Using Local Correlation Between Objectives to Detect Problem Modality

Tea Tušar[1,2](✉) and Jordan N. Cork[1,2](✉)

[1] Department of Intelligent Systems, Jožef Stefan Institute, Ljubljana, Slovenia
{tea.tusar,jordan.cork}@ijs.si

[2] Jožef Stefan Postgraduate School, Ljubljana, Slovenia

Abstract. Understanding the various characteristics of multiobjective optimization problems (MOPs) is crucial for designing and configuring optimization algorithms to efficiently solve them. This paper introduces a method that uses the estimation of local correlation between objectives to transform MOP landscapes into single-objective problem (SOP) landscapes. With this transformation, we make it possible to apply SOP landscape features to MOPs, thereby extracting valuable information about problem properties, such as modality. Our approach integrates both sample-based and search-based features, which are assessed for their ability to distinguish between unimodal, moderately multimodal, and highly multimodal MOPs. The proposed method is validated through a two-phase experimental setup. In the first phase, we select features that can reliably identify problem modality under ideal conditions with abundant data. The second phase evaluates their performance in more realistic scenarios with smaller samples and higher problem dimensions. The results show that features computed on the local correlation landscape achieve comparable or better performance than existing MOP features. These findings demonstrate the capability of SOP features to generalize to MOPs, showcasing their potential for characterizing MOP landscapes and inspiring future research on extending this approach to uncover additional problem properties.

Keywords: Multiobjective optimization · Correlation between objectives · Landscape features · Problem modality

1 Introduction

The efficiency of an optimization algorithm highly depends on the properties of the problem it is employed to solve. Being able to describe an optimization problem in terms of its characteristics, such as modality and separability, is therefore valuable as it enables one to choose and/or configure an algorithm to efficiently solve it. This work is concerned with continuous *black-box* multiobjective optimization problems, for which the objective function definitions are unknown to the optimizer. Because of this, most black-box problem properties are hard to detect. One way to assess them is by sampling the problem and using these solutions to

P. García-Sánchez et al. (Eds.): EvoApplications 2025, LNCS 15612, pp. 527–542, 2025.
https://doi.org/10.1007/978-3-031-90062-4_33

compute *problem landscape features*, low-level numerical attributes that can be used to successfully predict certain high-level problem properties [22].

However, most research on landscape features is focused on Single-objective Optimization Problems (SOPs; see [11] for a collection of many works proposing such features), while Multiobjective Optimization Problems (MOPs) have received much less attention. A notable exception is a fairly recent paper proposing features for continuous (unconstrained) MOP landscapes [17]. Because of this gap, it would be particularly beneficial to be able to apply the many SOP features to MOP landscapes, thus acquiring additional features to further characterize MOPs.

There are several ways in which the landscape of a continuous MOP can be reduced to a single function and, therefore, viewed similarly to that of a continuous SOP—imagined as a terrain with peaks, basins, valleys and plateaus. Examples include the global dominance rank ratio [7], local dominance [6], optimal trade-offs [30], gradient length [4,10], local correlation [3] and Pareto [15] landscapes. We base our work on the recent local correlation landscapes due to their close tie to problem modality and the interpretability of the correlation coefficient values.

The main idea of this paper is thus to use a sample of solutions to construct an approximate local correlation landscape of an MOP and then apply 'single-objective' Exploratory Landscape Analysis (ELA) [22] to compute its features. As the local correlation landscape is highly related to problem modality, we devise two experiments, testing whether the resulting features are able to express this important problem property and compare their performance to that of existing MOP features from [17].

In the following, Sect. 2 presents some basic concepts, the local correlation landscapes and problem landscape features, while Sect. 3 explains how they are used to extract problem modality features. Next, Sect. 4 details the experimental evaluation of our approach and Sect. 5 concludes the paper with a summary and ideas for future work.

2 Background

2.1 Multiobjective Optimization Problems

We are interested in continuous multiobjective *minimization* problems that can be formally defined as:

$$\min_{x \in \mathbb{R}^d} F(x) = (f_1(x), f_2(x), \ldots, f_m(x)),$$

where $\mathbb{R}^d$ is the *search space*, d is the problem dimension, and f_i, $i \in \{1, \ldots, m\}$, are the m objective functions.

Solution $x \in \mathbb{R}^d$ *dominates* solution $y \in \mathbb{R}^d$, iff $f_i(x) \leq f_i(y)$ for all $i \in \{1, \ldots, m\}$ and at least one of these inequalities is strict. Solutions which are not dominated by any other solution in the search space are *Pareto optimal*. All Pareto optimal solutions constitute the *Pareto set*. Its image in the *objective space* $\mathbb{R}^m$ is called the *Pareto front*.

A solution $x \in \mathbb{R}^d$ is *locally optimal* if it is not dominated by any other solution from its neighborhood N, $x \in N \subset \mathbb{R}^d$. If any locally optimal solution is also Pareto optimal, the problem in *unimodal*. Otherwise, it is *multimodal*.

In the remainder of the paper, we will be dealing with bi-objective problems, that is, $m = 2$ for all problems.

2.2 Local Correlation Landscapes

The concept that differentiates MOPs from SOPs is not the mere presence of multiple objectives, but the fact that they are typically *in conflict*, resulting in MOPs having multiple Pareto optimal trade-off solutions. In contrast, if the objectives would be in perfect *harmony*, i.e., completely equal, the MOP would be equivalent to the corresponding SOP. Therefore, when dealing with MOPs, we usually assume that they have conflicting objectives.

