

A feature-based performance analysis in evolutionary multiobjective optimization

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

▶ **To cite this version:**

Arnaud Liefooghe, Sébastien Verel, Fabio Daolio, Hernan Aguirre, Kiyoshi Tanaka. A feature-based performance analysis in evolutionary multiobjective optimization. 8th International Conference on Evolutionary Multi-Criterion Optimization (EMO 2015), 2015, Guimarães, Portugal. pp.95-109. hal-01093266

HAL Id: hal-01093266
https://hal.science/hal-01093266

Submitted on 19 Dec 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A Feature-based Performance Analysis in Evolutionary Multiobjective Optimization

Arnaud Liefooghe[1], Sébastien Verel[2],
Fabio Daolio[3], Hernán Aguirre[3], and Kiyoshi Tanaka[3]

[1] Université Lille 1, CRIStAL (UMR CNRS 9189) — Inria Lille-Nord Europe, France
arnaud.liefooghe@univ-lille1.fr
[2] Université du Littoral Côte d'Opale, LISIC, France
verel@lisic.univ-littoral.fr
[3] Shinshu University, Faculty of Engineering, Nagano, Japan
fdaolio@shinshu-u.ac.jp, ahernan@shinshu-u.ac.jp, ktanaka@shinshu-u.ac.jp

Abstract. This paper fundamentally investigates the performance of evolutionary multiobjective optimization (EMO) algorithms for computationally hard 0–1 combinatorial optimization, where a strict theoretical analysis is generally out of reach due to the high complexity of the underlying problem. Based on the examination of problem features from a multiobjective perspective, we improve the understanding of the efficiency of a simple dominance-based EMO algorithm with unbounded archive for multiobjective NK-landscapes with correlated objective values. More particularly, we adopt a statistical approach, based on simple and multiple linear regression analysis, to enquire the expected running time of global SEMO with restart for identifying a $(1+\varepsilon)-$approximation of the Pareto set for small-size enumerable instances. Our analysis provides further insights on the EMO search behavior and on the most important features that characterize the difficulty of an instance for this class of problems and algorithms.

1 Introduction

Black-box multiobjective combinatorial optimization problems are characterized by a discrete solution space and by multiple objective functions, such as cost, profit, or risk, that are ill-defined, computationally expensive, or for which an analytical form is not available. Due to the black-box nature of the objective functions, problem-specific algorithms are usually excluded to identify or approximate the Pareto set, so that an increasing number of general-purpose evolutionary multiobjective optimization (EMO) algorithms and other randomized search heuristics have been proposed in recent years [3]. However, the overall amount of understanding about the pros and cons of different EMO algorithm designs and configurations with respect to a given problem structure is rather scarce. Due to the increasing number and complexity of black-box multiobjective optimization problems and algorithms, one of the most difficult challenges is to devise and exhibit a number of general-purpose problem characteristics and

statistical methodologies allowing to explain the dynamics and the performance of EMO algorithms.

Recently, a few attempts to explain the performance of randomized search heuristics based on relevant fitness landscape features have been proposed for single-objective optimization problems of continuous and combinatorial nature; see *e.g.* [4, 6, 12]. In this paper, we address the issue of feature-based performance analysis for EMO algorithms according to the main characteristics of 0–1 multiobjective optimization problems. We first extend our previous works by summarizing a number of problem properties and fitness landscape features for black-box 0–1 multiobjective optimization [11]. They include features extracted from the problem input data, like variable correlation, objective correlation, and objective space dimension [17], as well as features from the Pareto set [1, 8], the Pareto graph [14] and the ruggedness and multimodality of the fitness landscape [17]. Then, we analyze the correlation between those features and the performance of an EMO algorithm. More particularly, we investigate the expected running time of the global SEMO algorithm [9] with restart to identify a $(1+\varepsilon)-$approximation of the Pareto set on a large number of small-size enumerable multiobjective NK-landscapes with objective correlation, *i.e.* ρMNK-landscapes [17]. Our analysis shows the relative influence of each *individual* problem feature on the algorithm performance. In particular, the running time of global SEMO appears to be predominantly impacted by the ruggedness of the fitness landscape, more than other features like the number of Pareto optimal solutions. Additionally, we investigate different formulations of a multiple linear regression model. This allows us to discuss the *joint* effect of different subsets of features in capturing the dynamics of the algorithm. The ruggedness and the multimodality, but also the number of objectives, the correlation between them, and the hypervolume of the Pareto front turn out to be the most impactful characteristics that allow to explain the performance of global SEMO for ρMNK-landscapes.

The remainder of the paper is organized as follows. Section 2 details the problem and algorithm settings of our analysis. Section 3 summarizes the problem features under consideration in the paper. Section 4 introduces different regression models to explain the performance of global SEMO for enumerable ρMNK-landscapes. Section 5 concludes the paper and suggests further research.

2 Problem and Algorithm Settings

In this paper, we are interested in the ability of evolutionary multiobjective optimization (EMO) algorithms to identify a Pareto set approximation for black-box multiobjective combinatorial optimization problems. In particular, we investigate the (estimated) running time of global SEMO [9] with restart to identify a $(1+\varepsilon)-$approximation of the Pareto set on a large bench of enumerable ρMNK-landscapes with different structural properties. We consider the maximization of an objective function vector $f = (f_1, \ldots, f_m)$ over the discrete set of solutions $X = \{0,1\}^n$, where m is the number of objectives, and n is the problem size. X is the *solution space*, and $Z = f(X) \subseteq \mathbb{R}^m$ is the *objective space*. A solution

$x \in X$ is dominated by a solution $x' \in X$ if $\forall i \in \{1,\ldots,m\}$, $f_i(x) \leqslant f_i(x')$ and $\exists i \in \{1,\ldots,m\}$ such that $f_i(x) < f_i(x')$. The set of solutions that are not dominated by any other is the *Pareto set*, and its image in the objective space is the *Pareto front*.

