



HAL
open science

Global vs local search on multi-objective NK-landscapes: contrasting the impact of problem features

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

► To cite this version:

Fabio Daolio, Arnaud Liefoghe, Sébastien Verel, Hernan Aguirre, Kiyoshi Tanaka. Global vs local search on multi-objective NK-landscapes: contrasting the impact of problem features. Genetic and Evolutionary Computation Conference (GECCO 2015), Jul 2015, Madrid, Spain. pp.369-376, 10.1145/2739480.2754745 . hal-01151882

HAL Id: hal-01151882

<https://hal.science/hal-01151882v1>

Submitted on 4 Jan 2017

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.

Global vs Local Search on Multi-objective NK-Landscapes: Contrasting the Impact of Problem Features

Fabio Daolio
Shinshu University
fdaolio@shinshu-u.ac.jp

Arnaud Liefooghe
Univ. Lille, CRISTAL, Inria
arnaud.liefooghe@inria.fr

Sébastien Verel
Univ. du Littoral Côte d'Opale
verel@lisic.univ-littoral.fr

Hernán Aguirre
Shinshu University
ahernan@shinshu-u.ac.jp

Kiyoshi Tanaka
Shinshu University
ktanaka@shinshu-u.ac.jp

ABSTRACT

Computationally hard multi-objective combinatorial optimization problems are common in practice, and numerous evolutionary multi-objective optimization (EMO) algorithms have been proposed to tackle them. Our aim is to understand which (and how) problem features impact the search performance of such approaches. In this paper, we consider two prototypical dominance-based algorithms: a global EMO strategy using an ergodic variation operator (GSEMO) and a neighborhood-based local search heuristic (PLS). Their respective runtime is estimated on a benchmark of combinatorial problems with tunable ruggedness, objective space dimension, and objective correlation (ρ MNK-landscapes). In other words, benchmark parameters define classes of instances with increasing empirical problem hardness; we enumerate and characterize the search space of small instances. Our study departs from simple performance comparison to systematically analyze the correlations between runtime and problem features, contrasting their association with search performance within and across instance classes, for both chosen algorithms. A mixed-model approach then allows us to further generalize from the experimental design, supporting a sound assessment of the joint impact of instance features on EMO search performance.

1. MOTIVATIONS

Despite the increasing number of available heuristics for evolutionary multi-objective optimization, the understanding of their search dynamics is comparatively less advanced than in the single-objective case. This knowledge is all the more valuable in a black-box scenario, where such approaches are mostly adopted. Finding what makes a problem hard, which features have an influence on algorithm performance, and how that changes across different algorithm and problem classes, are relevant and still open challenges in multi-objective optimization.

Following [17], we precisely investigate the impact of instance features on search performance. We decline this research question in three complementary sub-questions: (i) how strongly are features associated to performance? (ii) what performance change is

to be expected when varying each feature? (iii) what is the relative importance of features in explaining the performance variance? We consider a global and a local EMO heuristics and try to highlight the differences and similarities in the behavior of both algorithm classes, with respect to a set of relevant problem features. As such, the emphasis is more on making inferences, e.g. [8], rather than making predictions, e.g. [11]. Hence, this work attempts to provide both insightful understandings and methodological suggestions.

We introduce the experimental setup of our analysis in Section 2, in particular the considered problems, ρ MNK-landscapes [24], and the search algorithms, namely the global simple EMO optimizer (GSEMO) [14] and the Pareto local search (PLS) [21], of which we measure the estimated runtime to find a $(1 + \varepsilon)$ -approximation of the Pareto set. In Section 3, we examine the performance of GSEMO and PLS depending on the ruggedness, the objective space dimension, and the objective correlation of ρ MNK-landscapes. In Section 4, we provide an overview of the features that characterize the hardness of problem instances, relating to the difficulties randomized search heuristics might have to face. Via a correlation and regression analysis, we are able to point out how such features influence the performance of global and local search algorithms like GSEMO and PLS. The last section concludes the paper.

2. INGREDIENTS

2.1 Multi-objective Optimization

We are interested in maximizing a black-box objective function vector $f : X \rightarrow Z$, that maps any solution from the solution space $x \in X$ to a vector in the objective space $z \in Z$, with $Z \subseteq \mathbb{R}^M$, such that $z = f(x)$. We assume that the solution space is a discrete set $X = \{0, 1\}^N$, where N is the problem size, i.e. the number of binary (zero-one) variables. An objective vector $z \in Z$ is dominated by an objective vector $z' \in Z$, denoted by $z \prec z'$, iff for all $i \in \{1, \dots, M\}$ $z_i \leq z'_i$, and there is a $j \in \{1, \dots, M\}$ such that $z_j < z'_j$. Similarly, a solution $x \in X$ is dominated by a solution $x' \in X$ iff $f(x) \prec f(x')$. An objective vector $z^* \in Z$ is non-dominated if there does not exist any objective vector $z \in Z$ such that $z^* \prec z$. A solution $x^* \in X$ is non-dominated, or Pareto-optimal, if $f(x)$ is non-dominated. The set of Pareto-optimal solutions is the Pareto set (PS); its mapping in the objective space is the Pareto front (PF). One of the main challenges in multi-objective optimization is to identify the Pareto set/front, or a good approximation of it, for large-size and difficult problems.

2.2 ρ MNK-Landscapes

The family of ρ MNK-landscapes constitutes a problem-independent model used for constructing multi-objective multimodal land-

scapes with objective correlation [24]. They extend single-objective NK-landscapes [12] and multi-objective NK-landscapes with independent objective functions [1]. Candidate solutions are binary strings of size N , i.e. the solution space is $X = \{0, 1\}^N$. The objective function vector $f = (f_1, \dots, f_i, \dots, f_M)$ is defined as $f : \{0, 1\}^N \rightarrow [0, 1]^M$ such that each objective function f_i is to be maximized. As in the single-objective case, each separate objective function value $f_i(x)$ of a solution $x = (x_1, \dots, x_j, \dots, x_N)$ is an average value of the individual contributions associated with each variable x_j . Indeed, for each objective f_i , $i \in \{1, \dots, M\}$, and each variable x_j , $j \in \{1, \dots, N\}$, 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 $\{x_{j_1}, \dots, x_{j_K}\}$. These f_{ij} -values are uniformly distributed in the range $[0, 1]$. Thus, the individual contribution of a variable x_j depends on its value and on the values of $K < N$ other variables $\{x_{j_1}, \dots, x_{j_K}\}$. The problem can be formalized as follows.

