with *multi-objective* and *combinatorial* optimization problems.

Related Work. Independently of whether the target problems and algorithms aim at optimizing a single or multiple objectives, and of whether they have a combinatorial or continuous nature, every landscape-aware methodology needs to address the following two research challenges: (i) the design of a set of informative and high-level *landscape features*, and (ii) the development of automated recommendation systems integrating the so-designed features on the basis of *statistical or machine learning prediction models*. Looking at the specialized literature, a large amount of work has been conducted for single-objective optimization since pioneering works in the field [4]. For instance, in single-objective continuous optimization, the exploratory landscape analysis (ELA) constitutes one major achievement made by the community to collect and combine existing features under a common tool and methodology [7]. Similarly, a number of features for single-objective combinatorial optimization have been developed over the years, and recent studies integrate them into sophisticated automated approaches for algorithm selection and configuration [1,9]. Those single-objective features are either based on problem-specific characteristics such as the maximum cost between two cities in the traveling salesperson problem [13], or on general descriptors from the underlying landscape. In the latter case, this is achieved by relying on a neighborhood relation over the search space in order to define a (combinatorial) landscape, and by studying its properties and characteristics in terms of multimodality, ruggedness, or neutrality [15].

Despite the significant progress made in the last decades, the existing literature on the development of a unified landscape-aware approach targeting multi-objective optimization problems is more scarce [6,8]. Although landscape features can in principle be applied to the multi-objective case, the statistical and machine learning models considered in the single-objective case are still to be studied and validated when turning into a multi-objective setting. For multi-objective combinatorial optimization, one can refer to the recent study in [8], providing a comprehensive analysis of problem-independent multi-objective landscape features, and showing their effectiveness in predicting algorithm performance and in selecting from an algorithm portfolio. The features described there are mostly based on *dominance* and (hypervolume) *indicator*. In fact, they were designed to grasp the landscape characteristics, but also to capture the search behavior of dominance-based multi-objective algorithms. In a subsequent study [10], those previous features were shown to provide useful information about the performance of different classes of multi-objective evolutionary algorithms (MOEAs).

However, their correlation with the performance of decomposition-based MOEAs reveal to be less significant [10]. Getting inspiration from the fact that successful MOEAs may rely on different search paradigms, our work considers applying other mechanisms, such as decomposition, to design new multi-objective landscape features, hence pushing one step further the development of a landscape-aware automated evolutionary approach.

Methodology and Contribution. We rely on the concept of decomposition [16] to develop a new set of general-purpose multi-objective landscape features and to study their effectiveness when integrated into an automated algorithm selection task. Our interest in the concept of decomposition stems from the fact that it provides a state-of-the-art framework, represented by the MOEA/D algorithm [18], by simply decomposing the multi-objective problem into a number of single-objective sub-problems. We view the decomposition paradigm as an opportunity to leverage existing single-objective features for multi-objective landscape analysis. Our contributions can be summarized as follows.

(i) We propose a new set of high-level multi-objective landscape features based on decomposition. Intuitively, we attempt to capture the multi-objective landscape by aggregating the characteristics from the single-objective sub-problem landscapes obtained by decomposition. As such, the proposed features are obtained by first defining single-objective features for each sub-problem, and then aggregating them by means of descriptive statistics.

(ii) We consider the task of predicting performances of three variants of the state-of-the-art MOEA/D algorithm using a tree-based regression model to study the effectiveness of a decomposition-based landscape-aware methodology for automatically selecting the best performing algorithm for a given instance.

(iii) Throughout an extensive set of experiments using ρ`mnk`-landscapes with two objectives as a case study, we conduct a systematic analysis on the association between the designed features and the benchmarked landscapes, as well as the association between features and algorithm performance. Our findings reveal that the designed features are able to capture the benchmark parameters (ρ and `k`), and to substantially improve the so-called single best solver when integrated into a landscape-aware algorithm selection approach.

Outline. In Section 2, we describe the proposed multi-objective landscape features. In Section 3, we study the association of features with benchmark parameters. In Section 4, we investigate the integration of the proposed features into an automated landscape-aware approach. In Section 5, we conclude the paper.

2 From Single- to Multi-objective Features based on Decomposition

2.1 Multi-objective Optimization

A multi-objective combinatorial optimization problem can be defined by a set of $\mathtt{m}$ objective functions $f = (f_1, f_2, \ldots, f_{\mathtt{m}})$, and a discrete set X of feasible solutions in the *decision space*. Let $Z = f(X) \subseteq \mathbb{R}^{\mathtt{m}}$ be the set of feasible outcome vectors in the *objective space*. To each solution $x \in X$ is assigned an objective vector $z \in Z$, on the basis of the vector function $f : X \rightarrow Z$. In a maximization context, an objective vector $z \in Z$ is *dominated* by a vector $z' \in Z$ iff $\forall m \in \{1, \ldots, \mathtt{m}\}$, $z_m \leqslant z'_m$ and $\exists m \in \{1, \ldots, \mathtt{m}\}$ s.t. $z_m < z'_m$. A solution $x \in X$ is dominated by a solution $x' \in X$ iff $f(x)$ is dominated by $f(x')$. A solution $x^\star \in X$ is *Pareto optimal* if there does not exist any other solution $x \in X$ such that $x^\star$ is dominated by x. The set of all Pareto optimal solutions is the *Pareto set*. Its mapping in the objective space is the *Pareto front*. Our goal is to identify a good Pareto set approximation, for which multi-objective evolutionary algorithms (MOEAs) constitute a popular effective option [3].