2.1 ρMNK-Landscapes

The family of ρMNK-landscapes constitutes a problem-independent model used for constructing multiobjective multimodal landscapes with objective correlation [17]. They extend single-objective NK-landscapes [7] and multiobjective NK-landscapes with independent objective functions [1]. Feasible solutions are binary strings of size n, *i.e.* the solution space is $X = \{0,1\}^n$. The parameter k refers to the number of variables that influence a particular position from the bit-string. The objective function vector $f = (f_1,\ldots,f_i,\ldots,f_m)$ is defined as $f : \{0,1\}^n \rightarrow [0,1]^m$ such that each objective function f_i is to be maximized. The problem can be formalized as follows.

$$\begin{array}{rll} \max & f_i(x) = \dfrac{1}{n}\displaystyle\sum_{j=1}^{n} f_{ij}(x_j, x_{j_1},\ldots,x_{j_k}) \ , & i \in \{1,\ldots,m\} \\ \text{s.t.} & x_j \in \{0,1\} & , \quad j \in \{1,\ldots,n\} \end{array} \tag{1}$$

As in the single-objective case, each separate objective function value $f_i(x)$ of a solution $x = (x_1,\ldots,x_j,\ldots,x_n)$ is an average value of the individual contributions associated with each variable x_j. Indeed, for each objective f_i, $i \in \{1,\ldots,m\}$, and each variable x_j, $j \in \{1,\ldots,n\}$, a component function $f_{ij} : \{0,1\}^{k+1} \rightarrow [0,1]$ assigns a real-valued contribution for every combination of x_j and its k *epistatic interactions* $\{x_{j1},\ldots,x_{jk}\}$. These f_{ij}-values are uniformly distributed in the range $[0,1]$. As a consequence, the individual contribution of a variable x_j depends on the value of x_j, as well as on the values of $k < n$ other variables $\{x_{j_1},\ldots,x_{j_k}\}$. In this work, the epistatic interactions, *i.e.* the k variables that influence the contribution of x_j, are set uniformly at random among the $(n-1)$ variables other than x_j, following the random neighborhood model from [7]. By increasing the number of epistatic interactions k from 0 to $(n-1)$, problem instances can be gradually tuned from smooth to rugged. In ρMNK-landscapes, f_{ij}-values additionally follow a multivariate uniform distribution of dimension m, defined by an $m \times m$ positive-definite symmetric covariance matrix (c_{pq}) such that $c_{pp} = 1$ and $c_{pq} = \rho$ for all $p,q \in \{1,\ldots,m\}$ with $p \neq q$, were $\rho > \frac{-1}{m-1}$ defines the objective correlation degree; see [17] for details. The positive (respectively negative) data correlation ρ allows to decrease (respectively increases) the degree of conflict between the objective function values. The same correlation coefficient ρ is then defined between all pairs of objectives, and the same epistatic degree k and epistatic interactions are set for all the objectives.

2.2 Global SEMO

Global SEMO [9], or G-SEMO for short, is a simple elitist steady-state EMO algorithm for black-box 0–1 optimization problems dealing with an arbitrary

objective function vector defined as $f : \{0,1\}^n \rightarrow Z$ such that $Z \subseteq \mathbb{R}^m$, like in ρMNK-landscapes. It maintains an *unbounded* archive A of non-dominated solutions found so far. The archive is initialized with one random solution from the solution space. At each iteration, one solution is chosen at random from the archive $x \in A$. Each binary variable from x is independently flipped with a rate of $\frac{1}{n}$ in order to produce an offspring solution x'. This mutation operator is ergodic, meaning that there is a non-zero probability of jumping from any point to any other point in the solution space. The archive is then updated by keeping the non-dominated solutions from $A \cup \{x'\}$. In its general form, the G-SEMO algorithm does not have any explicit stopping rule [9]. In this paper, we are interested in its running time, in terms of a number of function evaluations, until an $(1+\varepsilon)-$approximation of the Pareto set has been identified and is contained in the internal memory A of the algorithm, subject to a maximum budget of function evaluations.

2.3 Performance Measure

Let ε be a constant value such that $\varepsilon \geqslant 0$. The (multiplicative) ε-dominance relation ($\preceq_\varepsilon$) can be defined as follows. For all $x, x' \in X$, $x \preceq_\varepsilon x'$ if $f_i(x) \leqslant (1+\varepsilon) \cdot f_i(x')$, $\forall i \in \{1, \ldots, m\}$. A set $X^\varepsilon \subseteq X$ is an $(1+\varepsilon)-$approximation of the Pareto set if for any solution $x \in X$, there is one solution $x' \in X^\varepsilon$ such that $x \preceq_\varepsilon x'$. This is equivalent to finding an approximation set whose multiplicative epsilon quality indicator value with respect to the (exact) Pareto set is lower than $(1+\varepsilon)$, see *e.g.* [18]. Interestingly, under some general assumptions, there always exists an $(1+\varepsilon)$-approximation, for any given $\varepsilon \geqslant 0$, whose cardinality is both polynomial in the problem size and in $\frac{1}{\varepsilon}$ [13].