$$\begin{aligned} \max \quad & f_i(x) = \frac{1}{N} \sum_{j=1}^N f_{ij}(x_j, x_{j_1}, \dots, x_{j_K}) \quad i \in \{1, \dots, M\} \\ \text{s.t.} \quad & x_j \in \{0, 1\} \quad j \in \{1, \dots, N\} \end{aligned}$$

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 [12]. By increasing the number of epistatic interactions K from 0 to $(N - 1)$, problem instances can be gradually tuned from smooth to rugged. In ρ MNK-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 $p, q \in \{1, \dots, M\}$ with $p \neq q$, where $\rho > \frac{-1}{M-1}$ defines the objective correlation degree; see [24] for details. The positive (respectively negative) data correlation ρ decreases (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.

2.3 Multi-objective Search Heuristics

In this paper, we consider two randomized search heuristics for ρ MNK-landscapes: (i) global SEMO (GSEMO) [14], a simple elitist steady-state *global* EMO algorithm (see Algorithm 1); and (ii) Pareto local search (PLS) [21], a population-based multi-objective *local* search (Algorithm 2). Both algorithms maintain an *unbounded* archive A of mutually non-dominated solutions. This archive is initialized with one random solution from the solution space. Then, at each iteration, one solution is selected at random from the archive $x \in A$. In GSEMO, each binary variable from x is independently flipped with rate $\frac{1}{N}$ in order to produce an offspring solution x' . The archive is then updated by keeping the non-dominated solutions from $A \cup \{x'\}$. In PLS, the solutions located in the neighborhood of x are evaluated. Let $N(x)$ denote the set of solutions located at a Hamming distance 1. The non-dominated solutions from $A \cup N(x)$ are stored in the archive, and the current solution x is then tagged as *visited* in order to avoid a useless reevaluation of its neighborhood. This process is iterated until a stopping condition is satisfied. While for GSEMO there does not exist any explicit stopping rule [14], PLS has a natural stopping condition which is satisfied when all solutions from the archive are tagged as *visited*.

In other words, while PLS is based on the exploration of the whole 1-bit-flip neighborhood from x , GSEMO rather uses an ergodic operator, i.e. an independent bit-flip mutation. This means

Algorithm 1: GSEMO

```

1 Choose an initial solution  $x_0$  uniformly from  $X$ ;
2  $A \leftarrow \{x_0\}$ ;
3 repeat
4   | Select an element  $x$  out of  $A$  uniformly;
5   | Create  $x'$  by flipping each bit of  $x$  with a rate  $1/N$ ;
6   |  $A \leftarrow$  non-dominated solutions from  $A \cup \{x'\}$ ;
7 until  $success \vee maxeval$ ;
```

Algorithm 2: PLS

```

1 Choose an initial solution  $x_0$  uniformly from  $X$ ;
2  $A \leftarrow \{x_0\}$ ;
3 repeat
4   | Select a non-visited element  $x$  out of  $A$  uniformly;
5   | Create  $N(x)$  by flipping each bit of  $x$  in turns;
6   | Flag  $x$  as visited;
7   |  $A \leftarrow$  non-dominated solutions from  $A \cup N(x)$ ;
8 until  $all-visited \vee success \vee maxeval$ ;
```

that there is a non-zero probability of reaching any solution from the solution space at every GSEMO iteration. This makes GSEMO a *global* optimizer rather than a *local* optimizer as PLS. In this paper, we are interested in the running time, in terms of a number of function evaluations, until an $(1 + \varepsilon)$ -approximation of the Pareto set is identified and is contained in the internal memory A of the algorithm, subject to a maximum budget of function evaluations.

2.4 Estimated Expected Running Time (ert)

Let ε be a constant value such that $\varepsilon \geq 0$. The (multiplicative) ε -dominance relation (\preceq_ε) can be defined as follows. For $x, x' \in X$, x is ε -dominated by x' ($x \preceq_\varepsilon x'$) iff $f_i(x) \leq (1 + \varepsilon) \cdot f_i(x')$, $\forall i \in \{1, \dots, m\}$. 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. [25]. Interestingly, under some general assumptions, there always exists an $(1 + \varepsilon)$ -approximation, for any given $\varepsilon \geq 0$, whose cardinality is both polynomial in the problem size and in $\frac{1}{\varepsilon}$ [20].

Following a conventional methodology from single-objective continuous black-box optimization [3], we measure algorithm performance in the expected number of function evaluations to identify an $(1 + \varepsilon)$ -approximation. However, as any heuristic, GSEMO or PLS can either succeed or fail to reach an accuracy of ε in a single run. In case of success, we record the number of function evaluations until an $(1 + \varepsilon)$ -approximation was found. In case of failure, we simply *restart* the algorithm at random. Thus we obtain a “*simulated running time*” [3] from a set of independent trials on each instance. Such performance measure allows us to take into account both the success rate $p_s \in (0, 1]$ and the convergence speed of the algorithm with restarts. Precisely, after $(t - 1)$ failures, each one requiring T_f evaluations, and the final successful run of T_s evaluations, the total runtime is $T = \sum_{i=1}^{t-1} T_f + T_s$. By taking the expectation and by considering independent trials as a Bernoulli process stopping at the first success, we have:

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

In our case, the success rate p_s is estimated with the ratio of suc-

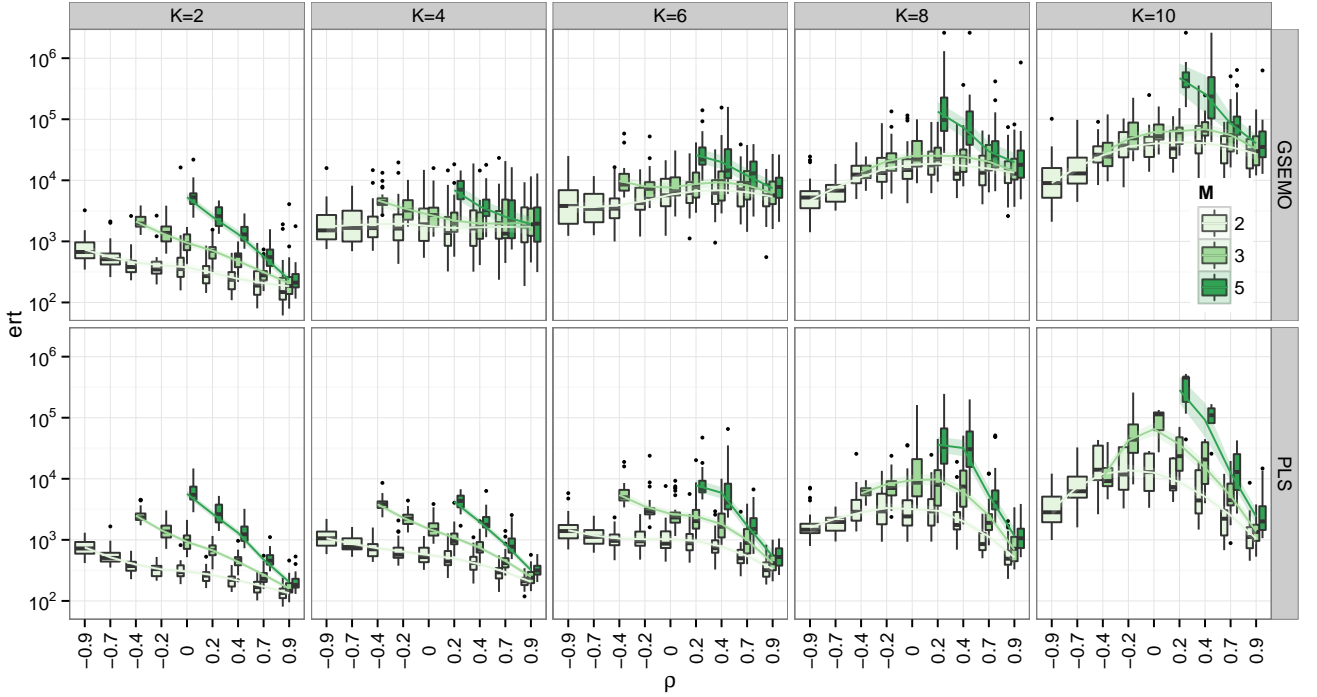


Figure 1: distribution of estimated runtime ert (y-axis, fixed log-scale) w.r.t. to objectives correlation ρ (x-axis) for both algorithms (see facets right labels). Results are grouped by problem non-linearity K (see facets top labels) and by number of objectives M (see legend). Box-and-whisker plots give median and inter-quantile range; LOESS smooth curves show trends.

successful runs over the total number of executions (\hat{p}_s), the expected running time for unsuccessful runs $\mathbb{E}[T_f]$ is set as 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:

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

where t_s is the number of successful runs, and T_i is the number of evaluations for successful run i . For more details, we refer to [3].

3. EXPERIMENTAL ANALYSIS

3.1 Experimental Setup

We consider ρ MNK-landscapes with an epistatic degree $K \in \{2, 4, 6, 8, 10\}$, an objective space dimension $M \in \{2, 3, 5\}$, and $\rho \in \{-0.9, -0.7, -0.4, -0.2, 0.0, 0.2, 0.4, 0.7, 0.9\}$, such that $\rho > \frac{-1}{M-1}$, as objective correlation. The problem size is set to $N = 18$ in order to enumerate the solution space exhaustively. A set of 30 different landscapes are independently generated at random for each parameter combination ρ , M , and K . They are made available at the following URL: <http://mocobench.sf.net>.

The target is set at $\varepsilon = 0.1$. The time limit is set to $T_{max} = 2^N \cdot 10^{-1} < 26215$ function evaluations without identifying an $(1 + \varepsilon)$ -approximation. Each algorithm is executed 100 times *per* instance. From these 100 repetitions, the success rate and the expected number of evaluations for successful runs, hence the expected runtime on the given instance, are estimated. For the comparative analysis, we only consider pairwise-complete cases, i.e. instances that have been solved by both algorithms. This brings the total number of available observations to 2874 *per* algorithm.

3.2 Exploratory Analysis

The expected runtime (ert) distribution across the experimental blocks that are defined by each combination of benchmark parameters is presented in Figure 1. For both algorithms, the ert clearly increases with the ruggedness (K) and the number of objectives (M), whereas the trend w.r.t. the objective correlation (ρ) is a bit more complex. Indeed, for a small K and a large M , the ert decreases when ρ increases. On the contrary, for large K , problem instances seem to get harder when the objectives are independent ($\rho \approx 0$) rather than anti-correlated ($\rho < 0$), despite the fact that the cardinality of the Pareto set (PS) gets higher in such cases [24]. This gives rise to the inverted u-shape observed on the right side of the figure, which is particularly pronounced for PLS. Actually, we observed that the ε -value of random approximation sets follows a similar u-shaped trend w.r.t. objective correlation. Also, this ε -value tends to increase with K . This holds for approximation sets containing a constant number of randomly generated solutions. Moreover, we know from [24] that the number of Pareto local optimal solutions increases with K and decreases with ρ . This could explain the relative advantage of PLS on problem instances with positively correlated objectives. Notice also that the opposite is true for the connectedness of the PS: the smaller K , the larger ρ , the more clustered are Pareto optimal solutions in the solution space.

Figure 2 displays the runtime aggregated over all the instance parameters (ρ , M , K) but one. We clearly see that PLS is significantly outperforming GSEMO overall, and in particular for positively correlated objectives. In fact, the runtime of PLS is shorter than that of GSEMO for 88% of the instances; compared to PLS, on average GSEMO requires more than 17 000 additional function evaluations to identify a 1.1-approximation of the PS. The performance dif-