2.2 Rationale, Methodology and Features Overview

In this work, we get inspiration from the so-called MOEAs based on decomposition [16] in order to design new high-level multi-objective features. This algorithm class is based on searching for good-performing solutions in multiple regions of the Pareto front by *decomposing* the original multi-objective problem into a number of scalarized *single-objective sub-problems*. Each sub-problem is obtained by a different parameterization of the same underlying scalarizing function. This is typically what the state-of-the-art MOEA/D algorithm [18] performs, while introducing a cooperation among sub-problem solving. In particular, this offers much flexibility for integrating existing single-objective search operators and solvers, which is actually one of the main reasons for the success of decomposition-based MOEAs. Let us however recall that our main goal is *not* to design a new multi-objective algorithm, but to design new multi-objective features that can feed the design of a general-purpose landscape-aware approach.

Therefore, we propose to rely on the simple observation that *each* of the so-defined sub-problems also implies a single-objective landscape that we can attempt to analyze and characterize. In other words, by studying the single-objective landscape implied by the sub-problems, we should be able to extract some knowledge about the original multi-objective problem. More precisely, the methodology that we adopt in the reminder of this paper consists in: (i) defining a number of single-objective landscapes using decomposition, (ii) extracting single-objective features for each sub-problem landscape, and (iii) aggregating those single-objective features into new multi-objective features. These steps are detailed below.

Sub-problem Landscape Definition. Firstly, we define μ scalarized single-objective sub-problems, where both the scalarizing function and the μ value are user-defined parameters. Among the different scalarizing functions that may be used, the Chebyshev function is one of the most effective since it can be shown that any Pareto optimal solution can be achieved by solving a well-parameterized Chebyshev sub-problem. In the rest of this paper, we should hence use the Chebyshev scalarizing function: $\mathtt{g}(x|\omega) = \max_{i \in \{1,\ldots,\mathtt{m}\}} \omega_i \cdot \big|z^\star_i - f_i(x)\big|$, such that $x \in X$,

$\omega = (\omega_1, \ldots, \omega_{\mathtt{m}})$ is a positive weight vector, and $z^\star = (z^\star_1, \ldots, z^\star_{\mathtt{m}})$ is a reference point such that $z^\star_i > f_i(x)\ \forall x \in X,\ i \in \{1, \ldots, \mathtt{m}\}$. It should be clear that μ different sub-problems can be obtained by choosing μ different weight vectors, denoted by $\omega^j,\ j \in \{1, \ldots, \mu\}$.

It is worth noticing that we do not make any assumption about the original (black-box) multi-objective problem, so that we have no information about what region every sub-problem is actually mapping to. Hence, the value of μ as well as the procedure to generate the weight vector can be a critical issue. This is studied in more details later when reporting our empirical results.

Single-objective Landscape Features. Next, we define a landscape for every single-objective sub-problem $j \in \{1, \ldots, \mu\}$, for which we compute a number of underlying high-level single-objective features. Following the standard literature on single-objective landscape analysis [15], the landscape of sub-problem j can be defined as a triplet $(X, \mathcal{N}, \mathsf{g}(\cdot|\omega^j))$, such that $\mathcal{N} : X \longrightarrow 2^X$ is a neighborhood relation defined among solutions for the considered problem; e.g., 1-bit-flips for binary strings, or swaps for permutations.

The considered sub-problem features are based on sampling the so-defined landscape to compute some statistics. Following the standard literature [15,5], we consider two simple sampling strategies, namely random walks (rws) and adaptive walks (aws). Generally speaking, a walk is an ordered sequence of solutions $(x_0, x_1, \ldots, x_\ell)$ such that $x_0 \in X$, and $x_t \in \mathcal{N}(x_{t-1})\ \forall t \in \{1, \ldots, \ell\}$. In a random walk, x_t being the current solution, the next solution x_{t+1} is simply a random neighbor under $\mathcal{N}$. The length of a random walk is a user-defined parameter. In an adaptive walk, the next solution x_{t+1} is selected to be an improving neighbor with respect to the single-objective scalarizing function $\mathsf{g}(\cdot|\omega^j)$. Consequently, the length of an adaptive walk is the number of steps performed until no further improvement is possible, x_ℓ is then a local optimum. Notice that the reference point z^* required for computing the scalar fitness values is updated on the basis of the best objective values seen so far during the walk.

Given a sub-problem $j \in \{1, \ldots, \mu\}$ and a walk $(x_0, x_1, \ldots, x_\ell)$, we consider the following four classes of single-objective features, as summarized in Table 1:

- **Fitness value (fv_*).** In the first class, we compute some statistics informing about the distribution of fitness values observed along the walk. More precisely, we consider the mean (avg) and standard deviation (sd) of the fitness values of collected solutions. We also consider three additional statistics, namely the first auto-correlation coefficient (r1), the kurtosis (kr), and the skewness (sk) of fitness values. The kurtosis and the skewness are standard measures in statistical analysis, while the first auto-correlation coefficient is mostly used in the landscape analysis literature. Denoting by $\bar{\mathsf{g}}(\cdot|\omega^j)$ the average fitness value of solutions in the walk, the first auto-correlation coefficient is defined as follows [5]:

$$\mathsf{r1} = \frac{\sum_{t=0}^{\ell-1} \left(\mathsf{g}(x_t|\omega^j) - \bar{\mathsf{g}}(\cdot|\omega^j)\right) \cdot \left(\mathsf{g}(x_{t+1}|\omega^j) - \bar{\mathsf{g}}(\cdot|\omega^j)\right)}{\left(\sum_{t=0}^{\ell-1} \mathsf{g}(x_t|\omega^j) - \bar{\mathsf{g}}(\cdot|\omega^j)\right)^2}$$

