Evolutionary Computation Just Accepted MS.
doi:10.1162/EVCO_a_00177

Identifying Features of Fitness Landscapes and Relating Them to Problem Difficulty

I. Moser imoser@swin.edu.au
Department of Computer Science and Software Engineering, Swinburne University of Technology, Melbourne, Victoria, Australia

M. Gheorghita mgheorghita@swin.edu.au
Department of Computer Science and Software Engineering, Swinburne University of Technology, Melbourne, Victoria, Australia

A. Aleti aldeida.aleti@monash.edu
Faculty of Information Technology, Monash University, Melbourne, Victoria, Australia

Abstract

Complex combinatorial problems are most often optimised with heuristic solvers which usually deliver acceptable results without any indication of the quality obtained. Recently, Predictive Diagnostic Optimisation was proposed as a means of characterising the fitness landscape while optimising a combinatorial problem. The scalars produced by Predictive Diagnostic Optimisation appeared to describe the difficulty of the problem with relative reliability. In this study, we record more scalars that may be helpful in determining problem difficulty during the optimisation process and analyse these in combination with other well-known landscape descriptors using exploratory factor analysis on four landscapes that arise from different search operators, applied to a varied set of quadratic assignment problem instances. Factors are designed to capture properties by combining the collinear variances of several variables. The extracted factors can be interpreted as features of landscapes detected by the variables, but disappoint in their weak correlations with the result quality achieved by the optimiser, which we regard as the most reliable indicator of difficulty available. It appears that only the prediction error of Predictive Diagnostic Optimisation has a strong correlation with the quality of the results produced, followed by a medium correlation of the fitness distance correlation of the local optima.

Keywords

Search space characterisation, fitness landscapes, local search, combinatorial optimisation.

1 Introduction

Combinatorially complex problems are often optimised using stochastic approaches such as evolutionary algorithms (EA), which improve a number of candidate solutions iteratively by making changes to them. The type and degree of change applied is defined by the *search operators* used, which is partly prescribed by the solution representation (encoding) and partly devised empirically by researchers with experience of the type of problem at hand.

The operator gives rise to a *neighbourhood* of a solution, which is the set of all solutions obtained by applying the operator to the original solution in all possible ways. The fitnesses of the solutions which are created through repeated application of the operator can be considered as a landscape with varying elevation.

For a stochastic solver to outperform random search, it is necessary for the search landscape to contain gradients - areas where repeated application of the search operator leads to gradually improving fitness. Researchers and practitioners intuitively shape the search landscape of an optimisation problem by trialling candidate encodings and operators and adopting the most successful ones that lead to the best search result.

The fitness landscapes of complex problems are often *multimodal*, containing many ‘hills’ or ‘basins’. The landscape features are often described in mixed terms derived partly from maximisation (‘peaks’,

Evolutionary Computation Just Accepted MS.
doi:10.1162/EVCO_a_00177

'hill climbing') and partly from minimisation ('basins of attraction', 'steepest descent'). In this work, we will use the terminology of minimisation.

It is not clear what exactly makes a fitness landscape hard for a stochastic search algorithm and how information regarding the fitness landscape can be used to improve performance. It has been argued that the number, distribution and shapes of the modes in a multimodal landscape influence its difficulty (Garnier and Kallel, 2002), which suggests that the choice of solution representation and the search operator can make a problem easier or harder to solve, a notion which has frequently been observed in practice, e.g. Moser and Mostaghim (2010).

Local Search (LS) is a successful class of approximation algorithms which has been shown to achieve near-optimal solutions to many difficult problems when coupled with a suitable global search (Stützle, 1999). Fitness landscape characterisation, an area of research dedicated to the investigation of search landscape topology and difficulty, often uses small LS steps to extract features of the fitness landscape. Operators which cause minimal change evoke more detailed landscapes, often with equally sized neighbourhoods. Paradigms like Local Optima Networks (Daolio et al., 2011) and Autocorrelation (Weinberger, 1990) are based on LS. Predictive Diagnostic Optimisation (PDO) (Moser and Gheorghita, 2012) used the fitness improvement ensuing from the first steepest descent (SD) step to predict the quality of the local optimum.

The ultimate goal of fitness landscape characterisation is to determine which algorithm is best suited for solving a problem. However, fitness landscape metrics are rarely used for the algorithm selection problem. A study by Smith-Miles (2008) appears to be the only application produced at this stage.