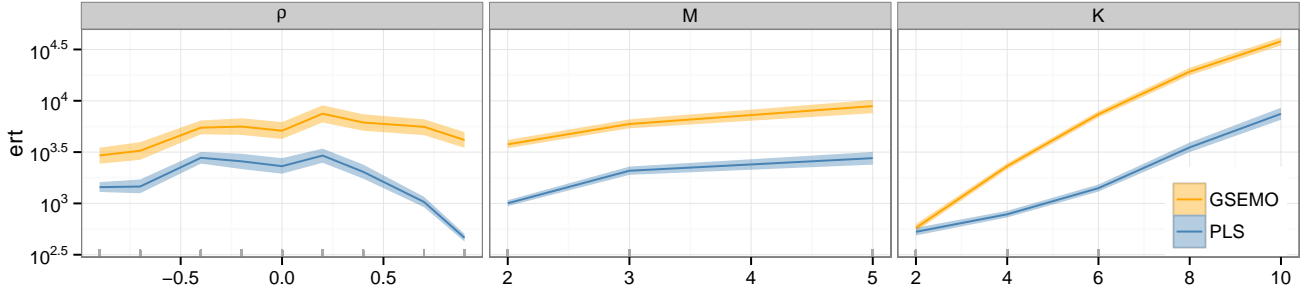


Figure 2: interaction plots between the average estimated runtime ert (y-axis, log-scale) and the benchmark parameters: objective correlation ρ (left), the number of objectives M (center), and the problem non-linearity K (right, see titles). Results are grouped by algorithm (see legend). The average (lines) and its 0.95 confidence interval (shaded areas) are evaluated through bootstrapping.

ference between the two algorithms seems to be constant, except for large ρ and w.r.t. K . Notably, the ruggedness of the underlying single-objective objective functions appears to have the highest impact on the search performance. In particular, the ruggedness seems to have more impact on the performance of GSEMO than PLS. In general, finding an $(1 + \varepsilon)$ -approximation becomes harder as the number of objectives grows and much harder for highly-rugged instances, whereas the trend w.r.t. objective correlation is less clear, more algorithm-dependent.

In the next section, we list the problem features that are *intuitively* impacting the performance of dominance-based search heuristics for the class of ρ MNK-landscapes, and we *explicitly* assess their separate and joint effect on the runtime of PLS and GSEMO.

4. FEATURE-BASED ANALYSIS

4.1 Characterizing Problem Difficulty

Besides the benchmark parameters defining ρ MNK-landscapes, a total of twelve general-purpose problem features are summarized in Table 1; they can be classified into the following three groups:

- (i) Features related to the Pareto set (PS) includes the number of non-dominated solutions, the number of supported solutions [9], and the hypervolume of the Pareto front, setting the reference point to the origin.
- (ii) Features related to the distance and the connectedness of Pareto optimal solutions includes the average and the maximum distance between any pair of solutions from the PS, the number of clusters relative to the PS size, the proportional size of the largest cluster, and the minimum distance to consider such that all solutions belong to the same cluster. The distance measure is here taken as the Hamming distance between binary strings, which is directly related to the *bit-flip* neighborhood operator.
- (iii) Features related to multimodality or ruggedness of the landscape includes the number of Pareto local optimal (PLO) solutions, the length of a single-solution Pareto hill-climbing that iteratively accepts a dominating neighbor until a PLO is found, as well as the autocorrelation of hypervolume of single solutions or of their neighborhood sets (local hypervolume), along a random walk. The random walk length is set to $\ell = 10^4$, and the neighborhood is the *1-bit-flip*.

Obviously, since some of those features require the solution space to be enumerated exhaustively, they are not practical for a performance prediction purpose. However, we decided to include them

Table 1: Summary of ρ MNK-landscape benchmark parameters and problem instance features investigated in the paper.

Benchmark parameters (3)	
K	Number of variable (epistatic) interactions
M	Number of objective functions
ρ	Correlation between the objective values
Problem instance features (12)	
<code>np0</code>	Number of solutions from the PS [1, 13]
<code>supp</code>	Proportion of supported solutions in the PS [13]
<code>hv</code>	Hypervolume [25] of the PS [1]
<code>avgd</code>	Avg. dist. between solutions from the PS [16]
<code>maxd</code>	Max. dist. between solutions from the PS [13]
<code>ncomp</code>	Relative number of clusters in the PS [23]
<code>lcomp</code>	Proportional size of the PS largest cluster [15]
<code>dconn</code>	Minimal distance to connect the PS [23]
<code>nplo</code>	Number of Pareto local optima [22]
<code>ladapt</code>	Length of adaptive walk [24]
<code>corhv</code>	Autocorrelation of solution hypervolume [16]
<code>corlhv</code>	Autocorrelation of local hypervolume [16]

in order to examine their impact on the algorithm performance. For a more comprehensive explanation of those features and their intercorrelation, we refer to [16]. In the next section, we relate the value of those features for enumerable ρ MNK-landscapes to the performance of both GSEMO and PLS.

4.2 Correlation Analysis

A first assessment of the dependency of the search performance on instance features can be done through visual inspection of scatter plots, supported by a correlation analysis. Naturally, *correlation* does not imply *causation*, and we do not draw any *direct* link between each considered feature and the algorithm runtime, even if in our case the eventual link could only go in one direction. Instead, we restrict ourself to measure the association of each feature to the performance metric (ert). We quantify this dependency strength via the Kendall's τ statistic [18]. This rank-based nonlinear correlation measure is based on the ordering of all possible pairs, and its value is proportional to the difference between the number of concordant and discordant pairs among all possible pairwise comparisons. As such, when the null hypothesis of mutual independence ($H_0 : \tau = 0$) is rejected, τ can be directly interpreted as the prob-