Table 1. A summary of the proposed landscape features.

description	sub-problem features: random walk	sub-problem features: adaptive walk	MO features
Fitness values	fv_rws_s	fv_aws_s	fv_rws_s_r ; fv_aws_s_r
	$s \in \{$avg, sd, r1, kr, sk$\}$		$r \in \{$avg, sd, c1, c2$\}$
Fitness differences	fd_rws_s	fd_aws_s	fd_rws_s_r ; fd_aws_s_r
	$s \in \{$avg, sd, min, max$\}$		$r \in \{$avg, sd, c1, c2$\}$
Improving neighbors	in_rws_s	in_aws_s	in_rws_s_r ; in_aws_s_r
	$s \in \{$avg, sd, min, max$\}$		$r \in \{$avg, sd, c1, c2$\}$
Walk length	—	law	law_r
			$r \in \{$avg, sd, c1, c2$\}$

– **Fitness difference (fd_*).** In the second class, we compute the average fitness difference with the neighboring solutions for every x_i, $i \in \{1, \ldots, \ell\}$: $\frac{1}{|\mathcal{N}(x_i)|} \sum_{x \in \mathcal{N}(x_i)} \big(\mathsf{g}(x_i|\omega^j) - \mathsf{g}(x|\omega^j)\big)$. Similarly, we consider the mean (avg), standard deviation (sd), minimum (min) and maximum (max) fitness difference over solutions from the walk.
– **Improving neighbors (in_*).** In the third class, we compute the proportional number of improving neighbors for each solution x_i, $i \in \{0, \ldots, \ell\}$. Then, we consider the mean, standard deviation, minimum, and maximum of this measure. It is worth noticing that the second and third classes of features require to evaluate the fitness value of neighbors from all solutions from the walk.
– **Length of the adaptive walk (law_*).** The fourth class only contains features extracted from the adaptive walk. In particular, we consider the length of the adaptive walk as a feature to characterize the sub-problem landscape. This length was shown to provide an estimation of the number of local optima in single-objective landscape analysis [5].

Aggregated Multi-objective Features. The features described above are computed for each sub-problem $j \in \{1, \ldots, \mu\}$, and then have a dimension μ. For $\mu > 1$, we need to aggregate these μ-dimensional single-objective features into 1-dimensional multi-objective features. To do so, we use two standard statistics, namely the mean (avg) and the standard deviation (sd). In addition, we use a polynomial regression in order to fit each single-objective feature as a function of the weight vector ω^j of sub-problem j. The coefficient of the polynomial model are then used as additional aggregated features. In this study, since we experiment bi-objective optimization problems, we consider a second order polynomial regression and propose to use the first (c1) and the second (c2) coefficients as additional multi-objective features. As summarized in Table 1, we end up with 4 aggregation statistics over respectively 5, 4, and 4 fv, fd, and in features, in addition to 4 aggregated features on the length of adaptive walk. This amounts to a total of 108 decomposition-based multi-objective features.

3 A Preliminary Exploratory Analysis

As a first step, we analyze the relevance of the proposed features in capturing the characteristics of multi-objective optimization problems regardless of any particular evolutionary search algorithm. Therefore, we conduct a preliminary exploratory analysis in order to highlight the association between the designed features and the properties of well-established benchmark landscapes.

3.1 Experimental Setup

Following previous work [8], we consider ρ`mnk`-landscapes [17] as a problem-independent model used for constructing multi-objective multimodal landscapes with objective correlation. Candidate solutions are binary strings of size $\mathtt{n}$. The objective function vector $f = (f_1, \ldots, f_i, \ldots, f_{\mathtt{m}})$ is defined as $f\colon \{0,1\}^{\mathtt{n}} \mapsto [0,1]^{\mathtt{m}}$ such that each objective f_i is to be maximized. The objective value $f_i(x)$ of a solution $x = (x_1, \ldots, x_j, \ldots, x_{\mathtt{n}})$ is an average value of the individual contributions associated with each variable x_j. Given objective f_i, and each variable x_j, a component function $f_{ij}\colon \{0,1\}^{\mathtt{k}+1} \mapsto [0,1]$ assigns a real-valued contribution for every combination of x_j and its $\mathtt{k}$ *epistatic interactions* $\{x_{j_1}, \ldots, x_{j_{\mathtt{k}}}\}$. These f_{ij}-values are uniformly distributed in $[0,1]$. The objective functions to be maximized can written as: $f_i(x) = \frac{1}{\mathtt{n}} \sum_{j=1}^{\mathtt{n}} f_{ij}(x_j, x_{j_1}, \ldots, x_{j_{\mathtt{k}}})$, $\forall i \in \{1, \ldots, \mathtt{m}\}$. In this work, the $\mathtt{k}$ epistatic interactions are set uniformly at random among the $(\mathtt{n}-1)$ variables other than x_j. By increasing the value of $\mathtt{k}$ from 0 to $(\mathtt{n}-1)$, problem instances can be gradually tuned from smooth to rugged. The f_{ij}-values additionally follow a multivariate uniform distribution of dimension $\mathtt{m}$, defined by an $\mathtt{m} \times \mathtt{m}$ positive-definite symmetric covariance matrix (c_{pq}) s.t. $c_{pp} = 1$ and $c_{pq} = \rho$ for all $p \neq q$ where $\rho > \frac{-1}{\mathtt{m}-1}$ defines the correlation among the objectives.

