Investigating the Viability of Existing Exploratory Landscape Analysis Features for Mixed-Integer Problems

Raphael Patrick Prager
raphael.prager@uni-muenster.de
Data Science: Statistics and Optimization
University of Münster, Germany

Heike Trautmann
trautmann@wi.uni-muenster.de
Data Science: Statistics and Optimization,
University of Münster, Germany
University of Twente, The Netherlands

ABSTRACT

Exploratory landscape analysis has been at the forefront of characterizing single-objective continuous optimization problems. Other variants, which can be summarized under the term landscape analysis, have been used in the domain of combinatorial problems. However, none to little has been done in this research area for mixed-integer problems. In this work, we evaluate the current state of existing exploratory landscape analysis features and their applicability on a subset of mixed-integer problems.

CCS CONCEPTS

• **Theory of computation → Mixed discrete-continuous optimization**.

KEYWORDS

Mixed-Integer Optimization, Exploratory Landscape Analysis, Fitness Landscape

ACM Reference Format:
Raphael Patrick Prager and Heike Trautmann. 2023. Investigating the Viability of Existing Exploratory Landscape Analysis Features for Mixed-Integer Problems. In *Genetic and Evolutionary Computation Conference Companion (GECCO '23 Companion), July 15–19, 2023, Lisbon, Portugal.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3583133.3590757

1 INTRODUCTION

Exploratory landscape analysis (ELA) [4] has been the driver of many research advancements related to optimization in the past decade. By providing valuable insights into high-dimensional fitness landscapes or being used as fundamental components in automated algorithm selection studies, ELA facilitated areas such as algorithm design and optimization of real-world problems [1, 3]. However, ELA has been largely restricted to the continuous single-objective domain. This is not to be conflated with fitness landscape analysis in general which has been applied in some way, shape, or form in other optimization domains as well. These methods, however, have been used in either a pure combinatorial or continuous setting. Yet, these domains are not intrinsically disjunct, rather they can intersect in form of mixed-integer problems. In this preliminary study, we want to create a bridge in feature landscape analysis between these domains. To disseminate existing ELA features to different optimization domains, we consider a subset of mixed-integer problems which exhibit continuous and discrete decision variables, i.e., each variable possesses an ordinal nature.

2 EXPLORATORY LANDSCAPE ANALYSIS

For single-objective continuous optimization, ELA provides the means to characterize an optimization landscape in terms of a given set of properties via numerical surrogates. Any ELA feature requires a sample - a so-called initial design - of the decision space $\{x^{(1)}, ..., x^{(n)}\} \in X^{n \times d}$, where d refers to the dimensionality of the decision space and n governs the size of the initial design. To generate the respective objective values $f(x^{(1)})$, every individual observation $x^{(1)}$ has to be evaluated on a given objective function $f : X \rightarrow \mathbb{R}$. The final sample then consists of values of the decision and objective space in form of $(X, f(X))$.

Based on $(X, f(X))$, a multitude of different feature sets can be calculated. The feature sets used in this study have been selected based on past research [8, 9]. The chosen feature sets are given in the following and are computed with the Python package `pflacco`[1]:

- `ela_distr`: ELA distribution
- `ela_meta`: ELA meta model
- `fitness_distance`: Fitness distance correlation
- `nbc`: Nearest better clustering
- `pca`: Principal component analysis
- `ic`: Information content
- `disp`: Dispersion

3 EXPERIMENTAL METHODS

To investigate the viability of continuous ELA features within the mixed-integer setting, we make use of the mixed-integer variant of the Black-Box Optimization benchmark (BBOB) suite [10]. To avoid confusion, we will refer to the mixed-integer variant of BBOB as `mixint-BBOB` and to the continuous as `cont-BBOB` from here on forward. Similar to the single-objective continuous version, `mixint-BBOB` comprises the same 24 distinct optimization problems which are labeled as functions (FID). Each function can be subject to different linear transformations and rotations. Thereby, these functions offer different variations, so called instances (IID). The span of possible problem dimensions of `mixint-BBOB` functions is comparatively larger than the conventional BBOB suite. Here, functions can only be created for dimensions $d = \{5, 10, 20, 40, 80, 160\}$. Whether

[1] https://github.com/reiyan/pflacco

GECCO '23 Companion, July 15–19, 2023, Lisbon, Portugal

ACM ISBN 979-8-4007-0120-7/23/07.
https://doi.org/10.1145/3583133.3590757

a particular dimension is on a discrete or continuous scale is determined by the following (recurring) sequence $d' := (d_1, d_2, d_3, d_4, d_5)$, where $d_1 = \{0, 1\}, d_2 = \{0, 1, 2, 3\}, d_3 = \{0, 1, ..., 7\}, d_4 = \{0, 1, ..., 15\}$, and $d_5 \in \mathbb{R}$. If a problem has 10 dimensions, then the sequence d' will repeat itself twice, whereas only the 5th and 10th dimension are on a continuous scale. A 20 dimensional problem would repeat the sequence d' four times and so forth.