Following a conventional methodology from single-objective continuous black-box optimization benchmarking [5], the expected number of function evaluations to identify an $(1+\varepsilon)-$approximation is here chosen as a performance measure. However, as any EMO algorithm, G-SEMO can either succeed or fail to reach an accuracy of ε in a single simulation run. In case of a success, the running time is the number of function evaluations until an $(1+\varepsilon)-$approximation was found. In case of a failure, we simply *restart* the algorithm at random. We then obtain a "*simulated running time*" [5] from a set of given trials of G-SEMO on a given instance. Such a performance measure allows to take into account both the success rate $p_s \in (0,1]$ and the convergence speed of the G-SEMO algorithm with restarts. Indeed, after $(t-1)$ failures, each one requiring T_f evaluations, and the final successful run with T_s evaluations, the total running time is $T = \sum_{i=1}^{t-1} T_f + T_s$. By taking the expectation value and by considering that the probability of success after $(t-1)$ failures follows a Bernoulli distribution of parameter p_s, we have:

$$\mathbb{E}[T] = \left(\frac{1-p_s}{p_s} \right) \mathbb{E}[T_f] + \mathbb{E}[T_s] \tag{2}$$

In our case, the success rate p_s is estimated with the ratio of successful runs over the total number of executions ($\hat{p}_s$), the expected running time for unsuccessful

runs $\mathbb{E}[T_f]$ is set to a constant limit on the number of function evaluation calls T_{max}, and the expected running time for successful runs $\mathbb{E}[T_s]$ is estimated with the average number of function evaluations performed by successful runs.

$$\texttt{ert} = \left(\frac{1-\hat{p}_s}{\hat{p}_s}\right) T_{max} + \frac{1}{t_s}\sum_{i=1}^{t_s} T_i \tag{3}$$

where t_s is the number of successful runs, and T_i is the number of evaluations required for successful run i. For more details, we refer to [5].

2.4 Parameter Setting

In the following, we investigate ρMNK-landscapes with an epistatic degree $k \in \{2, 4, 6, 8, 10\}$, an objective space dimension $m \in \{2, 3, 5\}$, and an objective correlation $\rho \in \{-0.9, -0.7, -0.4, -0.2, 0.0, 0.2, 0.4, 0.7, 0.9\}$, such that $\rho > \frac{-1}{m-1}$. The problem size is set to $n = 18$ in order to enumerate the solution space exhaustively. The solution space size is then 2^{18}. A set of 30 different landscapes, independently generated at random, are considered for each parameter combination: ρ, m, and k. They are made available at the following URL: `http://mocobench.sf.net`.

We set a target $\varepsilon = 0.1$. The time limit is set to $T_{max} = 2^n \cdot 10^{-1} < 26\,215$ function evaluations without identifying an $(1+\varepsilon)-$approximation. The G-SEMO algorithm is executed 100 times *per* instance. For a given instance, the success rate and the expected number of evaluations for successful runs are estimated from those 100 executions. However, let us note that G-SEMO was not able to identify a $(1+\varepsilon)-$approximation set for any of the runs on one instance with $m = 3$, $\rho = 0.2$ and $k = 10$, one instance with $m = 3$, $\rho = 0.4$ and $k = 10$, ten instances with $m = 5$, $\rho = 0.2$ and $k = 10$, six instances with $m = 5$, $\rho = 0.4$ and $k = 10$, as well as two instances with $m = 5$, $\rho = 0.7$ and $k = 10$. Moreover, G-SEMO was not able to solve the following instances due to an overload of CPU resources available: $m = 5$ and $\rho \in \{-0.2, 0.0\}$. Those experiments have then been discarded due to missingness. Overall, this represents a total amount of 2 980 instances times 100 executions, that is 298 000 simulation runs.

3 Features to Characterize Problem Difficulty

In Table 1, we give a number of general-purpose problem features, either directly extracted from the problem instance (*low-level features*), or computed from the enumerated Pareto set and solution space (*high-level features*). Obviously, since the features require the solution space to be completely enumerated, they are not practical for performance prediction purposes. However, we still include them in order to examine their impact on the algorithm performance. For the case of ρMNK-landscapes, the neighborhood is induced by the *bit-flip* operator, which is directly related to the Hamming distance between solutions. For the computation of the hypervolume, the reference point is set to the origin $z^\star = (0.0, \ldots, 0.0)$. For

Table 1. Summary of low-level and high-level features investigated in the paper.

low-level features

(k)	Number of variable (epistatic) interactions
(m)	Number of objective functions
(ρ)	Correlation between the objective function values

high-level features

(`npo`)	Number of Pareto optimal solutions	[1, 8]
(`hv`)	Hypervolume value [18] of a the Pareto set	[1]
(`avgd`)	Average distance between Pareto optimal solutions	[11]
(`maxd`)	Maximum distance between Pareto optimal solutions	[11]
(`nconnec`)	Number of connected components in the Pareto set	[14]
(`lconnec`)	Proportion of the largest connected component of the Pareto set	[10]
(`kconnec`)	Minimal Hamming distance to connect the Pareto set	[14]
(`nplo`)	Number of Pareto local optimal solutions	[15]

a more comprehensive explanation of those features and a correlation analysis between them, we refer to [11]. In the next section, we relate the value of those features for enumerable ρMNK-landscapes to the performance of G-SEMO.

4 Problem Features *vs.* Algorithm Performance

In this section, we conduct a linear regression analysis on the correlation between the problem features presented in the previous section and the performance of G-SEMO. The algorithm performance is defined as the expected running time `ert`, in terms of the number of evaluation function calls, required by the algorithm to identify a $(1+\varepsilon)-$approximation of the Pareto set. We first detail our methodological setup. Then, we analyze the *individual* as well as the *joint* impact of problem features on the algorithm performance. At last, we compare the accuracy of regression models for different objective space dimensions.