In our work, we focus on bi-objective landscapes, i.e., $\mathtt{m} = 2$. We use a latin hypercube sampling to generate a set of 1 000 balanced instances spanning parameters ranges: $\mathtt{n} \in \{50, 51, \ldots, 200\}$, $\mathtt{k} \in \{0, 1, 2, \ldots, 8\}$ and $\rho \in]-1, 1]$. The random walk length is set to $\ell = 1\,000$ across all problem sizes. A unique random walk is performed for *all* sub-problems, whereas one adaptive walk is performed for *each* sub-problem. The number of sub-problems is set to $\mu = 20$ and weight vectors are distributed uniformly; i.e., $\omega^j = ((j-1)/(\mu-1), 1-(j-1)/(\mu-1))$. The neighborhood relation is the standard 1-bit-flip.

For our analysis, we first conduct an exploratory analysis to better visualize and understand the proposed features. Next, we construct a regression model to study the accuracy of features in grasping the global properties of ρ`mnk`-landscapes, and we analyze the correlation among features.

3.2 Visual Analysis of Single-objective Features

In Fig. 1, we report the values of *single-objective* features as a function of sub-problems. Due to space restriction, we report a single representative feature for each of the four classes. The blue curves correspond to ρ`mnk`-landscapes with $\mathtt{k} = 0$, the green ones to $\mathtt{k} = 2$ and the red ones to $\mathtt{k} = 4$. The color scales from

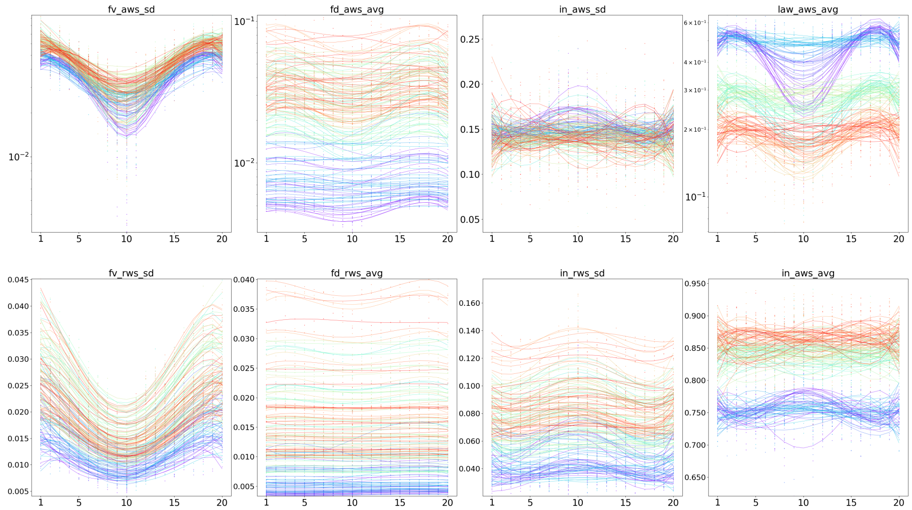

Fig. 1. Feature values of ρmnk-landscapes decomposed into 20 sub-problems (n $= 25$, each color correspond to a particular configuration of ρ and k).

red to orange, green to cyan, and blue to purple, respectively, and correspond to the objective correlation parameter ρ varying from 1 to -1; i.e., from highly correlated to highly conflicting objectives. For example, the standard deviation of fitness values from a random walk (fv_rws_sd, bottom left), the average fitness difference from a random walk (fd_rws_avg, second line, second column) and the standard deviation of improving neighbors (in_rdw_sd, second line, third column) gives a clear differentiation between landscapes with different k-values. The lower the benchmark parameter k, the lower the standard deviation of fitness values and the average fitness difference. Similarly, the standard deviation of fitness values from an adaptive walk (vf_aws_sd, top left) and the length of an adaptive walk (law_aws_avg, top right) seems to be clearly associated with parameter ρ. The flatter the curve rendering the evolution of these two features as a function of weight vectors, the higher the objective correlation ρ.

From our visual inspection, we can conclude that the landscape features are representative of the different *global* benchmark parameters, which are *unknown* in a black-box optimization scenario. However, this first analysis considers the single-objective features and not the aggregated 1-dimensional multi-objective features, which are discussed next.

3.3 Correlation Analysis of Features and Landscape Parameters

Investigating the accuracy of the designed multi-objective features, we consider a typical machine learning task consisting in predicting the value of the (unknown) global benchmark parameters k and ρ. We respectively construct a random forest classification model and a random forest regression model [2], using the whole set

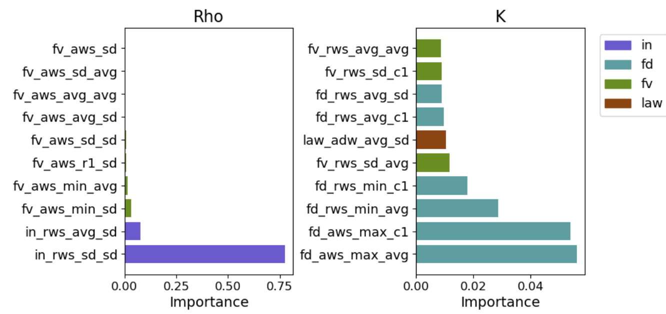

Fig. 2. Relative importance of features to predict ρ (left) and k (right).

of multi-objective features computed over all considered ρmnk-landscapes. Random forest has the nice property of providing a measure of feature importance for model fitting. In Fig. 2, we report the relative importance of each feature extracted from the random forest models, using the Gini impurity as a measure of quality. Values are averaged over 10 independent repetitions of model fitting.