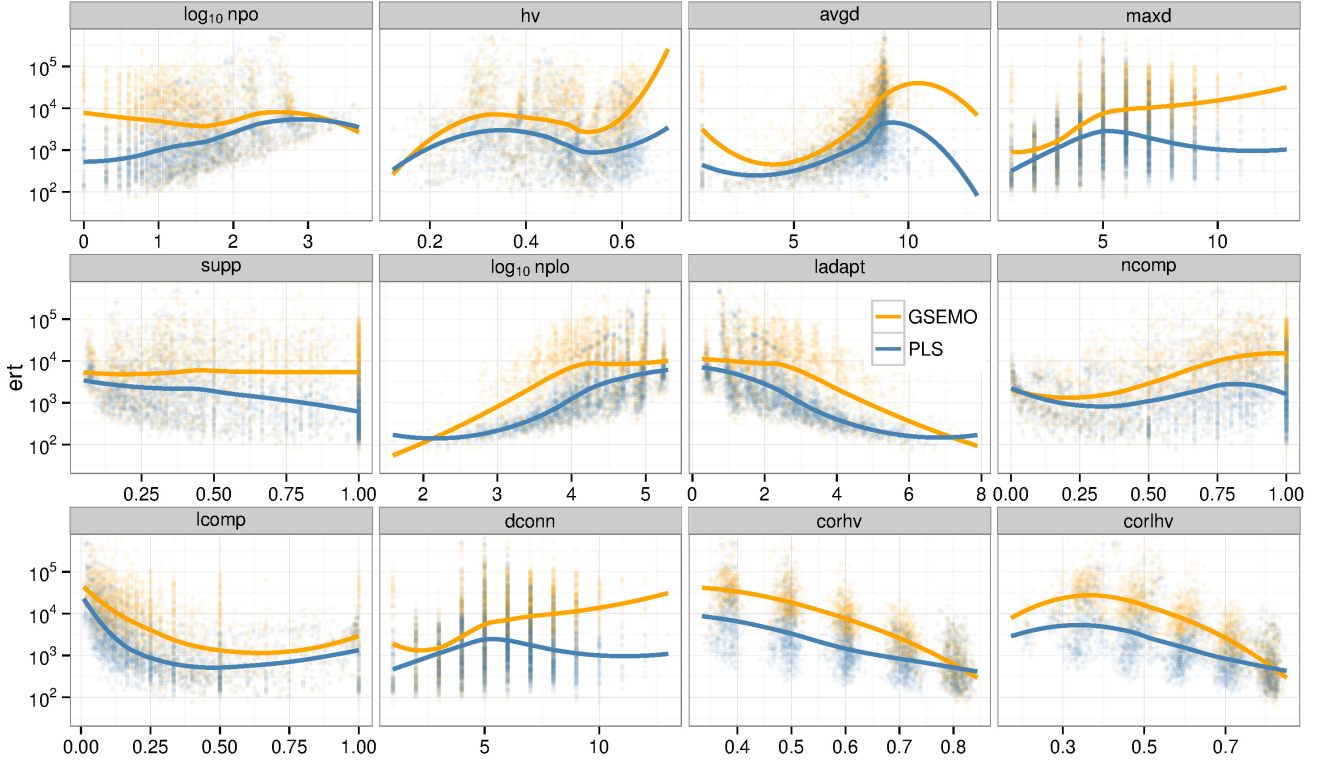


Figure 3: interaction plots between the average estimated runtime ert (y-axis, fixed logscale) and the instance features (x-axis, see facets titles). Results are grouped by algorithm (see legend). Lines represent a locally-fitted polynomial function (LOESS).

ability of observing agreement ($\tau > 0$) or disagreement ($\tau < 0$) between the ranks of paired values.

The scatter plots and the regressions (with local polynomial fitting) of the average running time of both algorithms as a function of the instance features are provided in Figure 3. In addition, Kendall’s τ coefficients are given by the red points in Figure 4. For both algorithms, the average distance between Pareto optimal solution (avgd) is highly positively correlated with ert : the larger this distance, the longer the running time. On the contrary, both ruggedness-related features based on measures of hypervolume autocorrelation ($corhv$, $corlhv$), and one feature related to the connectedness, i.e. the size of the largest cluster in the PS ($lcomp$), are highly negatively correlated. As expected, ruggedness and connectedness play a major role for both algorithms: the runtime decreases with $corhv$ and $corlhv$, and when a large number of Pareto optimal solutions are clustered in the solution space.

Some features have a different impact on the two algorithms, possibly highlighting their respective strengths and weaknesses. In particular, the runtime of PLS increases with the number of Pareto optimal solutions and with the number of PLO solutions. Contrastingly, the scatter plots show that having a high number of PLO has less impact on GSEMO than PLS. Moreover, the runtime of GSEMO is correlated with three others features related to the connectedness of the PS ($maxd$, $ncomp$, $dconn$). Indeed, the relations between Pareto optimal solutions have a large effect on the runtime of GSEMO, especially when the distance between those solutions is large. Surprisingly, the runtime of PLS does not increase when non-dominated solutions are disconnected.

However, we have to be careful when drawing conclusions by aggregating data from different areas of the instance parameters

space, since feature values and their range depend, in turn, on the levels of ρ , M , and K . This can be visually appreciated in Figure 3: the autocorrelation measures $corhv$ and $corlhv$, for example, are clearly clustered around five levels that actually correspond to the different K -values. Similarly, we are able to distinguish three clusters in the hypervolume metric hv , which actually follow the objective space dimension M .

Therefore, we deepen the analysis by evaluating the correlation *within* the instance groups defined by each possible combination of the $\langle \rho, M, K \rangle$ -values under study. Black points and lines in Figure 4 display the average τ -value within groups, together with the confidence interval associated to the mean. By comparing them with red points, we can clearly notice how data aggregation slightly enhances the correlation statistic in the $corhv$ and $corlhv$ case, leading to the same inference nonetheless. On the contrary, although hypervolume is very weakly associated with runtime overall, and that its impact is contradictory between GSEMO and PLS, group results are more consistent, showing a strong positive association between ert and hv for both algorithms. Unfortunately, as for features related to the connectedness of the PS, our confidence on the average correlation within groups is too low to make further comments, mainly due to the fact that, in many cases, we could not reject the null hypothesis of mutual independency at the group level. Nevertheless, the previous observations on data blocks and their possible effect motivate the remainder of our analysis.

4.3 Regression Analysis

In this section we aim to quantify the impact of instance features, and possibly disentangle their individual contribution to the performance variance, by taking into account the dependency among