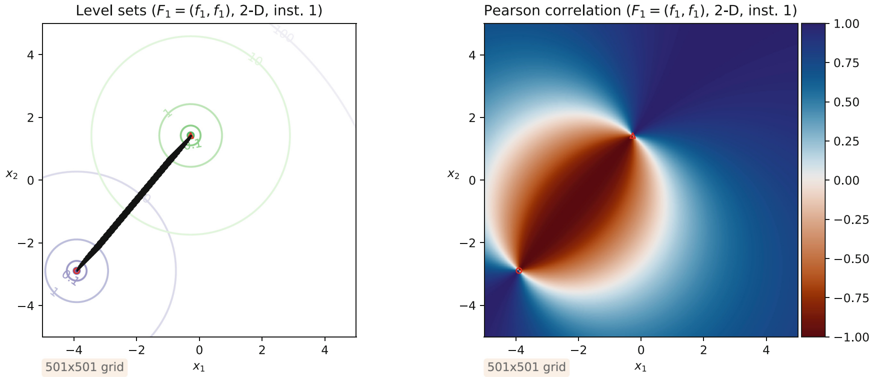
However, the conflict between two objectives exists primarily on the locally optimal sets and their 'vicinity', not the entire search space. In fact, it is a local problem property, not a global one, which is often ignored or disregarded.

To explain its local nature, we first need to formalize the relationship between two objectives. We can do this by considering their *correlation*. The correlation between two objectives can be estimated by the Pearson correlation coefficient [29], which measures the linear correlation between the objectives of a sample of solutions and takes a value between -1 (perfect linear anti-correlation that corresponds to conflicted objectives) and 1 (perfect linear correlation that corresponds to harmonious objectives). A zero value implies there is no linear dependency between the objectives, i.e., the objectives are neither conflicted nor in harmony.

Consider the simple example of the two-dimensional double sphere problem presented in Fig. 1. This is a bi-objective problem defined on the search space $[-5, 5]^2$, where each objective is a sphere function with the optimum located at a different point in the search space. The Pareto set of this function is the line segment connecting the two single-objective optima (shown in black[1] in Fig. 1a). The Pearson correlation coefficient visualized in Fig. 1b is computed for each grid point from a set of 100 solutions in its close proximity (see [3] for more details).

We can see that the Pearson correlation coefficient between the two objectives depends on the position in the search space. The correlation values along the Pareto set equal -1. This is to be expected as on the Pareto set, one cannot improve in one objective without deteriorating in the other. With increasing distance from the Pareto set in a direction perpendicular to it, the correlation coefficient increases, eventually becoming positive. On the parts of the line with the two single-objective optima that go beyond the Pareto set, the correlation coefficient takes on the value of 1, which is again understandable since at that location, a move in the direction toward the Pareto set results in simultaneous improvement in both objectives.

[1] Note that the black region in the plot is thicker than a line because of the discretization of the search space into a 501×501 grid for visualization purposes.



(a) Level sets for the two objectives in purple and green and the Pareto set approximation in black.

(b) Values of the Pearson correlation coefficient ranging from -1 in dark red to 1 in dark blue.

Fig. 1. Two grid-based visualizations of the search space of the first instance of the 2-D double sphere problem F_1 from the `bbob-biobj` suite [4] of the COCO platform [8].

To summarize, even for a simple unimodal problem such as the one from Fig. 1, the correlation between objectives is not constant, but depends on the position in the search space. On the Pareto set and in certain regions that are close to it, the correlation is negative, in others, it is zero or positive.

The relationship between objectives becomes even more complex when they are less regular or multimodal—see the examples from Fig. 2 in Sect. 4. There, we can see that some unimodal problems have anti-correlated objectives not only close to the Pareto set, but also far away from it. Additionally, visualizations of local correlations on multimodal problems demonstrate that many distinct anti-correlated regions can be located throughout the search space, surrounded by regions with correlated objectives.

The correlation between objectives is closely connected to problem modality and to the bi-objective gradient [10] as it equals -1 on any locally optimal set of solutions, not just the Pareto set. This is why the plots of multimodal problems in Fig. 2 contain many distinct regions with a negative correlation—one per locally optimal set of solutions.

Although these examples demonstrate that the concept of a 'global correlation between objectives' is effectively meaningless, the relationships between the objectives, as well as their mutual correlations, are almost always discussed solely on the global scale. Even a recently published book chapter [5] that provides an overview of the use of correlations among objectives in multiobjective optimization, explores several ways of estimating correlation in addition to the Pearson correlation coefficient and reviews the use of correlations for reducing redundant objectives, does not address their local nature. Similarly holds for the MOP feature `f_cor` from [17], which equals the (Spearman) correlation among objective values and is measured on the entire sample, i.e., is treated globally.

The local correlation landscape, therefore, provides an insightful view of the problem that is worth further exploration. However, this is a limited view as it does not contain enough information by itself to infer whether a locally optimal solution is also Pareto optimal. This is why any features computed solely from the local correlation landscape can meaningfully characterize only certain problem features, like (but not necessarily limited to) modality.

2.3 Problem Landscape Features

Most problem landscape features require only a set of solutions, called a *sample*, to be computed. Typically, the sample is generated with a procedure that tries to evenly cover the search space, such as Latin Hypercube Sampling (LHS) [21], and is evaluated beforehand. We will call such features *sample-based features*. However, there are also other problem features that require additional solution evaluations to be computed. For example, they can be based on a random walk [20], a hill climber run [1] or basin hopping iterations [2], to name a few. We will refer to these as *search-based features*. In real-world optimization scenarios, especially those with time-consuming evaluations, the latter might not always be retrievable. In this work, we use both sample- and search-based features, but only those search-based ones for which we can limit the number of additional evaluations.