The first notable observation is that feature importance is different depending on whether we aim at predicting benchmark parameter ρ or k. The objective correlation ρ appears mostly related to a single feature: the standard deviation of the number of improving neighbors (in_rws_sd). By contrast, deciding on parameter k is related to multiple different features, that mostly correspond to the fitness difference computed from adaptive walks, in particular the maximum fitness difference (fd_aws_max). To push the analysis further, we show in Fig. 3 the Spearman correlation matrix among a subset of features as well as benchmark parameters ρ and k. It is worth noticing that we do not show all 108 features, but only a subset of 60 features with representative correlation values due to space restriction. Interestingly, we can distinguish four clusters, denoted C_1, C_2, C_3 and C_4 in the figure. The largest cluster C_1 contains more than 30 features. There is a positive correlation between the first auto-correlation coefficient of fitness values vf_r1 and k, of at least 0.25. This cluster also contains a small subset of 2 features related to the number of improving neighbors in_*, with a particularly high correlation value (> 0.8) (the 3×3 red points at the center of the figure). In the second cluster C_2, we observe a large subset of features being negatively correlated with k (the dark blue columns/lines intersecting in a completely red square in the middle-right of the figure). For both clusters related to k (C_1 and C_2), features are not extracted from a unique class. This means that features from different classes can be used to characterize k.

The remaining 2 clusters (C_3 and C_4) are associated with the second benchmark parameter ρ. In particular, besides the average number of improving neighbors (in_rws_avg), cluster C_3 also contains the standard deviation (in_rws_sd), both computed from a random walk. This latter feature, that was found to be the most important to predict ρ in Fig. 2, has a nearly perfect negative correlation with ρ, equals to -1. The last cluster C_4 contains features based on the

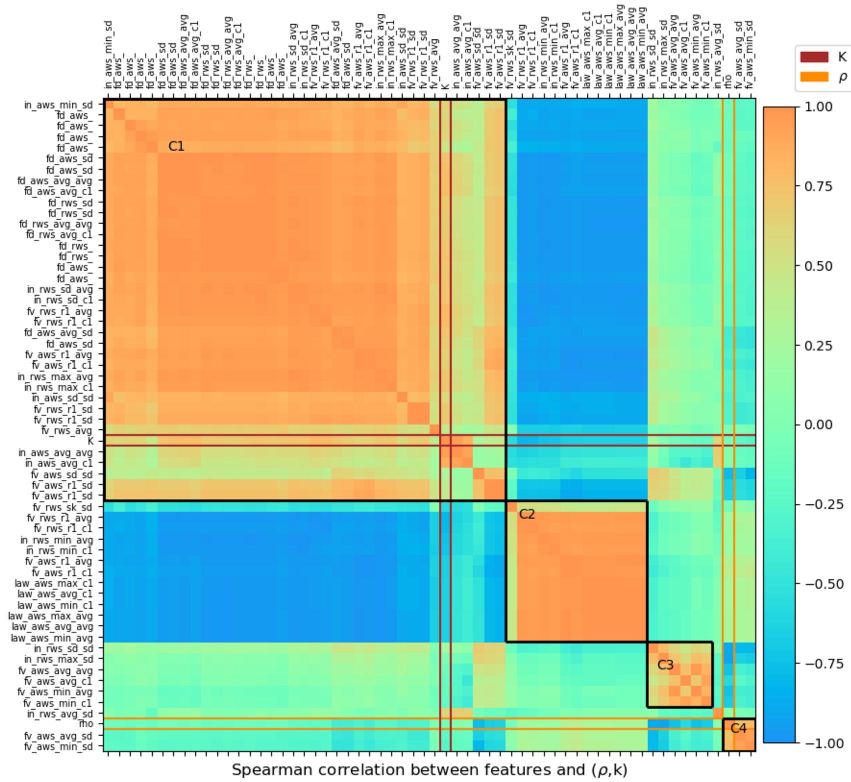

Fig. 3. Pairwise correlation among a subset of features.

fitness values computed over adaptive walks (fv_aws), which appear to be the most positively correlated with ρ (3×3 red points at the bottom right of the figure), with a correlation higher than 0.7.

To summarize, we find that all feature classes are useful to characterize the (unknown) global benchmark parameter k, rendering the degree of non-linearity of the problem. However, the fitness distribution (fv) and the number of improving neighbors (in) classes have more importance than the fitness difference (fd) and the length of adaptive walks (law) for characterizing the (unknown) global benchmark parameter ρ, relating to objective correlation.

4 Landscape-aware MOEA/D Selection

In this section, we conduct a second set of experiments in order to study the accuracy of the designed features when integrated into an automated algorithm selection approach. We consider the more sophisticated task of selecting the best performing algorithm among an algorithm portfolio. More precisely, we consider three variants of the well-established MOEA/D algorithm [18,12] as a case study.

In the following, we start by briefly describing the considered portfolio before addressing our main target task.

4.1 Algorithm Portfolio

As mentioned earlier, the MOEA/D algorithm is based on a flexible decomposition-based framework that can be configured in different manners. In its baseline variant [18], MOEA/D first decomposes the problem into a number of scalarized sub-problem, as discussed previously. Then, a solution is evolved for each sub-problem in a cooperative way. The algorithm iterates over the sub-problems and, at each iteration, an offspring is generated by means of crossover and mutation on the basis of parent solutions selected from the so-called T-neighborhood; i.e., the sub-problems corresponding to the T closest weights in the objective space. The new offspring can then replace any of the sub-problem solutions in the T-neighborhood of the current sub-problem. This corresponds to a standard evolutionary process, where selection and replacement are performed iteratively over sub-problems. In [12], it is shown that the selection and replacement underlying the standard MOEA/D framework are key algorithm components that highly impact performance. Several generational variants are proposed therein, allowing to tune the selection and replacement underlying the MOEA/D framework from fully cooperative (i.e., among *all* sub-problems) to fully selfish (i.e., independently of any other sub-problem).