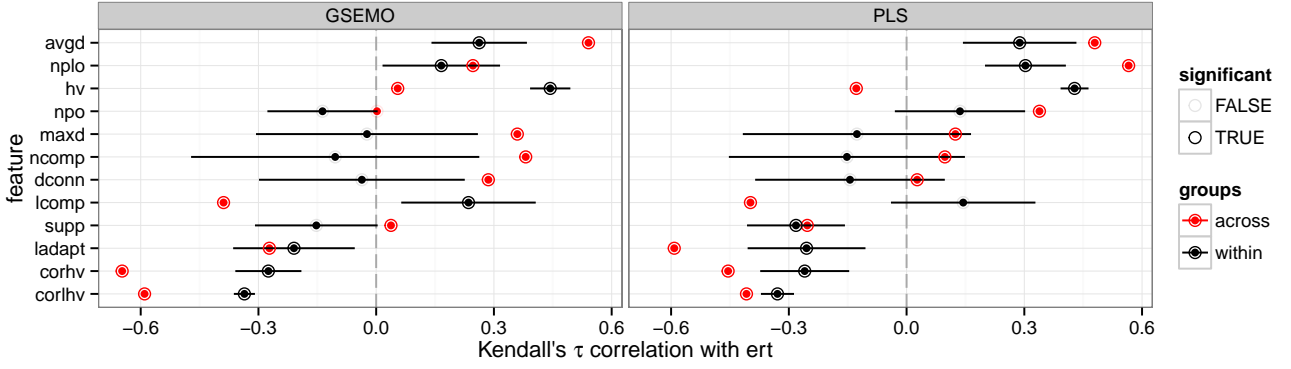


Figure 4: performance-feature association. Points give Kendall’s τ statistic for the correlation between runtime and instance feature, evaluated on the whole set of instances (red) or within instance groups (black, see left legend). Group averages, 0.95 confidence intervals, and significance, are estimated with a t-test considering only statistics that were already significant at the group level. Group and overall significance are based on Kendall’s test p -value at the 0.05 level ($H_0 : \tau_i = 0$) [18].

measurements that is induced by the experimental plan. Our goal is precisely to generalize from it as much as possible, in order to make inferences about the effect and relative importance of features.

4.3.1 Linear Mixed-Model

Let y_i be the log-transformed expected runtime (`ert`) on the i -th problem instance. We treat y_i as an observation from a random variable Y with expectation $\mathbb{E}(Y) = \mu$, i.e. $y_i = \mu + \varepsilon_i$ where ε_i are taken to be independent, identically distributed, and zero-mean.

In a classical multilinear regression, we would model μ as a linear combination of p predictors, notably (a subset of) the p problem instance features that we can measure. Thus, the performance observation on instance i can be written as:

$$y_i = \beta_0 + \sum_{k=1}^p \beta_k x_{ki} + \varepsilon_i \quad \varepsilon_i \sim \mathcal{N}(0, \sigma^2)$$

where ε_i is the usual random term, i.e. the regression residual. In this model, performance observations are supposed to be i.i.d. from a normal distribution $y_i \sim \mathcal{N}(\mu, \sigma^2)$. However, as discussed in the previous sections, our observations are mostly clustered around the different combinations of benchmark parameters; see Figure 1. In fact, a simple linear regression on a dummy categorical predictor having a different level for each combination of ρ , M , and K , would explain 84.51% and 86.85% of the `ert` variance of GSEMO and PLS, respectively.

Since we rather want to investigate the impact of instance features, we need to decompose that global performance variance into what is due to the grouping of benchmark parameters, from which we would like to generalize, and what is due to the randomness involved in the instance generation process; namely for ρ MNK-landscapes, the epistatic interaction links and their contributions to the objective values. That conveys the feature variance within the blocks of our experimental design. To this end, instead of fitting an independent regression model for each instance group, we build a linear mixed-model with random effects for experimental blocks. In such framework, the performance on instance i from group j can then be modelled as:

$$y_{ij} = \beta_0 + \sum_{k=1}^p \beta_k x_{kij} + \alpha_j + \varepsilon_{ij} \quad \varepsilon_{ij} \sim \mathcal{N}(0, \sigma^2)$$

where α_j are i.i.d random variables, with $\alpha_j \sim \mathcal{N}(0, \sigma_\alpha^2)$ denoting the group effect. By doing so, we suppose that features have

the same impact across groups; in linear terms, constant (*fixed*) slopes and *random* intercepts. Notice that estimates $\hat{\beta}$ and residuals ε will likely not be the same as in the previous model. Notice also that performance observations are now i.i.d. *conditionally* on the grouping factor $y_{ij} | \alpha_j \sim \mathcal{N}(\mu + \alpha_j, \sigma^2)$, whereas the unconditional model $y_{ij} \sim \mathcal{N}(\mu, \sigma^2 + \sigma_\alpha^2)$ carries a dependency between measurements on the same group and yields the aforementioned variance decomposition [8].

The usual approach to estimate the parameters of the unconditional model is the restricted maximum likelihood method; see [5, 10] for theoretical and implementation details. In the following, we present the results of such estimation on the full model comprising all instance features and for both algorithms. In particular, each estimated β_i is tested against the null hypothesis $H_0 : \beta_i = 0$, whereas as for the group effect we only need to check $H_0 : \sigma_\alpha^2 = 0$. In the multilinear framework, the regression coefficients that are statistically significant allow us to assess the runtime effect of each feature conditionally on all the others and, given the random effect formulation, across experimental blocks.

Finally, in order to assess the accuracy of a regression model, the conventional R^2 (ratio of variance explained by the regression) can be extended by taking into account the variance decomposition that is specific to mixed-models [19]. We obtain a *marginal* R^2 yielding the proportion of variance explained by instance features, and a *conditional* R^2 which, despite its name, gives the proportion of variance explained by the entire model, i.e. including the random effect of benchmark parameter combinations. Marginal and conditional R^2 are respectively 0.617 and 0.919 for the regression modelling GSEMO’s `ert`, respectively 0.482 and 0.911 for PLS.

4.3.2 Predictors Effect Size

We report in Figure 5 the values of the regression coefficients estimated from the mixed-model. In a multilinear regression, each coefficient β_i predicts the change in the conditional mean of the response variable after a unitary change of the corresponding covariate x_i . As such, higher values correspond, visually, to steeper slopes of the partial regression lines where the given predictor is the only covariate and all other predictors remain constant. This simple interpretation motivates the choice of a multilinear model.

For both algorithms, the features having the highest individual impact on the runtime are the hypervolume (`hv`) and its autocorrelation measures (`corhv` and `corlhv`, in order of importance).

