LIEFOOGHE, A., VEREL, S., LACROIX, B., ZĂVOIANU, A.-C. and MCCALL, J. 2021. Landscape features and automated algorithm selection for multi-objective interpolated continuous optimisation problems. In Chicano, F. (ed) Proceedings of 2021 Genetic and evolutionary computation conference (GECCO 2021), 10-14 July 2021, [virtual conference]. New York: ACM [online], pages 421-429. Available from: https://doi.org/10.1145/3449639.3459353

Landscape features and automated algorithm selection for multi-objective interpolated continuous optimisation problems.

LIEFOOGHE, A., VEREL, S., LACROIX, B., ZĂVOIANU, A.-C. and MCCALL, J.

2021
In this paper, we demonstrate the application of features from landscape analysis, initially proposed for multi-objective combinatorial optimisation, to a benchmark set of 1,200 randomly-generated multi-objective interpolated continuous optimisation problems (MO-ICOPs). We also explore the benefits of evaluating the considered landscape features on the basis of a fixed-size sampling of the search space. This allows fine control over cost when aiming for an efficient application of feature-based automated performance prediction and algorithm selection. While previous work shows that the parameters used to generate MO-ICOPs are able to discriminate the convergence behaviour of four state-of-the-art multi-objective evolutionary algorithms, our experiments reveal that the proposed (black-box) landscape features used as predictors deliver a similar accuracy when combined with a classification model. In addition, we analyse the relative importance of each feature for performance prediction and algorithm selection.

1 INTRODUCTION

Although the idea of feature-based performance prediction and algorithm selection is not new [27], research interest and effort has intensified over the past few years [2, 14, 30]. With the need to provide a better understanding of problem structure and complexity, various works have provided the community with some interesting sets of quantifiable features evaluating different aspects of the landscape, such as ruggedness or multimodality [16, 23, 24]. These landscape features not only provide a means for analysing the behaviour of search algorithms, but also provide an opportunity to select a priori the most appropriate algorithm for unseen problems. Such a methodology relies on two complementary components: (i) a set of landscape features that characterises the space sufficiently so as to discriminate the performance of given algorithms, and (ii) a comprehensive set of benchmark problems to supply a solid knowledge base for training and testing predictive models.

In this paper, we combine two recent works aimed at providing those two components in the context of continuous multi-objective optimisation problems (MOOPs). The first work [21] provides a comprehensive set of landscape features for multi-objective combinatorial optimisation. These features characterise different facets of difficulty, based on local measures computed over an affordable sample of solutions. In fact, while there exists a substantial literature on the landscape of single-objective continuous optimisation problems, features for continuous MOOPs are lacking [14, 22]. We notice, however, that features from single-objective optimisation have been used to characterise MOOPs [15], although single-objective features certainly do not capture intrinsic properties of multi-objective landscapes, such as interaction among objectives. Here, we propose to transfer the multi-objective landscape features developed in [21] from problems with a combinatorial search space to problems with a continuous search space. In particular, the random and adaptive walks used for sampling are replaced by a fixed-cost Latin hypercube sampling. This allows cost efficient application (in terms of evaluations) to automated algorithm selection and performance prediction. The second work this paper builds upon outlines a method for generating a diversified benchmark set of continuous MOOPs [34]. The generator uses interpolated continuous optimisation problems to produce MOOPs with various characteristics and difficulties.
structures. In total, 1 200 bi-objective problems were investigated on four well-established multi-objective evolutionary algorithms and discriminated the performance between those algorithms. Interestingly, the parameters used to generate the problems have some predictive power on algorithm performance. We evaluate our proposed multi-objective landscape features on these continuous problems, and provide a comprehensive analysis of the relationship between landscape features and algorithm performance. In particular, we analyse the correlation between the absolute performance of each algorithm in terms of hypervolume and the considered features. We then use the features as predictors for performance prediction and algorithm selection on random forest regression and classification models, respectively. We also analyse the predictive power of the proposed landscape features against the parameters used by the random multi-objective interpolated continuous optimisation problem generator from [34].

The paper is organised as follows. Section 2 recalls the construction of multi-objective interpolated continuous optimisation problems. Section 3 gives the experimental setup of our analysis. Section 4 introduces a number of features for characterising continuous multi-objective landscapes, and correlates them with the performance of multi-objective evolutionary algorithms. Section 5 analyses an automated algorithm selection approach using the proposed landscape features as predictors. The last section concludes the paper and discusses further research.