The set of considered SOP features thus includes a total of 117 features that can be categorized into the following groups: dispersion features [19], classical ELA features (convexity, y-distribution, levelset, and meta model features) [22], fitness distance correlation features [9,24], cell mapping features (angle, convexity and gradient homogeneity features) [12], information content features [25], gradient features [20], nearest better clustering features [13], length-scale features [23], linear model features [14], and principal component features [14]. Of these, only the ELA convexity features, the gradient features and the length-scale features (with a total of 18) are search-based, the rest (99) are sample-based. All SOP features were computed with the `pflacco` Python library [27,28].

The set of MOP features used in the comparison comprises the 49 features from [17], which include global landscape features (among them, the global correlation between objectives), multimodality features, evolvability features and ruggedness features. All features are sample-based and were computed with the freely available `features.R` R script [16,17].

3 Detecting Problem Modality

The basic idea of this paper is to test whether features computed on the local correlation landscape can be used to detect problem modality. This is essentially done in three steps:

1. Approximate the local correlation landscape of the problem.
2. Compute SOP features of this landscape.
3. Measure the feature success in detecting problem modality.

Step 1: Approximating the Local Correlation Landscape. The execution of this step depends on the type of features—whether they are sample- or search-based, since search-based features guide the choice of solutions in the sample.

For sample-based features, the sample of solutions is retrieved independently from the features. This is done using LHS. Since we are interested in the *local* correlation of objectives, we need to define the neighborhood of solutions. For each solution in the sample, the neighborhood is comprised of n closest solutions to it, in terms of the Euclidean distance. This always includes the solution itself. The local correlation between objectives at each solution is then estimated by computing the Pearson correlation coefficient using the objective values for all solutions in its neighborhood.

Because search-based features use some inherent procedure to select the solutions to be evaluated, we cannot build the entire local correlation landscape upfront. Therefore, we first take a small ratio of the entire sample size s to produce an initial sample of solutions using LHS. Then, we construct the initial local correlation landscape using the same neighborhood definition as for sample-based features. Next, we let the search-based feature guide the choice of the subsequent solutions. For each, we find its current n closest neighbors and use them to approximate its local correlation value with the Pearson correlation coefficient. Note that the estimation of the local correlation for search-based features is less accurate at the beginning (when only a few solutions are available) than at the end.

Step 2: Computing Landscape Features. This step is straightforward—it requires computing the SOP feature values using the local correlation landscape instead of an objective landscape.

Step 3: Measuring Feature Success. Finally, feature success is measured by determining whether the feature can successfully differentiate between three groups of problems: unimodal, moderately multimodal and highly multimodal ones. We use clustering for this, because we are interested in the prediction capabilities of the feature. First, feature values are clustered into three clusters by k-means clustering with a fixed $k = 3$ [18,26]. Then, we count the errors—number of problems that have not been clustered correctly[2]. The lower the error, the better the feature in detecting problem modality.

4 Experiments and Results

In this section we first explain the various problems used in the experiments. Then we present the two experiments and their results.

[2] This is not trivial to do because there is no fixed order in how k-means labels clusters and feature values can be increasing or decreasing with increasing problem modality. Therefore, we check all possible 2^3 orderings of the three clusters and use the one with the smallest error count.

Many of the features listed in Sect. 2.3 are not useful for detecting problem modality, which could diminish the predictive capability of the entire set of features. To avoid this, we split the study into two parts. In the first experiment, we identify individual features that are able to differentiate well between unimodal, moderately multimodal and highly multimodal problems when given a lot of data at their disposal. Then, in the second experiment, we use only these features to more comprehensively test their capabilities in a real-world-like scenario with less available data. Before detailing the two experiments, we present the problems used in both of them.

4.1 Problems

To test our idea, we need a selection of problems with diverse modality. While we first planned to use only problems from the `bbob-biobj` suite [4] of the COCO platform [8], they do not cover the modality range well enough, as they are either unimodal or highly multimodal. To fill this gap, we construct the moderately multimodal problems ourselves.

The set of moderately multimodal problems are Python implementations of Wessing's Multiple Peaks Model problems [31], here labeled multi-peak problems. For each objective, a multiple peak function is generated by taking the minimum value of a set of individual peak functions. Each individual peak function consists of a center point and a positive definite Hessian matrix. The separate problems within the set were generated by randomly configuring these center and matrix settings. The number of peaks per objective, however, were set manually, to provide different degrees of modality within the moderate range. The degree of modality is determined by the combinations of peaks between objectives, which each provide a basin of attraction.

Table 1 presents the 15 problems selected for this study. P_1–P_5 are unimodal `bbob-biobj` problems, P_6–P_{10} are moderately mutimodal multi-peak problems and P_{11}–P_{15} are highly multimodal `bbob-biobj` problems. While all problems can be instantiated in any dimension, $d \in \{2, 3, 5, 10\}$ is used in this work. The local correlation landscapes for all 15 2-D problems are shown in Fig. 2.

We can see that the multi-peak problems indeed represent the middle ground between the unimodal and highly multimodal `bbob-biobj` problems. We can also hypothesize that among all problems, P_{14} might be the hardest to categorize correctly, as its local optima are located in a relatively small region of the entire search space, which can be easily overlooked, especially with sparse sampling.