As is apparent, each dimension offers potentially different variables types but also ranges. In contrast, the function domain of `cont-BBOB` is box-constrained within the interval $[-5, 5]$ for all dimensions. To avoid an over importance of certain dimensions, we normalize the decision space as well as the objective space to $[0, 1]$ for both benchmark suites via min-max normalization. The lower and upper bounds of the decision space are given and therefore the normalization of the decision space is straightforward. For the objective space, we use the respective minima and maxima of a given initial design $(X, f(X))$ as is recommended in [6].

We calculate the set of ELA features for both benchmarks where we keep the selected problem instance comparable. Meaning, we consider the first five IIDs of all 24 FIDs in dimensions $d = \{5, 10, 20\}$ for *both* benchmark suites.

On each problem instance $p := (FID, d, IID)$, we create 10 independent initial designs based on Sobol sequences of size $50 \cdot d$ and compute the selected ELA features. Thereby, we hope to reduce the stochastic interference on our results any random sampling strategy might introduce.

In line with other works [8, 9], we conduct a classification study, where we try to predict the underlying FID based on its respective ELA values for each benchmark, `mixint-BBOB` and `cont-BBOB`, separately. For this endeavour, we consider SVMs, Random Forests, and Gradient Boosting Machines as an appropriate set of different machine learning models. Furthermore, we perform rudimentary hyperparameter tuning as well as extensive feature selection to identify a suitable subset of ELA features (out of 52) as this has been shown to improve the model accuracy [2, 8]. Each model is implemented in Python using the packages `scikit-learn` [5] and `mlxtend` [7].

For both benchmark suite classification scenarios, we use a 5-fold cross-validation strategy, where each fold consists of only a single IID, e.g., the first fold includes only the first instance of all 24 FIDs in all considered dimensions d. The best model in both cases is a Random Forest.

4 RESULTS

Before discussing the results of the classification scenario, we want to provide an overview of the ELA feature values in general and the similarities they share despite being computed on two different benchmark suites. The frequency in which certain ELA feature values occur is depicted in Figure 1. This is further divided by their source of origin, i.e., the benchmark suite in question. For most of the features, we can observe a clear overlap in terms of distributions. Most prominent are the features sets `ela_meta` and to some degree also `nbc` and `pca`. While these are most apparent, other singular features also exhibit a similar pattern. Noteworthy is that some features seem to share little in terms of raw values. An example for this is the feature `fitness_distance.distance_std`.

Table 1: Spearman correlation for each feature between the two benchmark suites `mixint-BBOB` and `cont-BBOB`.

Feature	Correlation
ela_distr.skewness	0.95
ela_distr.kurtosis	0.94
ela_distr.number_of_peaks	0.75
ela_meta.lin_simple.adj_r2	0.94
ela_meta.lin_simple.intercept	0.95
ela_meta.lin_simple.coef.min	0.89
ela_meta.lin_simple.coef.max	0.90
ela_meta.lin_simple.coef.max_by_min	0.73
ela_meta.lin_w_interact.adj_r2	0.96
ela_meta.quad_simple.adj_r2	0.95
ela_meta.quad_simple.cond	0.66
ela_meta.quad_w_interact.adj_r2	0.97
fitness_distance.fd_correlation	0.58
fitness_distance.fd_cov	0.74
fitness_distance.distance_mean	0.95
fitness_distance.distance_std	0.18
fitness_distance.fitness_mean	0.95
fitness_distance.fitness_std	0.95
nbc.nn_nb.sd_ratio	0.92
nbc.nn_nb.mean_ratio	0.97
nbc.nn_nb.cor	0.91
nbc.dist_ratio.coeff_var	0.96
nbc.nb_fitness.cor	0.91
pca.expl_var.cov_x	1.00
pca.expl_var.cor_x	1.00
pca.expl_var.cov_init	-0.36
pca.expl_var.cor_init	0.70
pca.expl_var_PC1.cov_x	0.89
pca.expl_var_PC1.cor_x	0.89
pca.expl_var_PC1.cov_init	0.84
pca.expl_var_PC1.cor_init	0.99
ic.h_max	0.88
ic.eps_s	0.95
ic_eps.max	0.88
ic.eps_ration	0.90
ic.m0	0.92
disp.ratio_mean_02	0.89
disp.ratio_mean_05	0.92
disp.ratio_mean_10	0.93
disp.ratio_mean_25	0.95
disp.ratio_median_02	0.81
disp.ratio_median_05	0.82
disp.ratio_median_10	0.83
disp.ratio_median_25	0.85
disp.diff_mean_02	0.82
disp.diff_mean_05	0.83
disp.diff_mean_10	0.85
disp.diff_mean_25	0.88
disp.diff_median_02	0.78
disp.diff_median_05	0.76
disp.diff_median_10	0.72
disp.diff_median_25	0.70