Interestingly, it was found that no variant outperforms the other independently of the global benchmark parameter values ρ and k for the considered ρmnk-landscapes. Since such parameters are unknown in a black-box optimization scenario, the study presented in [12] leaves open the challenging question of which variant to choose in an automated manner. In addition, this constitutes a perfect and typical setting for the main automated algorithm selection task addressed in this paper. We consider, in the following, three representative variants of the MOEA/D framework, exposing different degrees of cooperation among sub-problem solving. For reproducibility, and in order to be consistent with the notations from [12], these variants are denoted as follows: (i) MOEA/D, referring to the standard variant [18], (ii) MOEA/D-SC, a generational variant where selection is performed in a selfish manner for every sub-problem whereas replacement is performed in a cooperative manner, and (iii) MOEA/D-SS, a (selfish) generational variant exposing the lower degree of cooperation among sub-problems. Besides population size, the three variants have the same set of parameters: $\delta = 1$, $nr = 2$, $p_{mut} = \frac{1}{n}$ and $p_{cr} = 1$. Due to space restriction, we refer to [12] for a full description of these MOEA/D variants.

In order to highlight the relevance of this portfolio in studying the accuracy of our features when integrated into a landscape-aware algorithm selection approach, we briefly report their relative performance using exactly the same set of ρmnk-landscapes as in the previous section. Every algorithm is executed 20 times on each instance, using a population size equals to n, and a budget of 10^6 evaluations. The performance of an algorithm is computed as its hypervolume relative deviation w.r.t. the best-found approximation set for each instance. The

Table 2. Performance matrix of the three Moea/d variants. The diagonal reports the number of times where the corresponding algorithm is statistically outperformed by another one (the lower the better). The other cells report how many instances (out of 1000) the algorithm in the corresponding line is statistically better than the algorithm in the corresponding column (the higher the better).

	Moea/d-sc	Moea/d	Moea/d-ss
Moea/d-sc	205	18	310
Moea/d	85	137	312
Moea/d-ss	120	119	622

hypervolume measures the area covered by an approximation set and enclosed by a reference point [19]. For a given instance, the reference point is set to the best value seen across all runs for each objective. We then count the number of times an algorithm is statistically outperformed using a two-sided Mann-Whitney test with a p-value of 0.05 and a Bonferroni correction. Results are reported in Table 2.

We clearly see that each algorithm is outperformed by another one on a subset of instances. The basic Moea/d variant seems to have a reasonably good behavior, since it is less-often outperformed than the two other variants overall (see diagonal). A more detailed analysis, omitted due to space restriction, shows that there is a complex interaction between algorithm performance and the benchmark parameters ρ and `k`which can be summarized as follows: (i) the smaller `k` and ρ, the better Moea/d and Moea/d-sc against Moea/d-ss, (ii) the larger ρ (highly correlated), the better Moea/d-ss, and (iii) the larger `k` and the smaller ρ, the better Moea/d-sc. Of course, this general trend has some exceptions, but it shows the impact of the unknown benchmark parameters on the relative performance of algorithms.

4.2 Automated Algorithm Selection

Task and Experimental Methodology. We study the accuracy of the proposed features by investigating the selection of the best performing Moea/d variant. For this purpose, we adopt the following standard supervised-learning approach. We first train three models in order to predict the performance of every considered Moea/d variant. We use the average hypervolume deviation as defined in the previous section as a measure of algorithm performance on a given instance, which then corresponds to our output prediction variable. Considering an unseen test instance, the landscape features are first computed, the performance of each algorithm is then predicted on the basis of the trained models, and the algorithm having the best prediction is selected as the recommended one. We consider the same set of ρ`mnk`-landscapes described in the previous section. We adopt a standard validation methodology where an instance is selected for training with probability 0.9 and for testing with probability 0.1. We use a set of 100 random regression trees to learn and predict the expected relative hypervolume deviation. Reported values are computed over 50 independent runs.

We recall that the proposed multi-objective features rely on some weight vectors μ. We consider a variable number of weight vectors in the range $\mu \in \{1, 2, 3, 4, 5, 6, 10, 20\}$. Additionally, we consider two alternatives for generating weight vectors, namely uniform or random. In a random setting, a weight vector is generated uniformly at random. In a uniform setting, the weight vector are evenly distributed in the objective space. In particular, for $\mu = 1$, the weight vector selected in the uniform setting is $(0.5, 0.5)$; i.e., the "middle" of the objective space. In this case, the multi-objective features are simply the same than the corresponding single-objective features from the single scalarized sub-problem. For $\mu = 2$, our uniform setting corresponds to weight vector $(1, 0)$ and $(0, 1)$. This means that our features are obtained by aggregating the single-objective features computed independently for each objective of the original multi-objective problem. The impact of this setting is carefully analyzed in our experiments.