4.2 First Experiment

Experimental Setup. To find features with a potential for detecting problem modality, we simplify the task as much as possible. We use only 2-D problems and provide a large budget of $s = 10\,000$ solutions to compute the features. For sample-based features, all solutions are placed on the 100×100 grid, while for search-based features, the initial grid contains 32×32 solutions (which roughly equals 10% of the budget s) and the rest is made available to the method to

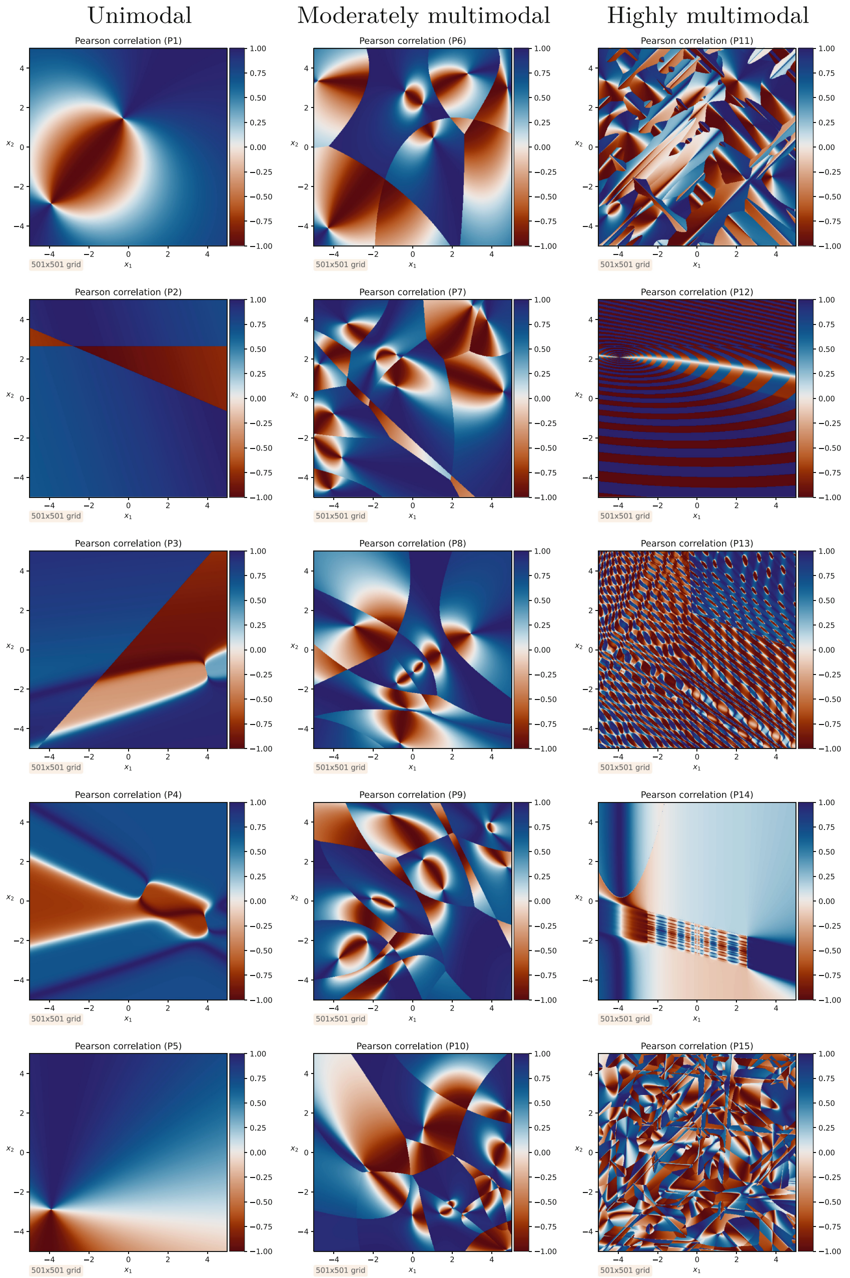


Fig. 2. Local correlation landscapes calculated with the Pearson correlation coefficient for 2-D unimodal problems P_1–P_5 (left column), moderately multimodal problems P_6–P_{10} (middle column) and highly multimodal problems P_{11}–P_{15} (right column). Red areas denote regions with negatively correlated objectives, while blue areas designate regions with positively correlated objectives.

Table 1. The 15 problems employed in this study. We always use only the first instance of a `bbob-biobj` problem. For multi-peak problems, the two numbers in brackets determine the number of peaks in the first and second objective.

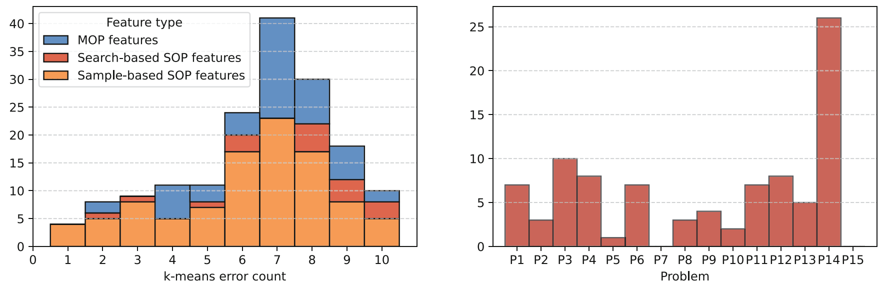
	Unimodal problems
P_1	`bbob-biobj` problem $F_1 = (f_1, f_1)$
P_2	`bbob-biobj` problem $F_{14} = (f_2, f_{13})$
P_3	`bbob-biobj` problem $F_{36} = (f_{13}, f_{14})$
P_4	`bbob-biobj` problem $F_{41} = (f_{14}, f_{14})$
P_5	`bbob-biobj` problem $F_{58} = (f_1, f_5)$
	Moderately multimodal problems
P_6	multi-peak problem with (2, 5) peaks
P_7	multi-peak problem with (2, 10) peaks
P_8	multi-peak problem with (4, 4) peaks
P_9	multi-peak problem with (4, 8) peaks
P_{10}	multi-peak problem with (6, 6) peaks
	Highly multimodal problems
P_{11}	`bbob-biobj` problem $F_{10} = (f_1, f_{21})$
P_{12}	`bbob-biobj` problem $F_{17} = (f_2, f_{17})$
P_{13}	`bbob-biobj` problem $F_{24} = (f_6, f_{15})$
P_{14}	`bbob-biobj` problem $F_{33} = (f_8, f_{20})$
P_{15}	`bbob-biobj` problem $F_{55} = (f_{21}, f_{21})$