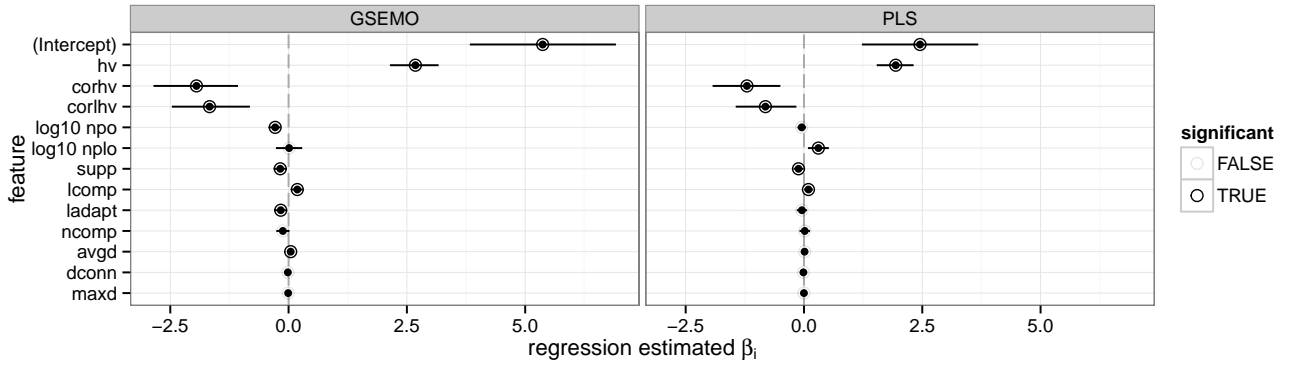


Figure 5: conditional impact of instance features on \log_{10} expected runtime in a linear mixed-model. Points and bars give, respectively, the estimates and 0.95 confidence intervals of model parameters for intercept and fixed effects β_i ($H_0 : \beta_i = 0$) [5].

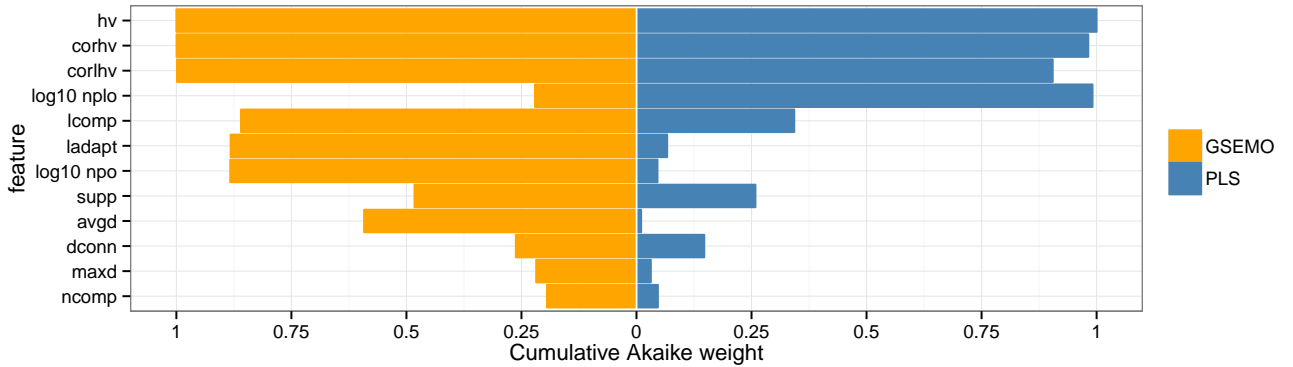


Figure 6: relative importance of instance features as performance predictors in a linear mixed-model. Each bar displays, among all 2^p possible models, the sum of the Akaike’s weights of the $2^{(p-1)}$ models including the given feature [4].

For GSEMO, the number of Pareto optimal solutions has a significant effect on estimated runtime, the larger `npo` the shorter `ert`, whereas we cannot reject the hypothesis that the number of PLO solutions has no effect on GSEMO’s `ert`. Still, the effect size of the length of an adaptive walk (`ladapt`), which is a good estimator of `nplo` [24], is significant for GSEMO. Conversely, the opposite is true for PLS. Once again, the relative size of the largest connected component of the PS (`lcomp`) impacts the performance of both algorithm. However, when controlling for all other features (i.e. conditionally on all other predictors), we find that an increase in the PS connectedness yields an increase in the expected runtime. Surprisingly also, the number of supported solutions (`supp`) is a significant predictor for both algorithms, even if none of them explicitly exploits this feature during the search process. Finally, despite being one of the features with the highest correlation to `ert`, the average distance between non-dominated solutions (`avgd`) has little impact on GSEMO and no significant impact on PLS.

4.3.3 Relative Importance of Predictors

Variable importance is commonly assessed via feature selection. However, stepwise selection can be misleading: intercorrelated predictors have a confounding effect on each other, but what the multilinear regression tries to measure is precisely the effect of one variable *controlling* for the others (i.e. fixing all the others), which leads to a poor estimation in the presence of collinearity. In an information theoretical framework, the Akaike Information Crite-

ron (AIC) [2] measures the relative quality of a model on a given dataset, not in terms of accuracy, but in terms of likelihood, i.e. the relative distance to the unknown true mechanism generating the observed data. The difference in AIC-values between two models can then be used to estimate the strength of evidence for one model against the other. On a set of alternate models, AIC differences can be transformed into so-called Akaike weights, which can be directly interpreted as the probability for each model to be the “best” one, conditionally on the considered set of models [6]. In this context, instead of performing feature selection, variable importance can be better assessed by making inference from all candidate models [7]. We perform an exhaustive search in the space of all 2^p models we can build with our p predictors. The sum of Akaike weights for the $2^{(p-1)}$ models that contain a given predictor, can give us an estimation of the relative importance of that particular variable.

Results are reported in Figure 6: `hv`, `corhv`, and `corlhv` are strong explanatory features for both models, albeit in different order of relative importance. For GSEMO, `npo`, `ladapt`, and `lcomp` can also be considered as important features, whereas `nplo` is the second most-important predictor of PLS runtime.