This, however, compares only the degree of which singular features' values overlap. We dissect the feature values further from a different perspective, namely by choosing to calculate the Spearman's rank correlation coefficient for each feature *between* the different benchmark suites. This is provided in Table 1. Again, the feature set `ela_meta` garners attention by displaying a large correlation of the feature values between the two benchmark suites. This is also the case for the feature sets `ic` and `nbc`.

The only exceptions to this phenomenon are the two features `fitness_distance.distance_std` and `pca.expl_var.cov_init`. The probable cause of this can be traced back to the discretized decision variables which only allow for a finite number of integer values. Meaning, it is more likely to share the exact same values for discretized decision variables between different samples of an initial design. This naturally reduces the standard deviation of distances between those samples.

In order to validate the suitability of continuous ELA features in the mixed-integer setting further, we depict the results of our

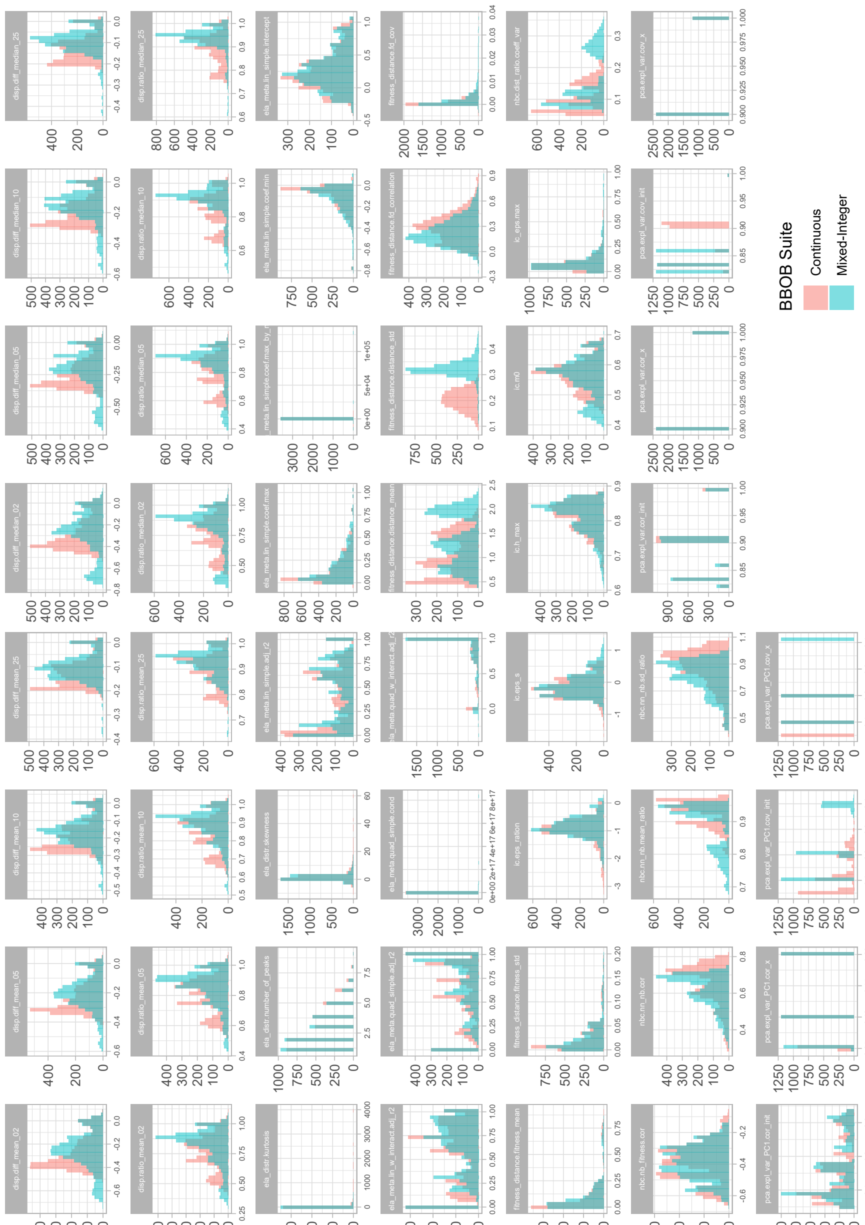

Figure 1: Histograms for each of the 52 ELA features.

classification undertaking in Figure 2. A cursory assessment reveals no clear disparity between the quality of the two models. Fundamentally, they exhibit the same pattern of classifying certain FIDs perfectly while struggling on the same subset of FIDs. Both models display even the same manner in which misclassifications occur. This is the case for FIDs 17 and 18 where they are often mistaken for one another. This hints to the fact that the chosen subset of ELA features is not descriptive enough to distinguish between the two.

5 CONCLUSION

In this work, we show that existing ELA features are intrinsically suited to be used for a subset of mixed-integer problems. We only adjusted for different scales of the search space but this would also be necessary for different scales in the continuous domain. We demonstrate that many ELA feature values, regardless of being calculated on the continuous BBOB suite or mixed-integer BBOB suite, exhibit a strong correlation and to some degree even identical value ranges. While this is certainly promising, this is neither conclusive