With f_i we denote a single-objective `bbob` function as follows:

f_1	sphere	f_8	original Rosenbrock	f_{17}	Schaffers F7
f_2	ellipsoidal	f_{13}	sharp ridge	f_{20}	Schwefel
f_5	linear slope	f_{14}	different powers	f_{21}	Gallaghers Gaussian 101-medium peaks
f_6	attractive sector	f_{15}	Rastrigin		

sample the space according to its principle. In both cases, the neighborhood size n equals 9, which corresponds to the Moore neighborhood for the internal grid solutions.

Results and Discussion. In this experiment, we apply k-means clustering separately for each feature. The number of clustering errors committed on the 15 problems ranges from one to ten and is collected for all features in the histogram in Fig. 3a. The colors distinguish among features of the three different types (search- and sample-based SOP features, and MOP features). We can see that the distribution over error counts is roughly similar for all feature types with most features performing very badly (making six or more mistakes on 15 problems). Among the best features (making less than five mistakes) we have 22% of all sample-based



(a) Histogram of the clustering error count grouped by feature type.

(b) Number of clustering errors on each problem using only the best features.

Fig. 3. Results of the first experiment applying k-means clustering separately for each feature.

Table 2. The errors of the best SOP features computed on the local correlation landscape with 10 000 sampled solutions. There are 24 features with a clustering error lower than five. The only two search-based features are denoted by an asterisk (*), while the rest of them are sample-based.

Feature	Error	Feature	Error
`cm_angle.y_ratio_best2worst_mean`	1	`cm_angle.dist_ctr2worst_mean`	3
`cm_grad.mean`	1	`cm_angle.dist_ctr2worst_sd`	3
`nbc.nb_fitness.cor`	1	`cm_conv.convex.hard`	3
`nbc.nn_nb.mean_ratio`	1	`disp.diff_median_02`	3
`cm_angle.angle_mean`	2	`disp.ratio_median_02`	3
`cm_conv.concave.hard`	2	*`ela_conv.lin_dev_abs`	3
*`gradient.g_avg`	2	`ic.eps_s`	3
`ic.costs_runtime`	2	`disp.diff_median_05`	4
`ic.eps_ratio`	2	`disp.ratio_median_05`	4
`limo.length_mean`	2	`ela_level.mmce_qda_50`	4
`cm_angle.dist_ctr2best_mean`	3	`ic.eps_max`	4
`cm_angle.dist_ctr2best_sd`	3	`ic.h_max`	4

SOP features, 11% of all search-based SOP features and 14% of all MOP features. We set the threshold for 'good' features to four or fewer to discard features which clearly cannot distinguish among problems of different modality even when provided with plenty of data, but still keep enough to experiment with.

The complete list of the best 24 SOP features is given in Table 2. We can see that they come from various groups, with cell mapping, nearest better clustering, information content and dispersion features being the most well represented. These results are very positive as they show that we have a large number of SOP features that can be applied on local correlation landscapes to detect problem modality.