4.1 Methodological Setup

Linear Regression. In order to provide an explanatory model for the algorithm performance, we perform a linear regression, whose general model can be formalized as follows:

$$y = \beta_0 + \beta_1 \cdot v_1 + \beta_2 \cdot v_2 + \ldots + \beta_p \cdot v_p + e \tag{4}$$

where y is the response variable, $(v_1, v_2, \ldots, v_p)$ are the explanatory variables, and e is the usual error term. In our case, the response variable to be explained is the expected running time of G-SEMO: $\log(\texttt{ert})$, and the p explanatory variables correspond to selected problem features as detailed in Section 3. The response of the linear model is here log-transformed in order to better approach linearity; see

Fig. 1 (top-left). Also, an order of magnitude is in general sufficiently relevant for the running time of an EMO algorithm. Such linear regression models are usually fitted using an ordinary least-square minimization. A least-square estimator $\hat{\beta}_i$ is then produced for each regression coefficient β_i, $i \in \{1, \ldots, p\}$, by minimizing the sum of squared residuals between the fitted and the observed values. The case of a single explanatory variable ($p = 1$) is known as *simple* linear regression, whereas the extension to more than one explanatory variables ($p > 1$) is known as *multiple* linear regression. Notice that, although a linear regression model might not be able to reliably catch the existing correlation between explanatory variables as well as non-linear dependencies with the response variable, it has the advantage of being simple and easily interpretable.

Regression Accuracy. In the following, when measuring the accuracy of a linear regression model, we will be interested in the following statistics:

- The absolute *correlation coefficient* (r) measures the linear association between the predicted and the actually observed values (the conventional Pearson correlation coefficient is here used). Its absolute value ranges from 0 to 1. The closer r to 1.0, the better the fitting. Actually, an r-value of 1.0 indicates that the linear regression line perfectly fits the data.
- The *mean absolute error* (MAE) measures the average value of the absolute difference between the values predicted by the regression model and the values actually observed (the residuals). It aggregates the magnitudes of the prediction errors into a single predictive power measure to compare different models. Clearly, the lower the MAE, the better the regression model.
- The *root mean-square error* (RMSE) measures the square root of the average squared difference between the values predicted by the regression model and the values actually observed. Similarly, the lower the RMSE, the better the regression model. Notice that the RMSE tends to favor a regression model that avoids large errors even though it produces a less satisfactory fit overall, whereas the MAE tends to favor a regression model that produces occasional large errors while being reasonably good on average.
- The *relative absolute error* (%RAE) corresponds to the MAE relative to the basic model that always predicts the mean, *i.e.* when no explanatory variables are used in the regression model ($p = 0$). As a consequence, smaller values are better, and a RAE higher than 100% indicates that the corresponding model is actually *worse* than this basic model in terms of MAE.
- The *root relative squared error* (%RRSE) corresponds to the RMSE relative to the basic model that always predicts the mean.

For each of those statistics, we report the values observed on the *training set*, *i.e.* the set of data used to build the model. In addition, we perform a 10-fold *cross-validation* in order to assess how the results of the regression model generalize to an independent data set. The original data set is *randomly* partitioned into 10 samples of equal size. The cross-validation process is repeated 10 times, with each of the samples being used exactly once as the validation data. For each sample, the above statistics are computed, and then averaged in order to produce a cross-validated r, MAE, RMSE, %RAE, and %RRSE value.

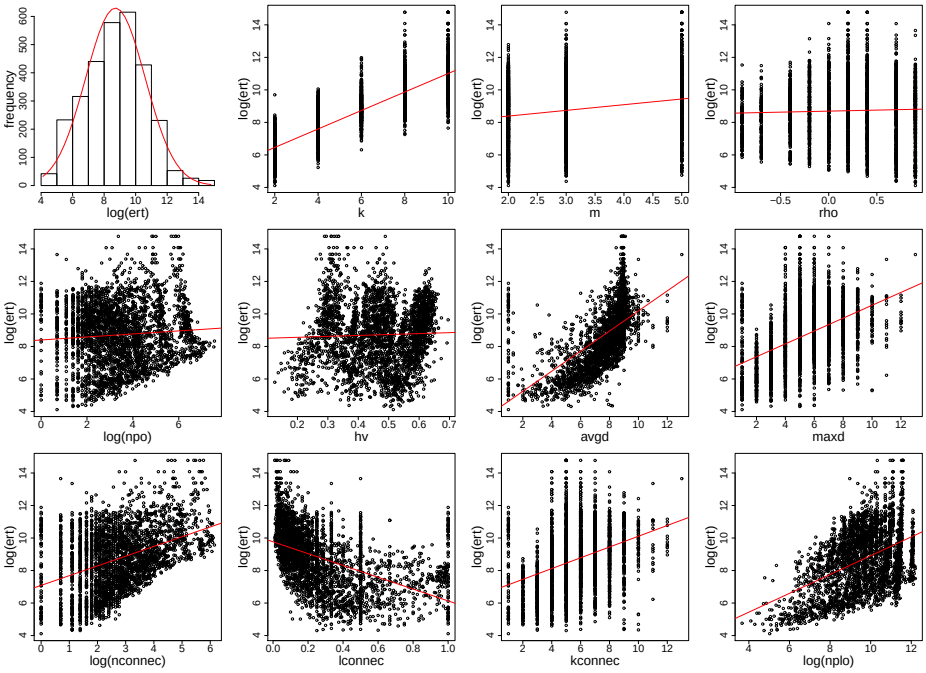


Fig. 1. Histogram of the distribution of log(`ert`)-values over all the instances (top-left), as well as scatter plots and regression lines for each feature *vs.* log(`ert`).

Data Preparation. Although features might usually have to be normalized appropriately in order to get rid of scaling issues and for a fair comparison between them, we here chose *not* to normalize them in order to ease the interpretation of the different models. Notice that normalizing the features would only result in a change on the value of the estimated regression coefficient $\hat{\beta}_i$, $i \in \{1, \ldots, p\}$. Moreover, this is not an issue within our experiments since we explicitly generate instances of the same size ($n = 18$) that takes their objective values in the same hyper-box $[0, 1]^m$.