2 MULTI-OBJECTIVE INTERPOLATED CONTINUOUS OPTIMISATION PROBLEMS

2.1 Multi-objective Optimisation

Let us consider an objective function vector $F: X \mapsto Z$ to be minimised, such that $X \subseteq \mathbb{R}^d$ is the variable space and $Z \subseteq \mathbb{R}^m$ is the objective space. Each solution $x \in X$ maps to an objective vector $z \in Z$ such that $z = F(x)$. Given two objective vectors $z', z'' \in Z$, $z'$ is dominated by $z''$ iff for all $i \in \{1, \ldots, m\}$ $z'_{i} \leq z_{i}$, and there exists 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 $z^* \in Z$ is non-dominated if there does not exist any $z \in Z$ such that $z^*$ is dominated by $z$. A solution $x^* \in X$ is Pareto optimal, or non-dominated, if $F(x^*)$ is non-dominated. The set of all Pareto optimal 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 optimisation is to identify the PS, or a good approximation of it for large-size and complex problems. A number of multi-objective evolutionary algorithms (MOEAs) have been designed to this end since the late eighties [6, 7].

2.2 Problem Definition

Multi-objective interpolated continuous optimisation problems (MO-ICOPs) have been recently introduced in [34] as a new class of benchmark problems with tunable difficulty. MO-ICOPs are defined by an objective function vector that combines $m$ single-objective ICOPs generated using a common set of seeds. A MO-ICOP can be formulated as follows:

$$\text{minimise } F_{S, U_{1}, \ldots, U_{m}}(x) = (f_{S, U_{1}}(x), \ldots, f_{S, U_{m}}(x))^T,$$

where each $f_{S, U_{i}}(x)$ represents a single-objective ICOP defined by:

- **A set of seeds $S \subset X$:** a (typically finite) set of distinct candidate solutions with an assigned objective value. Elements of $S$ and their assigned objective values $U_{i}$ will define the entire optimisation problem via interpolation.
- **An interpolation function** $(f_{S, U_{i}}: X \mapsto \mathbb{R})$: in this paper, we apply the inverse distance weighting method, originally defined by Shepard for spatial analysis [28]. Assuming the seed set $S$ contains $N$ seeds, labelled $S = \{s_1, \ldots, s_N\}$, with the assigned objective values $U_{i} = \{u_{i,1}, \ldots, u_{i,N}\}$, we define for any candidate solution $x \in X$:

$$f_{S, U_{i}}(x) = \frac{\sum_{j=1}^{N} \frac{u_{i,j}}{d_{i,j}(x)^k}}{\sum_{j=1}^{N} \frac{1}{d_{i,j}(x)^k}} \quad \text{if } e(x, s_j) \neq 0 \text{ for all } j \quad (2)$$

where $e(x, y) : X \times X \mapsto \mathbb{R}$ is the Euclidean distance, and $k$ is a positive real number called the power parameter. Higher values of $k$ increase the relative influence of nearby seeds on the interpolated value.

3 EXPERIMENTAL SETUP

3.1 Benchmark Dataset

We consider the set of bi-objective problems proposed in [34]. For each MO-ICOP, the set of seeds $S$ is randomly generated in the variable space $X = [-5, 5]^d$. The objective vector of each seed is first assigned to a subset of non-dominated seeds $S_{nd}$, ensuring that no seed in $S_{nd}$ dominates another seed from $S_{nd}$. Objective values are then assigned to the remaining dominated seeds $S_d$, ensuring that each seed in $S_d$ does not dominate any seed in $S_{nd}$.

Using this method, 50 problems were generated in 4 problem dimensions $d \in \{5, 10, 20, 30\}$ and 6 values for $k \in \{1, 2, 3, 4, 5, 6\}$, creating a total of 1 200 problems covering a wide range of MO-ICOPs. We emphasise that, although the problem dimension is typically given, the seeds and the power of interpolation are unknown for unseen (black-box) problems.