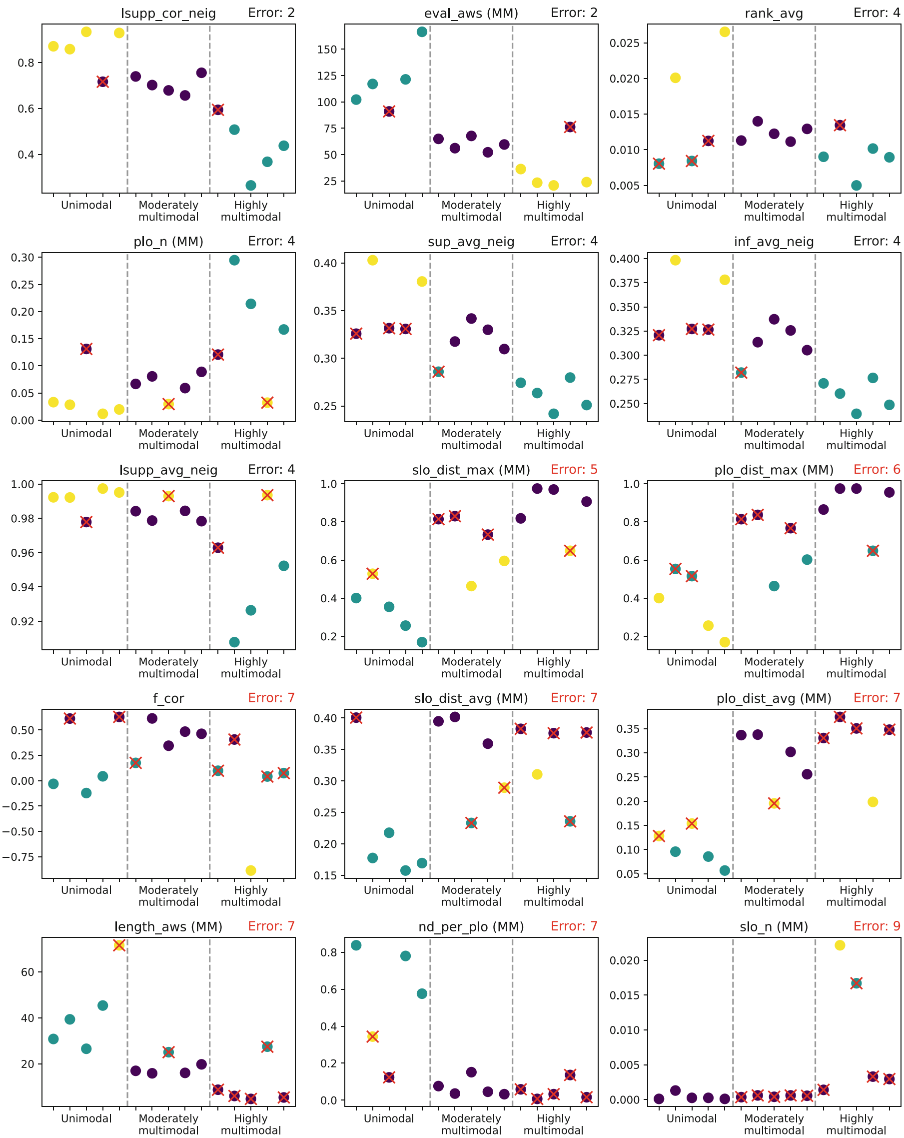


Fig. 4. Feature values (on the y-axis) for all 2-D problems (on the x-axis) computed on samples with 10 000 solutions for a selection of MOP features that is comprised by all seven features with an error smaller than five, all multimodal features (denoted by '(MM)' after their name) and the correlation feature `f_cor`. The features are sorted in ascending order of their error. The color of the dots represents the cluster determined by k-means and a red cross denotes every incorrectly categorized problem.

Next, the results for some chosen MOP features are presented in greater detail in Fig. 4. Every plot contains dots showing the feature value (y-axis) on each of the 15 problems (x-axis). Their colors denote the cluster assigned to that problem by k-means, while the red crosses represent wrongly categorized problems. This visualization comprises all seven MOP features with an error smaller than five, all nine multimodal (MM) features and the feature `f_cor` measuring correlation between objectives. The features are sorted in ascending error count. We see that only two of the nine multimodal features make less than five errors on 15 problems when detecting their modality and five other features outperform the rest of the multimodal ones. Also, we empirically show that the global correlation feature has very little meaning (see the first plot of the fourth row in Fig. 4). According to `f_cor`, most of the 15 problems have mildly correlated objectives (values between 0 and 0.5), with P_{13} being the only problem with highly anti-correlated objectives, which is incorrect.

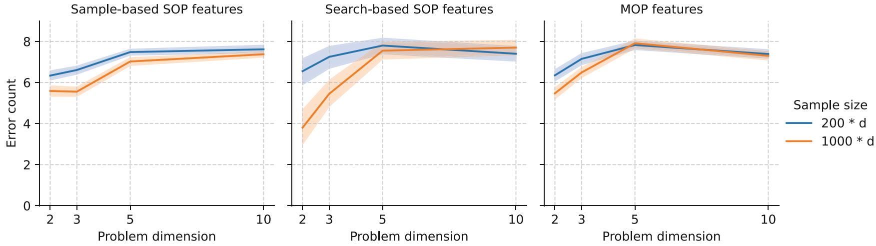
Finally, Fig. 3b shows which of the problems were most often wrongly clustered by the 31 best features. We see a clear outlier—problem P_{14} that stands out from other highly multimodal problems in our set because its local optima are concentrated in a relatively small part of the search space.

4.3 Second Experiment

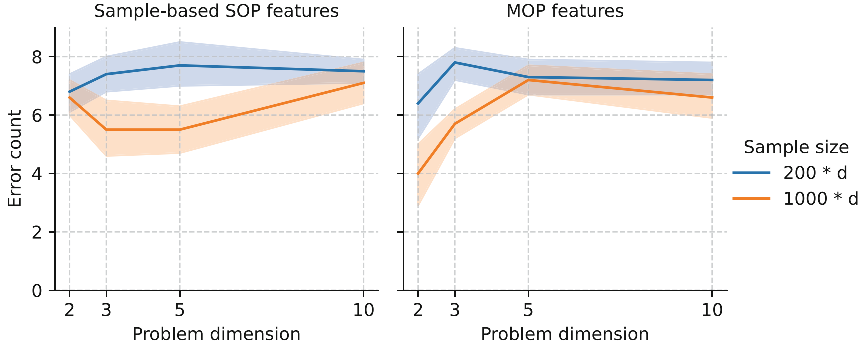
Experimental Setup. Only the 31 best features identified in the first experiment are included in the second part of this study. Here, we investigate how larger problem dimensions $d \in \{2, 3, 5, 10\}$ and smaller sample sizes $s \in \{200d, 1000d\}$ affect the capability of features to detect problem modality. In addition, to find a good neighborhood size n, we experiment with two settings, $n \in \{5, 10\}$. Similarly as before, for search-based SOP features, only 10% of the sample size s is created by LHS, while the rest is used to explore the search space according to the feature method. We repeat all the experiments five times, using different samples.

Results and Discussion. First, we discuss the results of using k-means clustering on separate features. Figure 5a shows how its error count depends on the type of feature, the problem dimension and sample size. The neighborhood size is not shown separately as it does not visibly affect the results. We can see that the sample size has a large effect on the feature capability to discern problem modality with the larger sample size ($1000d$) generally supporting better results than the smaller one ($200d$). The effect of problem dimension d is also visible—the error count typically (but not always) increases with higher dimensions. Both results are in line with expectations. Finally, the comparison among the three feature types shows that SOP features computed on the local correlation landscapes perform comparable to MOP ones. A visibly better performance is achieved by the two search-based SOP features only on 2- and 3-D problems with a large sample size.