4.2 Individual Impact of Problem Features

The scatter plots between each feature and the log-transformed estimated running time of G-SEMO log(`ert`) is reported in Fig. 1. Notice that some features (`npo`, `nconnec`, `nplo`) have been log-transformed in order to better approach linearity. Additionally, the statistics of all possible simple linear regression models, one for each feature, are reported in Table 2, from the lowest to the highest absolute correlation value. The individual impact of each feature is analyzed below.

First of all, four features are not directly linearly correlated to the expected running time of G-SEMO: the number of objective functions m, the objective correlation ρ, the cardinality of the Pareto set log(`npo`), and the hypervolume `hv`.

Table 2. Summary statistics of simple linear regression models, each one being based on a single problem feature. Values are rounded to 10^{-2}.

	training set					10-fold cross validation				
feature	r	MAE	RMSE	%RAE	%RRSE	r	MAE	RMSE	%RAE	%RRSE
ρ	0.03	1.52	1.89	99.66	99.94	0.01	1.53	1.88	99.75	100.00
`hv`	0.04	1.52	1.89	99.66	99.94	0.02	1.53	1.89	99.73	99.99
log(`npo`)	0.08	1.54	1.88	100.61	99.70	0.07	1.54	1.88	100.66	99.74
m	0.21	1.51	1.85	98.63	97.86	0.20	1.51	1.85	98.69	97.91
`kconnec`	0.37	1.38	1.76	89.99	93.03	0.37	1.38	1.76	90.01	93.06
log(`nconnec`)	0.40	1.44	1.73	94.21	91.47	0.40	1.44	1.73	94.27	91.52
log(`nplo`)	0.46	1.41	1.67	92.53	88.56	0.46	1.42	1.67	92.57	88.59
`maxd`	0.47	1.30	1.67	85.07	85.07	0.47	1.30	1.67	85.09	88.36
`lconnec`	0.49	1.31	1.65	85.67	87.35	0.49	1.31	1.65	85.69	87.37
`avgd`	0.60	1.15	1.51	75.31	79.95	0.60	1.15	1.51	75.33	80.00
k	0.85	0.77	1.00	50.67	52.95	0.85	0.78	1.00	50.67	52.96
none	0.00	1.53	1.89	100.00	100.00	0.04	1.53	1.89	100.00	100.00

For each of these features, the absolute correlation coefficient value is under 0.25, and the prediction error is around the one of the most basic model that always predicts the mean. Surprisingly, there is *no* direct connection with the two main low-level features from problem input data m and ρ. At least, the link between the running time and those features is not a direct linear correlation, but a more complex model will be analyzed in the next section. As well, the cardinality and the hypervolume of the Pareto set, features closely related to the final goal of the search process, do not explain the variance of log(`ert`) by themselves.

The features related to the connectedness of the Pareto set are all weakly correlated to log(`ert`). The absolute correlation coefficients of the number of connected components log(`nconnec`), the proportional size of the largest connected component `lconnec`, and the minimum distance to be connected `kconnec` are between 0.37 and 0.49. The more connected the Pareto set, the smaller the running time of G-SEMO. The algorithm performance is also moderately correlated with the average and maximal distance between Pareto optimal solutions, `avgd` and `maxd` (the absolute correlation coefficient values are 0.60 and 0.47, respectively). The larger the distance between Pareto optimal solutions in the solution space, the larger the running time of G-SEMO. Interestingly, the cardinality of the Pareto set has a smaller impact on the performance of G-SEMO than the distance between solutions in the Pareto set. Moreover, the multimodality of the landscape, in terms of the number of Pareto local optimal solutions log(`nplo`), is moderately correlated to the running time of G-SEMO: the more Pareto local optima, the longer the running time (the correlation coefficient is 0.46).

At last, the only strong correlation appears with the feature related to the ruggedness of the landscape. Indeed, the number of epistatic interactions k is highly correlated to the efficiency of G-SEMO (the correlation coefficient is 0.85). The more rugged the landscape, the longer it takes to identify a $(1+\varepsilon)-$approximation of the Pareto set. In other words, by taking the features individually, the model based on k is the one that gives the highest accuracy. On average, it allows to predict the logarithm of the runtime of G-SEMO within ± 0.77 of the observed value. Since the RMSE is much larger (1.00), this sug-

gests that the deviation to this average value might be large, as we can see also on Fig.1 (first line, second column). Such a regression accuracy is around twice better than the most basic model that always predicts the mean (± 1.53), and largely better than the second more accurate simple linear regression model based on `avgd` (± 1.15).

Having the ruggedness of the landscape k as a more important (individual) feature than the number of objectives m and the objective correlation ρ might be surprising at first sight. Indeed, the number of Pareto optimal solutions is known to increase exponentially with the number of objectives and the degree of conflict between them; see *e.g.* [17]. However, let us remind that the algorithm under consideration in the paper (G-SEMO) actually handles an *unlimited* approximation set size. It is then only slightly affected by the minimum number of solutions required to obtain a $(1+\varepsilon)-$approximation of the Pareto set [13]. Actually, depending on ρ, the estimated expected running time of G-SEMO is 23 to 118 times larger for rugged two-objective instances than smoother five-objective instances. For instance, when $\rho = 0.2$, the average `ert`$-$value is equal to 61 836 for $k = 10$ and $m = 2$ while it is equal to 2 691 for $k = 2$ and $m = 5$. Similarly, for $\rho = 0.9$, the average `ert`$-$value is 33 922 for $k = 10$ and $m = 2$, and only 287 for $k = 2$ and $m = 5$. In accordance with known results from single-objective optimization [2], the ruggedness of the landscape k seems to largely impact the running time of EMO algorithms.