3.2 Considered Algorithms

By analysing the outputs of the experiments reported in [34]1, the present work focusses on the comparative performance of four different MOEAs parameterised using recommended settings from the literature. In [34], the authors state that these MOEAs were chosen because they exemplify different well-proven multi-objective optimisation strategies. Thus, NSGA-II [8] pioneered alongside SPEA2 [38] is an effective evolutionary approach grounded in Pareto-based elitism (as a primary criterion) and objective space distancing (as a tie-breaker). GDE3 [19] also uses this strategy, but replaces the simulated binary crossover (SBX) and polynomial mutation (PM) variation operators [7] popularised by NSGA-II with a differential evolution (DE) strategy: DE/rand/1/bin. MOEA/D-DE-TRA [36] is a state-of-the-art solver that also integrates a DE strategy, but its main characteristic is that it applies a weighting-based aggregation of the objectives to decompose the original MOOP into multiple single-objective sub-problems that are solved concurrently during a single optimisation run. Finally, DECMO2++ [35]

---

1Accessible at: https://github.com/czavoianu/PiSN_2020
was designed to achieve a fast convergence across a wide range of benchmark MOOPs by actively pivoting between three search strategies implemented via two actively co-evolved sub-populations (based on SPEA2 and GDE3) and a largely passive decomposition-based archive. In the case of NSGA-II, GDE3 and DECMO2++, a population / archive size of 200 was used. For MOEAD/DE-DRA, the population size was set to 300, the standard setting for MOOPs with two objectives.

3.3 Budgets and Search Performance
A total computational budget of 50 000 evaluations was allowed for each optimisation run. As 100 independent runs were carried out for each solver on each MO-ICOP, the numerical experiments entailed a total of 480 000 independent optimisation runs across the 1 200 MO-ICOPs from the benchmark set. Search performance is assessed after 10 000, 20 000 and 50 000 calls to the evaluation function, using a normalised version of the hypervolume indicator [37] that quantifies the area of the objective space the MOEA population / archive dominates when compared against the true PF for the problem under consideration. A normalised hypervolume of 1 actually means that the MOEA hit the PF.

When considering end-of-the-run optimisation results across the 1 200 benchmark problems, MOEAD/DE-DRA achieved the best (i.e., highest) average hypervolume on 545 problems (≈ 45.4%), NSGA-II won on 509 problems (≈ 42.4%), while GDE3 and DECMO2++ scored 125 and 21 wins, respectively.

4 FEATURE-BASED LANDSCAPE ANALYSIS
In this section, we define a number of landscape features for characterising MO-ICOPs. In particular, we draw inspiration from the features introduced in [21] for combinatorial multi-objective landscapes. We start by describing them, and how we adapt them to continuous multi-objective landscapes. Then, we show how they individually correlate with the performance of the considered MOEAs. Finally, we measure their relative importance when predicting the performance of each MOEA for solving MO-ICOPs.

4.1 Features for Continuous Multi-objective Landscapes
The considered landscape features are based on different measures computed over a sample of solutions extracted from the problem under consideration. In order to gather information from a given continuous multi-objective landscape, we first sample a reasonable number of n solutions from the search space. The sample P is constructed according to a random latin hypercube design [5]. Sampled points are then evaluated by means of the objective function vector F, and the n evaluated solutions constitute the total budget devoted to the computation of landscape features. For each solution x ∈ P, we additionally define its d closest solutions from the sample other than x as its neighbours, d being the problem dimension. The closeness is measured as the Euclidean distance in the variable space. Based on the sample of size n and the definition of neighbourhood, we introduce four classes of landscape features below. In addition, we consider five problem-dependent features coming from the construction and mathematical formulation of MO-ICOPs, as discussed in Sect. 3.1. These features will be used as reference in the coming analysis on the predictive performance of landscape features in a black-box context. The 49 landscape features and 5 problem-dependent features are summarised in Tab. 1.

4.1.1 Global Landscape Features. The first class of landscape features is not based on the neighbourhood, and renders global properties from the continuous multi-objective landscape. This class ranges from the (Spearman) correlation among the objective values measured on the sample, to the average and maximum distance among sampled solutions in the variable space and in the objective space. We also measure the proportion of non-dominated solutions among the sample, the proportion of supported points [10] therein, the hypervolume covered by these non-dominated solutions, the average and maximum distance among non-dominated solutions in the variable space, as well as the (Spearman) correlation between the distance among them in the variable space vs. in the objective space — denoted as fitness-distance-correlation in [18]. Finally, we divide the sample into different ranks following the principles of non-dominated sorting used, e.g., in NSGA-II [8], and we measure the average and maximum rank [1, 11] as well as the entropy of the distribution of the number of solutions per rank.