5. CONCLUSIONS

In this paper we investigated the impact of problem characteristics on the performance of two simple dominance-based multi-objective search heuristics. More particularly, we analyzed the expected runtime of GSEMO and PLS to identify a $(1 + \varepsilon)$ -approx-

mation of the Pareto set for enumerable ρ MNK-landscapes. Beyond a more conventional performance assessment of competing algorithms, we studied the correlations between algorithm performance and a set of relevant problem features. We contrasted their association with algorithm runtime, within and across instance classes. By building a mixed-model regression taking into account the experimental design, we were able to make more general inferences about the impact of problem features, their relative importance, as well as similarities and differences between both algorithms in this respect. Notably, the global hypervolume indicator value and its local autocorrelation measures, which are related to the ruggedness of the fitness landscape, have the strongest impact for both algorithms. In addition, as expected, our analysis reveals that the runtime of PLS appears to be largely affected by the landscape multimodality, here defined as the number of Pareto local optimal solutions. Moreover, the Pareto set cardinality plays a more important role for GSEMO than for PLS. At last, the features reflecting the connectedness of the Pareto set are important predictors for the runtime of GSEMO, which navigates the search space with an ergodic variation operator. On the contrary, the same features have a non-significant effect on the performance of PLS, which is constrained by the exploration of a finite neighborhood.

A similar analysis on other problem classes and large-size instances, with computationally affordable features, would allow us to further understand and gain more knowledge about the main difficulties that EMO algorithms have to face, and eventually to predict their expected performance.

Acknowledgments

This work was partially supported by the Japanese-French JSPS project “Global Research on the Framework of Evolutionary Solution Search to Accelerate Innovation” (2013-2016), and by the Japanese-French JSPS-Inria project “Threefold Scalability in Any-objective Black-Box Optimization” (2015-2017).

6. REFERENCES

- [1] H. E. Aguirre and K. Tanaka. Working principles, behavior, and performance of MOEAs on MNK-landscapes. *Eur J Oper Res*, 181(3):1670–1690, 2007.
- [2] H. Akaike. Information theory as an extension of the maximum likelihood principle. In *Second International Symposium on Information Theory*, pages 267–281. Akademinai Kiado, 1973.
- [3] A. Auger and N. Hansen. Performance evaluation of an advanced local search evolutionary algorithm. In *Evolutionary Computation, 2005. The 2005 IEEE Congress on*, volume 2, pages 1777–1784. IEEE, 2005.
- [4] K. Bartoń. *MuMIn: Multi-Model Inference*, 2014. R package version 1.12.1.
- [5] D. Bates, M. Mächler, B. Bolker, and S. Walker. Fitting linear mixed-effects models using lme4. *arXiv preprint arXiv:1406.5823*, 2014.
- [6] K. P. Burnham and D. R. Anderson. *Model selection and multimodel inference: a practical information-theoretic approach*. Springer Science & Business Media, 2002.
- [7] K. P. Burnham and D. R. Anderson. Multimodel inference understanding aic and bic in model selection. *Sociological methods & research*, 33(2):261–304, 2004.
- [8] M. Chiarandini and Y. Goegebeur. Mixed models for the analysis of optimization algorithms. In *Experimental Methods for the Analysis of Optimization Algorithms*, volume 1, page 225. Springer, 2010.
- [9] M. Ehrgott. *Multicriteria optimization*. Springer, 2005.
- [10] J. J. Faraway. *Extending the linear model with R: generalized linear, mixed effects and nonparametric regression models*. Texts in Statistical Science. Chapman & Hall/CRC, 2005.
- [11] F. Hutter, L. Xu, H. H. Hoos, and K. Leyton-Brown. Algorithm runtime prediction: Methods & evaluation. *Artificial Intelligence*, 206:79–111, 2014.
- [12] S. A. Kauffman. *The Origins of Order*. Oxford University Press, 1993.
- [13] J. Knowles and D. Corne. Instance generators and test suites for the multiobjective quadratic assignment problem. In *Evolutionary Multi-Criterion Optimization (EMO 2003)*, volume 2632 of *LNCS*, pages 295–310. Springer, 2003.
- [14] M. Laumanns, L. Thiele, and E. Zitzler. Running time analysis of evolutionary algorithms on a simplified multiobjective knapsack problem. *Nat Comput*, 3(1):37–51, 2004.
- [15] A. Liefvooghe, L. Paquete, and J. R. Figueira. On local search for bi-objective knapsack problems. *Evol Comput*, 21(1):179–196, 2013.
- [16] A. Liefvooghe, S. Verel, H. Aguirre, and K. Tanaka. What makes an instance difficult for black-box 0–1 evolutionary multiobjective optimizers? In *International Conference on Artificial Evolution (EA 2013)*, volume 8752 of *LNCS*, pages 3–15, 2013.
- [17] A. Liefvooghe, S. Verel, F. Daolio, H. Aguirre, and K. Tanaka. A feature-based performance analysis in evolutionary multiobjective optimization. In *Evolutionary Multi-Criterion Optimization*, volume 9019 of *LNCS*, pages 95–109. Springer, 2015.
- [18] A. McLeod. *Kendall: Kendall rank correlation and Mann-Kendall trend test*, 2011. R package version 2.2.
- [19] S. Nakagawa and H. Schielzeth. A general and simple method for obtaining r^2 from generalized linear mixed-effects models. *Methods in Ecology and Evolution*, 4(2):133–142, 2013.
- [20] C. H. Papadimitriou and M. Yannakakis. On the approximability of trade-offs and optimal access of web sources. In *Symposium on Foundations of Computer Science (FOCS 2000)*, pages 86–92, 2000.
- [21] L. Paquete, M. Chiarandini, and T. Stützle. Pareto local optimum sets in the biobjective traveling salesman problem: An experimental study. In *Metaheuristics for Multiobjective Optimisation*, volume 535 of *Lecture Notes in Economics and Mathematical Systems*, chapter 7, pages 177–199. Springer, 2004.
- [22] L. Paquete, T. Schiavinotto, and T. Stützle. On local optima in multiobjective combinatorial optimization problems. *Ann Oper Res*, 156(1):83–97, 2007.
- [23] L. Paquete and T. Stützle. Clusters of non-dominated solutions in multiobjective combinatorial optimization: An experimental analysis. In *Multiobjective Programming and Goal Programming*, volume 618 of *LNMES*, pages 69–77. Springer, 2009.
- [24] S. Verel, A. Liefvooghe, L. Jourdan, and C. Dhaenens. On the structure of multiobjective combinatorial search space: MNK-landscapes with correlated objectives. *Eur J Oper Res*, 227(2):331–342, 2013.
- [25] E. Zitzler, L. Thiele, M. Laumanns, C. M. Fonseca, and V. Grunert da Fonseca. Performance assessment of multiobjective optimizers: An analysis and review. *IEEE Trans. Evol. Comput.*, 7(2):117–132, 2003.