Overall, analyzing the individual impact of problem features supports the hypothesis that the structural properties identified in the previous section can help to understand the performance of a simple dominance-based EMO algorithm like G-SEMO for ρMNK-landscapes. In the next section, different multiple linear regression models are examined in order to better explain the running time of G-SEMO, based on the *joint effect* of these problem-related characteristics.

4.3 Joint Impact of Problem Features

We start by fitting the response variable log(`ert`) against all the low-level and high-level features presented in Section 3. The statistics related to the model accuracy are provided in Table 3 (line 1). For this complete multiple linear regression model, the correlation coefficient is over 0.9. Overall, it allows to explain the performance of G-SEMO with a much higher accuracy compared to the simple linear regression model based on k only, and outperforms the basic model that always predicts the mean by around 60%. Moreover, the model has a high degree of generalization. Indeed, the RMSE and the cross-validated RMSE are very close to each other. The same happens with the MAE (the difference between both is always under 10^{-2}). The scatter plot of the actual *vs.* the predicted performance values is given in Fig. 2 (left). This allows us to visualize how the model accuracy varies depending on the hardness of the problem instance: The model seems to slightly underestimate the runtime for easier and harder instances whereas it rather overestimates it for an intermediate instance difficulty.

Table 3. Summary statistics of the multiple linear regression model with backward-elimination feature selection. Values are rounded to 10^{-2}.

	training set					10-fold cross validation				
	r	MAE	RMSE	%RAE	%RRSE	r	MAE	RMSE	%RAE	%RRSE
all features	0.91	0.58	0.76	37.74	40.34	0.91	0.58	0.76	37.86	40.48
\ `maxd`	0.91	0.58	0.76	37.73	40.38	0.91	0.58	0.76	37.84	40.50
\ log(`nconnec`)	0.91	0.58	0.76	37.73	40.49	0.91	0.58	0.77	37.83	40.60
\ log(`npo`)	0.91	0.58	0.77	37.72	40.53	0.91	0.58	0.77	37.80	40.62
\ `lconnec`	0.91	0.58	0.77	37.74	40.54	0.91	0.58	0.77	37.81	40.61
\ `avgd`	0.91	0.58	0.77	37.83	40.58	0.91	0.58	0.77	37.90	40.64
\ `kconnec`	0.91	0.58	0.77	37.95	40.66	0.91	0.58	0.77	38.00	40.71
\ ρ	0.91	0.61	0.80	39.98	42.20	0.91	0.61	0.80	40.04	42.26
\ `hv`	0.89	0.67	0.87	43.59	46.29	0.89	0.67	0.87	43.63	46.33
\ log(`nplo`)	0.88	0.70	0.91	45.83	48.19	0.88	0.70	0.91	45.86	48.21
\ m	0.85	0.77	1.00	50.67	52.95	0.85	0.78	1.00	50.67	52.96
\ k	0.00	1.53	1.89	100.00	100.00	0.04	1.53	1.89	100.00	100.00

Although the regression coefficients related to each feature are not interpretable due to the different scaling of the metric values, not all regression coefficients are statistically significant in a general multiple linear regression model. In order to eliminate the influence of the less significant regression coefficients, we proceed by *backward elimination*. Starting from the inclusion of all features, we iteratively remove the feature that has the lowest impact on the increase of the MAE until no feature remains. This allows us to produce a ranked list of features by traversing the feature space from one side to the other and recording the order that attributes are deleted. Hence, the attributes that are deleted in the last steps have a more meaningful impact on the model. The steps of the backward elimination are sketched in Table 3, where one feature is removed at every line. Notice that a forward selection, that does the opposite procedure of iteratively adding attributes, ends up with a similar ranking on the importance of features, except that `maxd` is in the latter case more important than log(`npo`) and log(`nconnec`) (detailed results are not reported due to space limitation).

This feature selection analysis allows us to gain further insights about which subset of features obtains the highest accuracy. Indeed, the error increase is almost insignificant until the deletion of ρ in the model (line 8 in Table 3), where the correlation coefficient drops from 0.91 to 0.89, the MAE rises from 0.58 to 0.61 and the RMSE rises from 0.77 to 0.80. We can then conclude that ρ and subsequent attributes constitute the most significant subset for explaining the algorithm performance. Actually, a more compact model, with only the five most significant features, constitute an acceptable alternative, and has almost the same accuracy than the full model; see also Fig. 2 (right). Once again, the most important feature seems to be k, now followed by m, log(`nplo`), `hv` and ρ, in the order of importance. As a consequence, although they are not able to catch all the variations of log(`ert`) individually (see Section 4.2), the joint effect of all three low-level features from the problem input data (k, m and ρ) is relevant for explaining the running time of G-SEMO. Moreover, there is one high-level feature related to the hypervolume of the Pareto set `hv` and to the multimodality log(`nplo`). As well, the number of Pareto optimal solutions is *not* a significant addition to the regression model. We attribute this to the fact that

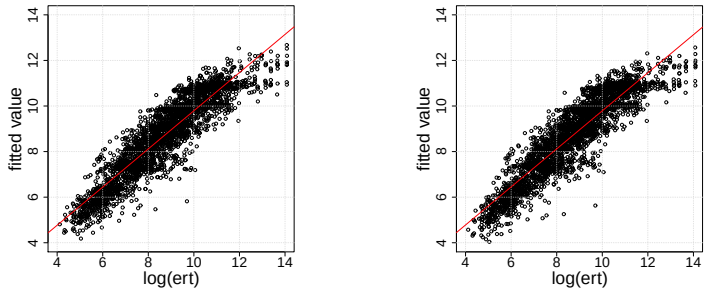


Fig. 2. log(`ert`) *vs.* fitted values for (left) the model with all features; and (right) the model with a selected subset of features (*i.e.* $k, m, \log(\texttt{nplo}), \texttt{hv}, \rho$).

the hypervolume incorporates a more relevant information related to the Pareto front for the algorithm behavior.