4.1.2 Features Characterising the Landscape Multimodality. In terms of multimodality, we define a single-objective local optimum (slo) in the sample as a solution with no improving neighbour for a given objective. Similarly, we define a Pareto local optimum (plo) as a sampled solution for which there does not exist any dominating neighbour [26]. We measure the proportions of slo and plo in the sample, and the average and maximum distances among them in the variable space. We also measure how many (globally) non-dominated points in the sample there are, on average, per plo.

In single-objective landscape analysis, an adaptive walk is often performed to estimate the multimodality: assuming isotropy in the landscape, the longer the length of an adaptive walk, the larger the basins size, hence the lower the number of local optima [13]. We here perform a simple multi-objective adaptive walk, where the neighbourhood is explored from the closest to the furthest neighbour w.r.t. the current solution, and where the first dominating neighbour is accepted at each step [32]. The length (length_aws) corresponds to the number of steps performed until no further improvement is possible, and the walk falls into a plo. We also record the number of solutions explored along the walk (eval_aws). Multiple adaptive walks are typically performed from different starting points to improve feature estimation. We report below average values from 30 independent adaptive walks. It is important to note that each adaptive walk is performed over the pre-computed sample, and does not require any additional evaluation.

4.1.3 Features Characterising the Landscape Evolvability. Let us now quantify the expected improvement that can be achieved from a solution’s neighbourhood as measures of evolvability [29] for multi-objective optimisation. For each solution from the sample, we measure the proportions of dominating (sup), dominated (1nf), and incomparable (1nc) neighbours. We also measure the proportion of locally non-dominated neighbours (1nd), as well as the proportion of locally supported solutions therein (1sup). In addition, we compute the average distance from each solution to its neighbours in the variable space and in the objective space (dist_x and dist_f, etc.)
### Table 1: Considered problem-dependent and landscape features for continuous multi-objective optimisation.