This means that the excellent results achieved in the first experiment, where SOP features on local landscapes were outperforming MOP features, were not



(a) k-means clustering results achieved using a single feature are here aggregated over all features of the same type, multiple samples and neighborhood sizes.



(b) k-means clustering results achieved using all features of the same type are here aggregated over multiple samples and neighborhood sizes. Search-based SOP features are missing because their aggregation is not meaningful.

Fig. 5. Results of k-means clustering on (a) separate features and (b) all features of the same type. The plots show how clustering error count (y-axis) depends on the problem dimension d (x-axis) and sample size s (color). The line represents the mean, while the shaded region corresponds to the 95 % confidence interval.

replicated in the more difficult scenario with higher problem dimensions and less available data. Still, the approach achieved results that are generally not worse than those by MOP features, meaning that it has established its merit.

However, the predictive power of features can be combined. Therefore, we present in Fig. 5b k-means clustering results using all features of the same type. The search-based SOP features are excluded from this analysis, because the two features of this type come from two different methods, meaning that their resulting samples are different and cannot be meaningfully combined. Similarly as before, we see a fairly reliable effect of the sample size and problem dimension (with the notable exception of sample-based SOP features with $1000d$ samples on dimension 2 that perform worse than expected). Surprisingly, combining the features does generally not help to (considerably) improve their separate results.

Finally, an analysis of the errors per problem (results not pictured) does not result in any stark outliers as the one from Fig. 3b. Rather, all problems are similarly difficult (or easy) to categorize, with slightly higher errors achieved on unimodal problems P_2 and P_4. A rather surprising result for which we cannot yet provide an explanation.

5 Conclusions

This paper demonstrates that estimating local correlation between objectives can effectively transform a multiobjective problem landscape into a single-objective one. This transformation enables the application of SOP features, including both sample-based and search-based features, to MOPs, facilitating the extraction of valuable information about problem characteristics—in this case, problem modality. Furthermore, this research paves the way for exploring alternative transformations that could be applied to similarly capture other important problem properties.

An important limitation of this work is its focus on bi-objective problems, as correlation can only be computed between two objectives. For problems with three or more objectives, only pairwise correlation values can be obtained, making it impossible to calculate a direct multi-way correlation. In future work, we would like to explore potential approaches to overcome this limitation.

Acknowledgments. We acknowledge financial support from the Slovenian Research and Innovation Agency (research core funding No. P2-0209 and projects No. GC-0001 "Artificial Intelligence for Science" and N2-0254 "Constrained Multiobjective Optimization Based on Problem Landscape Analysis").

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Abell, T., Malitsky, Y., Tierney, K.: Features for exploiting black-box optimization problem structure. In: Nicosia, G., Pardalos, P. (eds.) LION 2013. LNCS, vol. 7997, pp. 30–36. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-44973-4_4
2. Adair, J., Ochoa, G., Malan, K.M.: Local optima networks for continuous fitness landscapes. In: Companion Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2019). pp. 1407–1414. Association for Computing Machinery, New York, NY, USA (2019). https://doi.org/10.1145/3319619.3326852
3. Allmendinger, R., Fonseca, C.M., Sayin, S., Wiecek, M.M., Stiglmayr, M.: Multiobjective optimization on a budget (Dagstuhl Seminar 23361). Dagstuhl Reports **13**(9), 1–68 (2023). https://doi.org/10.4230/DAGREP.13.9.1
4. Brockhoff, D., Auger, A., Hansen, N., Tušar, T.: Using well-understood single-objective functions in multiobjective black-box optimization test suites. Evol. Comput. **30**(2), 165–193 (2022). https://doi.org/10.1162/EVCO_A_00298
5. Chugh, T., Gaspar-Cunha, A., Deutz, A.H., Duro, J.A., Oara, D.C., Rahat, A.: Identifying correlations in understanding and solving many-objective optimisation problems, pp. 241–267. Springer International Publishing, Cham (2023). https://doi.org/10.1007/978-3-031-25263-1_9