Prediction Accuracy. In order to assess the prediction accuracy, we compute three complementary measures. The first two directly relate to the prediction error: the percentage of times the selected algorithm does not have the best hypervolume deviation in average, and the percentage of times the selected algorithm is statistically outperformed by at least one other algorithm. The third indicator, which is a straightforward adaptation from [11], measures the gap between: (i) the performance of the single best solver (SBS) having the best performance in average over the training set (without model training), and (ii) the performance of the virtual best solver (VBS), obtained by a model that would make perfect predictions. More precisely, let I_{train} and I_{test} be the set of training and testing instances, respectively, and let $\mathsf{rhf}(A, i)$ be the average relative hypervolume deviation of a given algorithm $A \in \mathcal{A} = \{\text{MOEA/D}, \text{MOEA/D-SS}, \text{MOEA/D-SC}\}$ on instance $i \in I_{train} \cup I_{test}$. For every algorithm $A \in \mathcal{A}$ and instance subset J, let $\overline{\mathsf{rhf}}(A, J) = \frac{1}{|J|} \sum_{i \in J} \mathsf{rhf}(A, i)$. We define SBS as the algorithm having the best $\overline{\mathsf{rhf}}$ value on the training set I_{train}, i.e., $SBS = \arg\min_{A \in \mathcal{A}} \left\{ \overline{\mathsf{rhf}}(A, I_{train}) \right\}$. We define VBS as the virtual 'algorithm' obtained by a perfect prediction model (an oracle); i.e., the algorithm with the best $\mathsf{rhf}(\cdot, i)$ value for each $i \in I_{test}$. Finally, let Recommended Solver (RS) be the algorithm predicted by the actual trained model. The *merit* indicator is:

$$M = \frac{\overline{\mathsf{rhf}}(RS, I_{test}) - \overline{\mathsf{rhf}}(VBS, I_{test})}{\overline{\mathsf{rhf}}(SBS, I_{test}) - \overline{\mathsf{rhf}}(VBS, I_{test})}$$

It should be clear that: (i) A merit of 0 indicates that the model does not make any error, (ii) A merit in the range $[0, 1[$ indicates that the model is more efficient than the SBS but worse than the VBS, (iii) a merit greater than 1 indicates that the model is worse than the SBS. Achieving a merit value of 0 is clearly a very challenging task and one seeks for a merit value below 1 (better than the SBS) and as close as possible to 0 (the VBS).

Experimental Results and Discussion. Our main results are summarized in Fig. 4 showing the accuracy indicators as a function of the number of weight

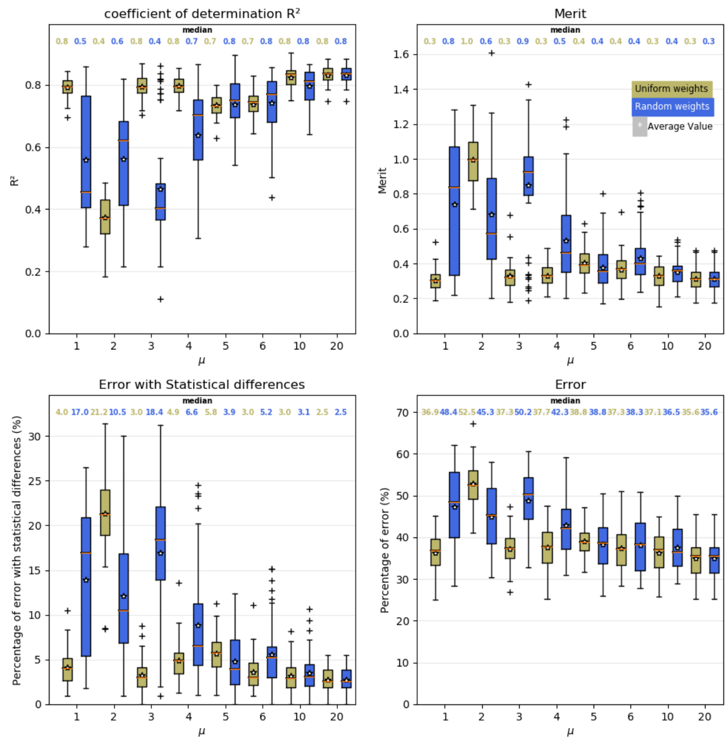

Fig. 4. R^2 (top left), merit (top right) and error rates (bottom) according to the number of weights μ and their distribution.

vectors μ and their type (random or uniform). For completeness, we also show the R^2 coefficient obtained by the training models. Different observations can be extracted from Fig. 4.

We first clearly see that the choice of the weight vector distribution is of critical importance. In fact, a random choice does not obtain a good accuracy, except when the number of weights μ is substantially large. By contrast, a uniform strategy appears to perform reasonably well, even when the number of weights is low. Interestingly, for uniform weights, the worst accuracy is obtained with $\mu = 2$. Notice that such a setting is even substantially outperformed by a random choice of weight vectors. This indicates that computing single-objective features independently for each objective is not a recommended strategy. By contrast, computing features for decomposed sub-problems is effective even when using a very low number of weights. This indicates that a decomposition-based approach for multi-objective landscape analysis contains a valuable information about algorithm performance. Surprisingly, we found that a uniform choice of few weight vectors with $\mu \in \{1, 3\}$ performs reasonably well, although increasing $\mu > 3$ allows to obtain a better accuracy. The relatively good results achieved with $\mu = 1$ are however to be interpreted very carefully, taking into account that the shape of the Pareto front for ρmnk-landscapes, although having different magnitude in the objective space, is convex, symmetric and centered in the middle of the objective space, regardless of the values of ρ and k [17]. Although this is a recurrent observation for many multi-objective combinatorial optimization problems, one might need to carefully choose μ when tackling problems with different Pareto front shapes. Such considerations are left for future investigations.