| name                  | description                                                                                      |
|-----------------------|--------------------------------------------------------------------------------------------------|
| **problem-dependent** |                                                                                                 |
| d                     | number of variables                                                                              |
| k                     | power of interpolation                                                                          |
| seed_n                | proportional number of seeds | S |
| nd_seed_n             | proportion of non-dominated seeds | S_d |
| dom_seed_n            | proportion of dominated seeds | S_d |
| f_cor                 | correlation among objective values                                                                |
| dist_x_avg            | average distance among solutions in the variable space                                             |
| dist_x_max            | maximum distance among solutions in the variable space                                              |
| dist_f_avg            | average distance among solutions in the objective space                                              |
| dist_f_max            | maximum distance among solutions in the objective space                                              |
| nd_n                  | proportion of non-dominated solutions                                                               |
| supp_n                | proportion of supported non-dominated solutions                                                    |
| hv                    | hypervolume of non-dominated solutions                                                                |
| dist_x_nd_avg         | average distance among non-dominated solutions in the variable space                                |
| dist_x_nd_max         | maximum distance among non-dominated solutions in the variable space                                |
| fdc                   | fitness-distance correlation among non-dominated solutions                                           |
| rank_avg              | average rank w.r.t. non-dominated sorting                                                           |
| rank_max              | maximum rank w.r.t. non-dominated sorting                                                           |
| rank_ent              | entropy of the number of solutions per rank w.r.t. non-dominated sorting                           |
| **global**            |                                                                                                 |
| slo_n                 | proportion of single-objective local optima per objective                                         |
| slo_dist_avg          | average distance among single-objective local optima in the variable space                          |
| slo_dist_max          | maximum distance among single-objective local optima in the variable space                          |
| plo_n                 | proportion of Pareto local optima                                                                 |
| plo_dist_avg          | average distance among Pareto local optima in the variable space                                   |
| plo_dist_max          | maximum distance among Pareto local optima in the variable space                                   |
| rd_per_plo            | proportion of non-dominated solutions per Pareto local optimum (i.e. nd_n / plo_n)               |
| length_aws            | average length of adaptive walks                                                                   |
| eval_aws              | average number of calls to the evaluation function performed by adaptive walks                   |
| **multimodality**     |                                                                                                 |
| sup_avg_neig          | average proportion of dominating neighbours                                                        |
| inf_avg_neig          | average proportion of dominated neighbours                                                         |
| inc_avg_neig          | average proportion of incommensurable neighbours                                                    |
| lsupp_avg_neig        | average proportion of supported locally non-dominated neighbours                                   |
| dist_x_avg_neig       | average distance from neighbours in the variable space                                             |
| dist_f_avg_neig       | average distance from neighbours in the objective space                                              |
| diff_f_avg_neig       | average difference per objective with neighbours                                                  |
| diff_f_dist_x_avg_neig| ratio of the average diff per objective with neighbours and the dist. in the variable space (diff_f_avg_neig / dist_x_avg_neig) |
| hv_avg_neig           | average (single) solution’s hypervolume                                                           |
| hvd_avg_neig          | average (single) solution’s hypervolume difference with neighbours                                 |
| nhv_avg_neig          | average hypervolume from the whole neighbourhood                                                    |
| **evolvability**      |                                                                                                 |
| sup_cor_neig          | neighbour’s correlation of the proportion of dominating neighbours                                 |
| inf_cor_neig          | neighbour’s correlation of the proportion of dominated neighbours                                   |
| inc_cor_neig          | neighbour’s correlation of the proportion of incommensurable neighbours                             |
| lnd_cor_neig          | neighbour’s correlation of the proportion of locally non-dominated neighbours                      |
| dist_x_cor_neig       | neighbour’s correlation of the average distance from neighbours in the variable space              |
| dist_f_cor_neig       | neighbour’s correlation of the average distance from neighbours in the objective space              |
| diff_f_cor_neig       | neighbour’s correlation of the average difference per objective from neighbours                    |
| diff_f_dist_x_cor_neig| neighbour’s correlation of the ratio of the avg. diff. per obj from neighbours and the dist. in the variable space |
| hv_cor_neig           | neighbour’s correlation of the average (single) solution’s hypervolume                             |
| hvd_cor_neig          | neighbour’s correlation of the average (single) solution’s hypervolume difference with neighbours   |
| nhv_cor_neig          | neighbour’s correlation of the average hypervolume from the whole neighbourhood                   |
| **ruggedness**        |                                                                                                 |
| respectively, the ratio of both (dist_f_dist_x), as well as the average differences from each solution to its neighbours per objective (diff_f), together with the ratio over their distance in the variable space (diff_f_dist_x). At last, we compute the hypervolume covered by each solution (hv), the average difference from the hypervolumes of its neighbours (hvd), and the hypervolume covered by all its neighbours (nhv). For each measure described above, we simply compute the average value over the whole sample to construct the corresponding feature quantifying the landscape evolvability.  

4.1.4 Features Characterising the Landscape Ruggedness. The correlation among the information and quality from neighbouring solutions characterises the ruggedness of the landscape [25, 33]: the larger the correlation, the smoother the landscape. Before being averaged to compute the features for evolvability, every measure presented in the previous paragraph is assigned to each solution from the sample. Based on this information, and for each of the thirteen measures, we compute the (Spearman) correlation coefficient over each pair of neighbours in the sample. We end up with thirteen additional features characterising the ruggedness of continuous multi-objective landscapes. |
Let us recall that all 49 landscape features considered in our analysis are summarised in Tab. 1, below the 5 problem-dependent features. For more details on landscape features, and to better understand their motivations, the reader is referred to [21]. They are all computed over the same fixed-size sampling of size \( n \). We thus stress that the overall cost of landscape features in terms of calls to the evaluation function is exactly \( n \). We analyse the correlation of these features with the performance of MOEAs below.

### 4.2 Features vs. Search Performance

In the following sections, we start by considering a sampling of size \( n = 200 \cdot d \) solutions, where \( d \) is the number of variables for the problem under consideration. Fig. 1 reports the Spearman correlation coefficient of each feature with the expected performance of the four considered MOEAs. Due to space restriction, we focus on a large search budget, where each MOEA stops after 50,000 calls to the evaluation function. We recall that search performance is measured in terms of normalised hypervolume, so that a higher value means a better performance. As such, a positive correlation means a favourable effect of the feature on search performance, whereas a negative correlation means the opposite.