4.4 Explanatory Models *vs.* Objective Space Dimension

In this section, we build a separate regression model for each objective space dimension $m \in \{2, 3, 5\}$. The statistics related to the model accuracy are provided in Table 4. The information for the complete model mixing all m-values is also given in the tables in order to facilitate the comparison between the different models. Additionally, the scatter plot of the actual *vs.* the predicted performance values is given in Fig. 3.

First of all, the regression model for a particular number of objectives is always slightly more accurate than the global model for all m-values (whatever the m-value, there is an improvement in terms of correlation, %RAE or %RRSE over the complete model). Actually, the worsening in terms of MAE or RMSE for larger m-values is only an artefact that the expected running time of G-SEMO increases with the number of objectives; see Fig. 1 (first line, third column). This means that constructing a regression model *per* objective space dimension only allows to reduce the prediction error to a very small extent.

As reported in Table 5, by focusing on the five most relevant features identified in the previous section ($k, m, \log(\texttt{nplo}), \texttt{hv}$ and ρ, following the order of importance), we are also able to construct a more compact regression model

Table 4. Summary statistics of the multiple linear regression models for all objective space dimensions ($\forall m$) and for each individual objective space dimension $m \in \{2, 3, 5\}$. All features are part of the models. Values are rounded to 10^{-2}.

	training set					10-fold cross validation					
	r	MAE	RMSE	%RAE	%RRSE	r	MAE	RMSE	%RAE	%RRSE	#inst
$\forall m$	0.91	0.58	0.76	37.74	40.34	0.91	0.58	0.76	37.86	40.48	2980
$m = 2$	0.92	0.52	0.68	35.86	38.62	0.92	0.53	0.68	36.15	38.95	1350
$m = 3$	0.92	0.52	0.69	36.12	39.02	0.92	0.53	0.70	36.60	39.51	1048
$m = 5$	0.93	0.64	0.81	36.86	37.72	0.92	0.65	0.83	37.62	38.53	582

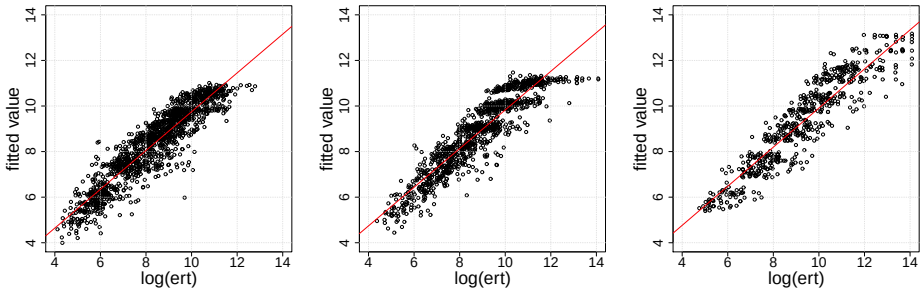


Fig. 3. log(`ert`) *vs.* fitted values for (left) $m = 2$, (middle) $m = 3$, (right) $m = 5$. All features are part of the models.

for each m-value with a satisfactory response compared to the model using all features. The difference in the correlation coefficient and the average error between the models with all features and their compact counterpart is always less than $2 \cdot 10^{-2}$. In fact, applying a feature selection procedure by backward elimination for each of the models until the MAE increases by more than 10^{-2} ends up with following subset feature selection, following the order of importance (detailed results are omitted due to space restriction):

- k, `hv` and log(`nplo`) for $m = 2$;
- k, log(`nplo`) and `hv` for $m = 3$;
- k and ρ for $m = 5$.

Interestingly, this means that the objective correlation ρ is *not* a relevant feature for the models with $m \in \{2, 3\}$. We attribute this to the fact that the number of Pareto local optimal solutions log(`nplo`) increases with ρ in these cases, whereas the correlation between both is much lower for $m = 5$. For the same reason, ρ is selected over `hv` in the latter case. In addition, given that the proportion of Pareto local optimal solutions in the solution space increases exponentially with m (we know, for instance, that more than 95% of the solution space correspond to Pareto local optimal solutions for $\rho = -0.2$ and $m = 5$ in average [17]), the multimodality of the landscape, corresponding to log(`nplo`), is not relevant anymore for larger objective space dimensions. However, in all cases, the ruggedness of the landscape k is again the most relevant feature.

Table 5. Summary statistics of the multiple linear regression models for all objective space dimensions ($\forall m$) and for each individual objective space dimension $m \in \{2, 3, 5\}$. Only the subset of significant features ($k, m, \rho,$ `hv`, log(`nplo`)) are part of the models. Values are rounded to 10^{-2}.

	training set					10-fold cross validation					
	r	MAE	RMSE	%RAE	%RRSE	r	MAE	RMSE	%RAE	%RRSE	#inst
$\forall m$	0.91	0.58	0.77	37.95	40.66	0.91	0.58	0.77	38.00	40.71	2980
$m = 2$	0.92	0.53	0.69	36.14	39.17	0.92	0.53	0.69	36.25	39.29	1350
$m = 3$	0.92	0.54	0.71	37.35	39.89	0.92	0.55	0.71	37.62	40.14	1048
$m = 5$	0.92	0.64	0.82	37.30	38.23	0.92	0.65	0.83	37.54	38.52	582