At last, in order to further show the accuracy of the proposed multi-objective landscape features, we experiment a baseline random forest model using the (unknown) global benchmark parameters ρ and k as input variables to predict algorithm performance. Contrary to a black-box scenario where the knowledge about ρ and k is not available, the accuracy of such a 'white-box' model should highlight the relevance and reliability of the proposed black-box features. We found that such a model trained with ρ and k obtains an average merit of 0.41. Comparatively, black-box features obtain an average merit of 0.31, 0.37 and 0.29 respectively, for $\mu \in \{1, 3, 20\}$ uniform weight vectors. This once again indicates that the proposed approach is very effective, and that the designed high-level black-box features seem to provide more accurate prediction models than the global benchmark parameters, hence allowing to substantially improve over the single best solver, and also to get closer to the ideal virtual best solver.

5 Conclusion and Open Issues

In this paper, we push a step towards the development of automated landscape-aware selection and configuration approaches by proposing a set of multi-objective landscape features and analyzing their effectiveness in grasping the global properties of black-box multi-objective combinatorial optimization problems, together with their efficiency in selecting the best performing algorithm among three variants of the MOEA/D state-of-the art algorithm. The proposed features are based on the simple idea of aggregating the single-objective features extracted from a number of sub-problems obtained by decomposition. Our empirical analysis on a wide range of ρmnk-landscapes provides insights into the accuracy of the proposed approach. However, it also raises some interesting research questions.

For instance, problems with more than two objectives, where the distribution of weight vector is expected to play an even more important role, as well as other real-like problems, are to be studied. Moreover, we excluded the cost of feature computation from our analysis, which can be an issue when the total affordable budget for the optimization task is restricted. An interesting observation is that we only need to extract features from a relatively low number of sub-problems, as considering a single sub-problem is already shown to provide a reasonably good performance. Additionally, solely one random walk is required to compute the considered single-objective features, so that one can eventually end up with negligible cost features. In this respect, a more fine-grained analysis of the cost-vs-importance of features is to be conducted in our future investigations. Finally, an interesting question is to conduct a systematic analysis of the proposed decomposition-based features w.r.t. existing dominance- and indicator-based features, and to analyze their relative cost and accuracy. Given that decomposition-based features were shown to be effective and does not need any (costly) dominance- and indicator-based computations, it is our hope that unifying the two classes of features would allow us to end up with efficient high-level state-of-the-art features for landscape-aware multi-objective optimization.

References

1. Beham, A., Wagner, S., Affenzeller, M.: Algorithm selection on generalized quadratic assignment problem landscapes. In: GECCO 2018. pp. 253–260
2. Breiman, L.: Random forests. Machine Learning **45**(1), 5–32 (2001)
3. Deb, K.: Multi-Objective Optimization using Evolutionary Algorithms (2001)
4. Grefenstette, J.J.: Predictive models using fitness distributions of genetic operators. In: Foundations of Genetic Algorithms, vol. 3, pp. 139–161. Elsevier (1995)
5. Kauffman, S.A.: The Origins of Order. Oxford University Press (1993)
6. Kerschke, P., Trautmann, H.: The r-package flacco for exploratory landscape analysis with applications to multi-objective optimization problems. In: CEC. pp. 5262–5269 (2016)
7. Kerschke, P., Hoos, H.H., Neumann, F., Trautmann, H.: Automated algorithm selection: Survey and perspectives. Evol. Comput. **27**(1), 3–45 (2019)
8. Liefooghe, A., Daolio, F., Verel, S., Derbel, B., Aguirre, H., Tanaka, K.: Landscape-aware performance prediction for evolutionary multi-objective optimization. IEEE Transactions on Evolutionary Computation (2020)
9. Liefooghe, A., Derbel, B., Verel, S., Aguirre, H.E., Tanaka, K.: Towards landscape-aware automatic algorithm configuration: Preliminary experiments on neutral and rugged landscapes. In: EvoCOP. pp. 215–232 (2017)
10. Liefooghe, A., Verel, S., Derbel, B., Aguirre, H.E., Tanaka, K.: Dominance, indicator and decomposition based search for multi-objective QAP: landscape analysis and automated algorithm selection. In: PPSN XVI. pp. 33–47 (2020)
11. Lindauer, M., van Rijn, J.N., Kotthoff, L.: The algorithm selection competition series 2015-17. CoRR **abs/1805.01214** (2018)
12. Marquet, G., Derbel, B., Liefooghe, A., Talbi, E.: Shake them all! - rethinking selection and replacement in MOEA/D. In: PPSN XIII. pp. 641–651 (2014)
13. Mersmann, O., Bischl, B., Trautmann, H., Wagner, M., Bossek, J., Neumann, F.: A novel feature-based approach to characterize algorithm performance for the traveling salesperson problem. Ann Math Artif Intel **69**, 151–182 (2013)
14. Rice, J.R.: The algorithm selection problem. Adv Comput **15**, 65–118 (1976)
15. Richter, H., Engelbrecht, A. (eds.): Recent Advances in the Theory and Application of Fitness Landscapes. Emergence, Complexity and Computation, Springer (2014)
16. Trivedi, A., Srinivasan, D., Sanyal, K., Ghosh, A.: A survey of multiobjective evolutionary algorithms based on decomposition. IEEE TEVC **21**(3), 440–462 (2017)
17. Verel, S., Liefooghe, A., Jourdan, L., Dhaenens, C.: On the structure of multi-objective combinatorial search space: MNK-landscapes with correlated objectives. European Journal of Operational Research **227**(2), 331–342 (2013)
18. Zhang, Q., Li, H.: MOEA/D: A multiobjective evolutionary algorithm based on decomposition. IEEE TEVC **11**, 712 – 731 (01 2008)
19. Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C.M., Grunert da Fonseca, V.: Performance assessment of multiobjective optimizers: An analysis and review. IEEE TEVC **7**(2), 117–132 (2003)