We first remark that the correlations are similar for the four algorithms, although MOEA/D-DE-DRA seems to slightly stand out. The features most correlated with favourable algorithm performance are dist_f_avg and dist_f_max. They measure the average and maximum distance in the objective space among the sampled solutions. In a separate analysis, we noticed that those features are highly correlated with the \( k \) parameter of the interpolation function. Indeed, low \( k \) values tend to produce landscapes concentrated around the average objective values of the seeds with sharp pits and peaks around the seeds [34]. This creates a large cluster around the expected values in the objective space. This increases the difficulty for MOEAs to reach the PF [34] and, as reflected in Fig. 1, MOEA/D-DE-DRA seems less affected by this effect.

Surprisingly, our coarse-grained analysis reveals that features describing the landscape multimodality (e.g., slo_n and plo_n) are positively correlated with search performance; that is, the hypervolume tends to be better when there are more local optima. This seems to corroborate recent results from continuous multi-objective optimisation [12]. We believe that a more fine-grained analysis on this specific class of features would be beneficial to better understand the impact of multimodality on MOEAs.

### 4.3 Importance of Features for Search Performance

We now investigate the combined effect of the considered features on search performance. For this purpose, we train a random forest regression model [3] that predicts the expected (relative) hypervolume based on input features, separately for each algorithm. We perform 50 independent trainings on the benchmark dataset using the randomForest R package [20] with default parameters, and we report average values below. The coefficients of determination (R²) of the models on training data are 0.88, 0.89, 0.86, and 0.88 respectively, for NSGA-II, GDE3, MOEA/D-DE-DRA, and DECMO2++. This means that more than 85% of the variance in the hypervolume of each algorithm between MO-ICOPs is explained by the features. In addition to prediction accuracy, random forest models yield the relative importance of each feature for prediction. A measure of importance is the so-called mean decrease of prediction accuracy after each split on a given predictor [3]: the larger the decrease, the more important the predictor. The (average) relative importance of features is reported in Fig. 2. For readability, only the 12 most important features are depicted for each algorithm, sorted in decreasing order of importance, from top to bottom. Although the subset of important features is similar for all MOEAs, their relative ranking is slightly different. Two notable features that consistently appear on top of the ranking for all algorithms are the average solution’s hypervolume (hv_avg_neig) and the average neighbourhood’s hypervolume (nhv_avg_neig), both characterising the landscape evolvability. The fact that algorithm performance is measured in terms of hypervolume partly explains the importance of these features, and the hypervolume measured over the entire sample (hv) also appears to be important, although to a lesser extent. As already pointed out in the correlation analysis reported above, the distance between sampled solutions in the objective space (dist_f_avg and dist_f_max) is also

![Figure 1: Correlation between the features and the performance of algorithms under a search budget of 50,000.](image-url)
found to be important, together with different distance-based features characterising evolvability (notably dist_f_avg_neig and dist_f_dist_x_avg_neig). Finally, the problem-dependent features related to the seeds (seed_n, nd_seed_n, dom_seed_n) seem to significantly impact the performance of all algorithms, and particularly MOEA/D-DE-DRA. Surprisingly, the problem dimension (d) is found to be less important than the features mentioned above. It is also worth noticing that there is no feature related to multimodality, nor to ruggedness, other than at the 11th and 12th position for NSGA-II. We will see, however, that these features are important to discriminate between different algorithms.

5 FEATURE-BASED ALGORITHM SELECTION

We conclude our study by analysing the selective power of the proposed landscape features. In contrast with the previous section, we train a random forest classification model to predict which of NSGA-II, GDE3, MOEA/D-DE-DRA, or DECMO2++ performs better, on average, for a given problem.

5.1 Importance of Features for MOEA Selection

We now focus on the importance of features for discriminating between the four considered MOEAs. The classification errors observed on the training data are 16.46%, 14.92%, and 22.78%, respectively, for search budgets of 10,000, 20,000 and 50,000 evaluations. This means that the classification model accurately selects the right algorithm in more than 75% of the cases. Features’ importance for algorithm selection is depicted in Fig. 3. Whatever the search budget, the maximum distance between sampled solutions in the objective space (dist_f_max) appears to be the most important feature to discriminate between different algorithms. For smaller search budgets (10,000 and 20,000), the average distance from a solution’s neighbourhood (dist_f_avg_neig) and the problem-dependent feature that tunes the power of the interpolation function (k) rank next, followed by the average distance among sampled solutions in the objective space (dist_f_avg) and several hypervolume-based evolvability features. In contrast, for a larger budget (50,000), the first problem-dependent feature (seed_n) is ranked 3, and a number of distance- and hypervolume-based features characterising the evolvability and the ruggedness appear on top of the ranking.