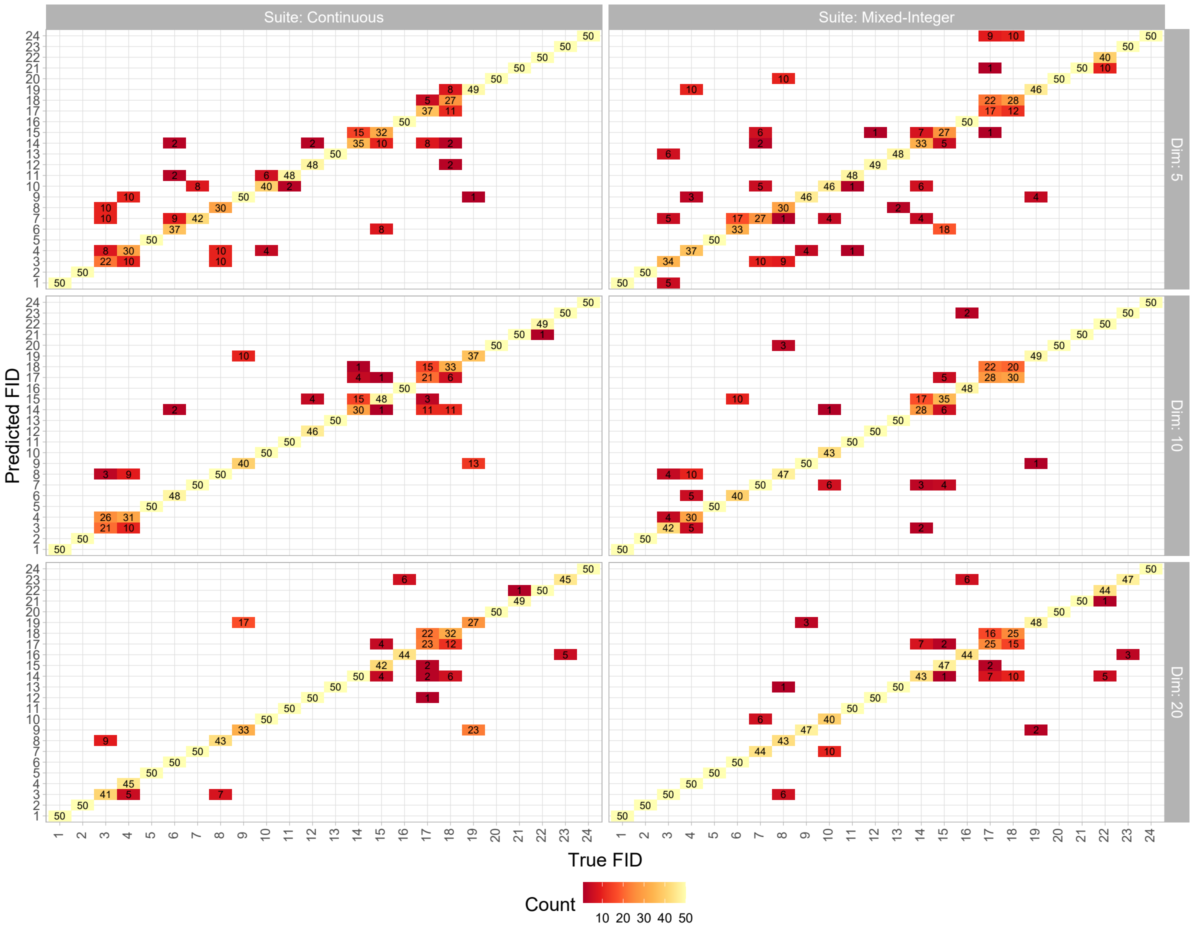

Figure 2: Confusion matrix of the classification scenario. The classifier trained for `cont-BBOB` achieves an $F1$-score of 0.863 on the validation set where in the case of `mixint-BBOB` an $F1$-score of 0.869 is reached.

nor indicative for the entire space of mixed-integer problems. As previously hinted at, mixint-BBOB is a discretized version of cont-BBOB where every decision variable is ordinal. To generalize to all mixed-integer problems, we also have to account for categorical decision variables in which binary variants are just a special case. Moreover, it can be questioned whether mix-int BBOB is truly indicative for mixted-integer problems and not just derivative of the existing cont-BBOB suite.

REFERENCES

[1] Pascal Kerschke, Holger H. Hoos, Frank Neumann, and Heike Trautmann. 2019. Automated Algorithm Selection: Survey and Perspectives. *Evolutionary Computation* 27, 1 (2019), 3 – 45.

[2] Pascal Kerschke and Heike Trautmann. 2019. Automated Algorithm Selection on Continuous Black-Box Problems By Combining Exploratory Landscape Analysis and Machine Learning. *Evolutionary Computation* 27, 1 (2019), 99 – 127.

[3] Fu Xing Long, Bas van Stein, Moritz Frenzel, Peter Krause, Markus Gitterle, and Thomas Bäck. 2022. Learning the Characteristics of Engineering Optimization Problems with Applications in Automotive Crash *(GECCO '22)*. Association for Computing Machinery, New York, NY, USA.

[4] Olaf Mersmann, Bernd Bischl, Heike Trautmann, Mike Preuss, Claus Weihs, and Günter Rudolph. 2011. Exploratory Landscape Analysis. In *Proceedings of the 13th Annual Conference on Genetic and Evolutionary Computation* (Dublin, Ireland) *(GECCO '11)*. Association for Computing Machinery, New York, NY, USA, 829–836.