5 Conclusions

In this paper, we investigated the impact of problem features on the running time of a simple dominance-based EMO algorithm with restart, that maintains an unbounded archive of non-dominated solutions found so far. The topology of an arbitrary problem instance, in terms of ruggedness, multimodality, objective space dimension, objective correlation, cardinality and hypervolume of the Pareto set, as well as distance and connectedness between non-dominated solutions, has been examined for a large set of enumerable multiobjective NK-landscapes with objective correlation. First, a simple linear regression analysis revealed that the ruggedness of the landscape had the more critical effect on the algorithm performance. Second, a more-advanced multiple linear regression analysis allowed us to highlight the more significant subset of problem features. As in the single-objective case [2, 7], the ruggedness and the multimodality of the landscape affect the algorithm running time to a large extent. Additionally, the number of objectives, the correlation between them, and the hypervolume of the Pareto front to be covered are all jointly impactful in the multiobjective case. At last, although problem features have a different impact depending on the objective space dimension, the degree of explanation they are able to provide together is always as meaningful for the algorithm performance. Overall, our feature-based analysis was able to highlight the main relationships between the structural properties of the landscape and the performance of the algorithm. This allowed us to better understand the behavior and the performance of this EMO algorithm class.

The problem characteristics under analysis in the paper validate the relevance of our methodology for explaining the performance of EMO approaches. However, it remains an open question if there exist supplementary features that could better capture the problem difficulty for different problem and algorithm classes, and if more general regression models would allow to better apprehend the correlations among the features as well as their (non-linear) dependencies with the algorithm performance. Furthermore, the goal of the paper was on *understanding* the algorithm behavior and performance rather than blindly recommending the best-performing approach, but a natural extension for future research is to investigate the *prediction* power of the regression models proposed in the paper, based on existing works from single-objective optimization [6]. Following the *algorithm selection problem* formulated by Rice in the 1970s [16], this would allow us to design a portfolio approach for selecting the most appropriate algorithm configuration, based on a relevant structural characterization of the multiobjective problem instance to be solved. For that purpose, extending our paradigm with more-advanced regression models based on problem features that can be estimated inexpensively for large-size instances is currently under investigation.

Acknowledgements. This work was partially supported by the Japanese-French research project "Global Research on the Framework of Evolutionary Solution Search to Accelerate Innovation" (JSPS, 2013–2016).

References

1. Aguirre, H.E., Tanaka, K.: Working principles, behavior, and performance of MOEAs on MNK-landscapes. Eur J Oper Res 181(3), 1670–1690 (2007)
2. Barnett, L.: Ruggedness and neutrality - the NKp family of fitness landscapes. In: Sixth International Conference on Artificial Life (ALIFE VI). pp. 18–27 (1998)
3. Coello Coello, C.A., Lamont, G.B., Van Veldhuizen, D.A.: Evolutionary Algorithms for Solving Multi-Objective Problems. Springer, second edn. (2007)
4. Daolio, F., Verel, S., Ochoa, G., Tomassini, M.: Local optima networks and the performance of iterated local search. In: Genetic and Evolutionary Computation Conference (GECCO 2012). pp. 369–376 (2012)
5. Hansen, N., Auger, A., Ros, R., Finck, S., Pošík, P.: Comparing results of 31 algorithms from the black-box optimization benchmarking BBOB-2009. In: Genetic and Evolutionary Computation Conference (GECCO 2010). pp. 1689–1696 (2010)
6. Hutter, F., Xu, L., Hoos, H.H., Leyton-Brown, K.: Algorithm runtime prediction: Methods & evaluation. Artif Intell 206, 79–111 (2014)
7. Kauffman, S.A.: The Origins of Order. Oxford University Press (1993)
8. Knowles, J., Corne, D.: Instance generators and test suites for the multiobjective quadratic assignment problem. In: Evolutionary Multi-Criterion Optimization (EMO 2003). LNCS, vol. 2632, pp. 295–310. Springer (2003)
9. Laumanns, M., Thiele, L., Zitzler, E.: Running time analysis of evolutionary algorithms on a simplified multiobjective knapsack problem. Nat Comput 3(1), 37–51 (2004)
10. Liefooghe, A., Paquete, L., Figueira, J.R.: On local search for bi-objective knapsack problems. Evol Comput 21(1), 179–196 (2013)
11. Liefooghe, A., Verel, S., Aguirre, H., Tanaka, K.: What makes an instance difficult for black-box 0–1 evolutionary multiobjective optimizers? In: International Conference on Artificial Evolution (EA 2013). LNCS, vol. 8752, pp. 3–15 (2013)
12. Mersmann, O., Bischl, B., Trautmann, H., Wagner, M., Bossek, J., Neumann, F.: A novel feature-based approach to characterize algorithm performance for the traveling salesperson problem. Ann Math Artif Intell 69(2), 151–182 (2013)
13. Papadimitriou, C.H., Yannakakis, M.: On the approximability of trade-offs and optimal access of web sources. In: Symposium on Foundations of Computer Science (FOCS 2000). pp. 86–92 (2000)
14. Paquete, L., Stützle, T.: Clusters of non-dominated solutions in multiobjective combinatorial optimization: An experimental analysis. In: Multiobjective Programming and Goal Programming, LNMES, vol. 618, pp. 69–77. Springer (2009)
15. Paquete, L., Schiavinotto, T., Stützle, T.: On local optima in multiobjective combinatorial optimization problems. Ann Oper Res 156(1), 83–97 (2007)
16. Rice, J.R.: The algorithm selection problem. Adv Comput 15, 65–118 (1976)
17. Verel, S., Liefooghe, A., Jourdan, L., Dhaenens, C.: On the structure of multiobjective combinatorial search space: MNK-landscapes with correlated objectives. Eur J Oper Res 227(2), 331–342 (2013)
18. Zitzler, E., Thiele, L., Laumanns, M., Foneseca, C.M., Grunert da Fonseca, V.: Performance assessment of multiobjective optimizers: An analysis and review. IEEE Trans Evol Comput 7(2), 117–132 (2003)