To complement our study, we also construct another classifier based on a simple decision tree [4, 31], as illustrated in Fig. 4, for a search budget of 50,000 evaluations. Even with such a simple decision tree of depth four, the proposed features are able to distinguish between the different algorithms with an error of 22.58%. The root of the decision tree corresponds once again to dist_f_max. While MOEA/D-DE-DRA is the best algorithm overall for this budget, the classifier recommends selecting NSGA-II instead when sampled solutions are more scattered in the objective space. Below the root of the tree appear landscape features related to multimodality (slo_n), evolvability (dist_f_dist_x_avg_neig, nhv_avg_neig, hvd_avg_neig) and ruggedness (hvd_corr_neig, sup_cor_neig, inc_cor_neig, nhv_cor_neig).

5.2 Prediction Accuracy

We finally investigate the ability of a feature-based model to accurately select a suitable algorithm for unseen problems. We measure
the accuracy of random forest classification models based on different subsets of features and different budgets allocated to their computation (i.e. the sampling size). In addition to the sample of $200 \cdot d$ solutions considered above, we investigate tighter values of 500 and 1000. For each setting, results are from 50 independent executions of repeated random sub-sampling cross-validation [9] with a 80/20% split. Three error values are reported in Fig. 5: the error rate in predicting the algorithm with the best performance on average (top), the error rate in predicting an algorithm that is not statistically outperformed by any other, according to a Mann-Whitney test at a significance level of $0.05$ with Bonferroni correction (middle), and the relative hypervolume deviation from the virtual best, an oracle that selects the best algorithm for any instance and budget (bottom).

Whatever the budget, a feature-based classifier is able to predict the best algorithm on average in more than 75% of the cases, and for one of the statistical best algorithms, in more than 85% of the cases. In terms of hypervolume, it is within 0.5% of the virtual best. This is significantly more accurate than a dummy (feature-less) classifier that always predicts the most-frequent best algorithm (NSGA-II for a budget of 10 000, and MOEA/D-DE-DRA for a budget of 20 000 and of 50 000), or than a classifier solely based on the problem dimension $d$, the only known problem-dependent feature in a black-box scenario. Interestingly, we do not observe any significant difference between a model based on landscape features only and a model based on (unknown) problem-dependent features. This means that the information extracted from a limited sampling of solutions allows a classification model to recommend...
which algorithm to run on a blind instance with the same level of accuracy than it would do with deep problem knowledge, related to the random seeds and the interpolation function in the case of MO-ICOPs.

When analysing the three sample sizes considered for computing the landscape features, we observe that using fewer solutions does not significantly impact the prediction accuracy. This is of particular interest when confronting the budget allocated to the calculation of features with the budget allocated to the search process. Indeed, using a very limited sampling of 500 solutions, that is 1 to 5% of the search budget depending on the stopping condition, is enough to accurately discriminate between the different MOEAs. In comparison to a classifier that does not use any feature, the feature-based algorithm selection approach always reduces the prediction error by a factor greater than 2, and reduces the hypervolume relative deviation by a factor of 10.

6 CONCLUSIONS

In this paper, we proposed a set of landscape features for continuous multi-objective optimisation. Those features were evaluated on a benchmark set of 1 200 multi-objective interpolated continuous optimisation problems. By means of correlation analysis and algorithm performance prediction, our analysis highlights that the features have a similar predictive power for the performance of all four tested algorithms. In addition, when combining them with a classification model for algorithm selection, the landscape features exhibit strong predictive power with an accuracy of over 85%, which is similar to the predictive power of problem-dependent features. This demonstrates the salience of the proposed features to characterising the landscape of multi-objective continuous optimisation problems, and to making viable recommendations for selecting a well-suited algorithm when faced with unseen problems.

In terms of benchmarks, we believe that MO-ICOPs provide a useful platform for generating diverse multi-objective continuous optimisation problems. It would be interesting to investigate problems with more than two objectives for assessing the scalability of landscape features. The framework proposed here also offers scope to compare our understanding of complexity for both combinatorial and continuous MOOPs. Finally, additional features could be considered and combined as well, including single- and multi-objective continuous landscape features [16, 17, 23, 24].