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

[6] Raphael Patrick Prager and Heike Trautmann. 2023. Nullifying the Inherent Bias of Non-invariant Exploratory Landscape Analysis Features. In *Applications of Evolutionary Computation*, João Correia, Stephen Smith, and Raneem Qaddoura (Eds.). Springer Nature Switzerland, Cham, 411–425.

[7] Sebastian Raschka. 2018. MLxtend: Providing Machine Learning and Data Science Utilities and Extensions to Python's Scientific Computing Stack. *The Journal of Open Source Software (JOSS)* 3, 24 (April 2018).

[8] Quentin Renau, Johann Dreo, Carola Doerr, and Benjamin Doerr. 2021. Towards Explainable Exploratory Landscape Analysis: Extreme Feature Selection for Classifying BBOB Functions. *Proceedings of the 24th International Conference, EvoApplications 2021* 12694 LNCS (2021), 17–33.

[9] Moritz Vinzent Seiler, Raphael Patrick Prager, Pascal Kerschke, and Heike Trautmann. 2022. A Collection of Deep Learning-Based Feature-Free Approaches for Characterizing Single-Objective Continuous Fitness Landscapes. In *Proceedings of the Genetic and Evolutionary Computation Conference* (Boston, Massachusetts) *(GECCO '22)*. Association for Computing Machinery, New York, NY, USA, 657–665.

[10] Tea Tušar, Dimo Brockhoff, and Nikolaus Hansen. 2019. Mixed-Integer Benchmark Problems for Single- and Bi-Objective Optimization. In *Proceedings of the Genetic and Evolutionary Computation Conference* (Prague, Czech Republic) *(GECCO '19)*. Association for Computing Machinery, New York, NY, USA, 718–726.