6. Fieldsend, J.E., Chugh, T., Allmendinger, R., Miettinen, K.: A feature rich distance-based many-objective visualisable test problem generator. In: Proceedigs of the Genetic and Evolutionary Computation Conference (GECCO 2019). pp. 541–549. Association for Computing Machinery, New York, NY, USA (2019). https://doi.org/10.1145/3321707.3321727
7. Fonseca, C.M.: Multiobjective genetic algorithms with application to control engineering problems. Ph.D. thesis, University of Sheffield (1995)
8. Hansen, N., Auger, A., Ros, R., Mersmann, O., Tušar, T., Brockhoff, D.: COCO: a platform for comparing continuous optimizers in a black-box setting. Optim. Methods Softw. **36**(1), 114–144 (2021). https://doi.org/10.1080/10556788.2020.1808977
9. Jones, T., Forrest, S.: Fitness distance correlation as a measure of problem difficulty for genetic algorithms. In: Proceedings of the International Conference on Genetic Algorithms (ICGA 1995), pp. 184–192. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1995). https://doi.org/10.5555/645514.657929
10. Kerschke, P., Grimme, C.: An expedition to multimodal multi-objective optimization landscapes. In: Trautmann, H., et al. (eds.) EMO 2017. LNCS, vol. 10173, pp. 329–343. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-54157-0_23
11. Kerschke, P., Preuss, M.: Exploratory landscape analysis. In: Companion Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 209), pp. 1137–1155. Association for Computing Machinery, New York, NY, USA (2019). https://doi.org/10.1145/3319619.3323389
12. Kerschke, P., et al.: Cell mapping techniques for exploratory landscape analysis. In: Tantar, A.A., et al. (eds.) EVOLVE – A Bridge between Probability, Set Oriented Numerics, and Evolutionary Computation V. Advances in Intelligent Systems and Computing, vol. 288, pp. 115–131. Springer International Publishing, Cham (2014). https://doi.org/10.1007/978-3-319-07494-8_9
13. Kerschke, P., Preuss, M., Wessing, S., Trautmann, H.: Detecting funnel structures by means of exploratory landscape analysis. In: Proceedings of the Conference on Genetic and Evolutionary Computation (GECCO 2015), pp. 265–272. Association for Computing Machinery, New York, NY, USA (2015). https://doi.org/10.1145/2739480.2754642
14. Kerschke, P., Trautmann, H.: Comprehensive feature-based landscape analysis of continuous and constrained optimization problems using the R-package flacco, pp. 93–123. Springer International Publishing, Cham (2019). https://doi.org/10.1007/978-3-030-25147-5_7
15. Liang, Z., Cui, Z., Li, M.: Pareto landscape: visualising the landscape of multiobjective optimisation problems. In: Affenzeller, M., et al. (eds.) Parallel Problem Solving from Nature (PPSN XVIII), pp. 299–315. Springer Nature Switzerland, Cham (2024). https://doi.org/10.1007/978-3-031-70085-9_19
16. Liefooghe, A.: Landscape features for MO-ICOPs. https://gitlab.com/aliefooghe/landscape-features-mo-icops (2024). https://gitlab.com/aliefooghe/landscape-features-mo-icops, gitLab repository
17. Liefooghe, A., Verel, S., Lacroix, B., Zăvoianu, A.C., McCall, J.: Landscape features and automated algorithm selection for multi-objective interpolated continuous optimisation problems. In: Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2021), pp. 421–429. Association for Computing Machinery, New York, NY, USA (2021). https://doi.org/10.1145/3449639.3459353
18. Lloyd, S.P.: Least squares quantization in PCM. IEEE Trans. Inf. Theory **28**(2), 129–137 (1982). https://doi.org/10.1109/TIT.1982.1056489

19. Lunacek, M., Whitley, D.: The dispersion metric and the CMA evolution strategy. In: Proceedings of the Conference on Genetic and Evolutionary Computation (GECCO 2006), pp. 477–484. Association for Computing Machinery, New York, NY, USA (2006). https://doi.org/10.1145/1143997.1144085
20. Malan, K.M., Oberholzer, J.F., Engelbrecht, A.P.: Characterising constrained continuous optimisation problems. In: Proceedings of the Congress on Evolutionary Computation (CEC 2015), pp. 1351–1358. IEEE (2015). https://doi.org/10.1109/CEC.2015.7257045
21. McKay, M.D., Beckman, R.J., Conover, W.J.: A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. Technometrics **42**(1), 55–61 (1979)
22. Mersmann, O., Bischl, B., Trautmann, H., Preuss, M., Weihs, C., Rudolph, G.: Exploratory landscape analysis. In: Proceedings of the Conference on Genetic and Evolutionary Computation (GECCO 2011), pp. 829–836. Association for Computing Machinery, New York, NY, USA (2011). https://doi.org/10.1145/2001576.2001690
23. Morgan, R., Gallagher, M.: Analysing and characterising optimization problems using length scale. Soft. Comput. **21**(7), 1735–1752 (2015). https://doi.org/10.1007/s00500-015-1878-z
24. Müller, C.L., Sbalzarini, I.F.: Global characterization of the CEC 2005 fitness landscapes using fitness-distance analysis. In: Di Chio, C., et al. (eds.) Applications of Evolutionary Computation. Lecture Notes in Computer Science, vol. 6624, pp. 294–303. Springer, Berlin, Heidelberg (2011). https://doi.org/10.1007/978-3-642-20525-5_30
25. Muñoz, M.A., Kirley, M., Halgamuge, S.K.: Exploratory landscape analysis of continuous space optimization problems using information content. IEEE Trans. Evol. Comput. **19**(1), 74–87 (2015). https://doi.org/10.1109/TEVC.2014.2302006
26. Pedregosa, F., et al.: Scikit-learn: machine learning in Python. J. Mach. Learn. Res. **12**, 2825–2830 (2011). http://jmlr.org/papers/v12/pedregosa11a.html
27. Prager, R.P.: Pflacco: feature-based landscape analysis of continuous and constrained optimization problems (2024). https://github.com/Reiyan/pflacco. Accessed 15 Nov 2024
28. Prager, R.P., Trautmann, H.: Pflacco: feature-based landscape analysis of continuous and constrained optimization problems in Python. Evol. Comput., 1–25 (2023). https://doi.org/10.1162/evco_a_00341
29. Rodgers, J.L., Nicewander, W.A.: Thirteen ways to look at the correlation coefficient. Am. Stat. **42**(1), 59–66 (1988). https://doi.org/10.1080/00031305.1988.10475524
30. Schäpermeier, L., Grimme, C., Kerschke, P.: One PLOT to show them all: visualization of efficient sets in multi-objective landscapes. In: Bäck, T., et al. (eds.) PPSN 2020. LNCS, vol. 12270, pp. 154–167. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58115-2_11
31. Wessing, S.: Two-stage methods for multimodal optimization. Ph.D. thesis, Technische Universität Dortmund, Fakultät für Informatik, Dortmund, Germany (2015). http://dx.doi.org/10.17877/DE290R-7804