Fitness landscapes and their possible shapes and features are described in section 2. In section 3, we provide an overview of fitness landscape characterisation approaches and the scalar descriptors of problem difficulty that have arisen from some of them. In section 5.3, we introduce a number of QAP instances that were chosen, respectively created, with a view to varying the properties that influence their difficulty for a heuristic solver - the variance, skewness and sparsity of the flow and distance matrices. Four search landscapes were created on these instances by applying four search operators which are explained in section 5.4.

In the experimental section (5), we apply PDO to the QAP instances in repeat trials to collect values on scalars identified in section 2 as well as some other values that can easily be applied during an LS-based PDO search, such as the variance in the fitnesses of local optima.

The goals of this study are multifaceted. Firstly, the usefulness of the existing and proposed descriptors in diagnosing landscape difficulty is evaluated through exploratory factor analysis. Factor analysis investigates collinearity among variables to extract common underlying properties in the data. This may help in determining the exact properties of the search landscapes the descriptors are indicative of. In section 6 we analyse the landscape properties, the variables (descriptors) that contribute to characterising them and their correlations with the actual problem difficulty, which, in our assumptions, is reflected in the solution quality the search produces.

2 Features of Fitness Landscapes

The notion of *fitness landscapes* was first introduced by biologist Wright (1932) as a means of visualising distributions of fitness values of gene combinations. In the domain of optimisation, the purpose of characterising landscapes is to relate the topological description of the search space with the dynamics of the search algorithms, and thus extract information about the difficulty of a given optimisation problem. Formally, a *fitness landscape* (Radcliffe and Surry, 1995; Stadler, 1995, 2002) is represented by a triplet {S,N,F} where:

- S is the set of potential solutions for the given problem, also known as the *search space*.
- $N : S \rightarrow S$ is the neighbourhood relation, which assigns each solution $s \in S$ to a set of neighbours $N(s) \subset S$ by means of a search operator. Two solutions are neighbours if by applying a search operator to one solution it is possible to reach the other.
- $F : S \rightarrow \Re$ is the fitness function (also known as objective function) that assigns a real value to each solution, which represents the quality of the solution to be minimised or maximised.

Evolutionary Computation Just Accepted MS.
doi:10.1162/EVCO_a_00177

As the neighbourhood of a solution depends on the operator applied in optimisation, a given problem can have any number of fitness landscapes. For the purpose of describing the fitness landscape, it is intuitive, and therefore common, to use local operators which make small changes to the solution. The implicit aim is to create landscapes where the distance between a solution and its neighbour is shorter than that between the solution and the neighbour's neighbour.

The topology of a fitness landscape is composed of certain *structures*, such as *local optima*, *global optimum* and *plateaus*. In a minimisation problem, given the search space S and a neighbourhood relation N, a *local optimum* is a point $s^l \in S$ such that for any solution $s^n \in N$, $F(s^l) <= F(s^n)$. A *global optimum* is a point $s^g \in S$ that is both locally optimal and $\forall s \in S, F(s^g) < F(s)$. A *plateau* is defined as a set $P \subseteq S$ such that $\forall x s^p \in P, F(s^p) = a$, where a is a constant. If for some $s^p \in P$ there exists a solution $s^e \in N(s^p)$ such that $F(s^e) < a$, the plateau is defined as a *bench* (Watson, 2010), and all solutions s^e are called *exits*. *Exit* points the only difference between a *bench*, which is a plateau that leads to a better region, and a *local plateau*, one that is not directly connected to a better region of the search space.

A plateau indicates the presence of neutrality in the landscape, where the progress of a gradient-based search algorithm potentially stagnates. Counteracting such stagnation requires special measures which have been the focus of much research, e.g. by Barnett (2001).

A *basin of attraction* B of a local optimum s^l is defined by Garnier and Kallel (2001) as a set of points $s_1 .... s_k$ of the search space, such that an SD algorithm starting at $s_i, (1 \leq i \leq k)$ ends at the local optimum s^l in a finite number of steps.

2.1 Modality

The degree of difficulty of a fitness landscape depends largely on the number of local optima, known as *modality*, and their distribution and as well as the surrounding basins of attraction (Garnier and Kallel, 2002). A basin of attraction is the space that 'surrounds' an optimum so that once in the centre, any gradient descent leads to the associated optimum. The number and length of the gradients in a multimodal landscape largely determine how difficult it is for a gradient-based search algorithm to find the global optimum among local 'basins of attraction'.

Horn and Goldberg (1994) discussed the relationship between modality and problem difficulty at length. Unimodal problems can be difficult if large parts of the landscape are flat, and highly multimodal landscapes can be easy to solve both for hill climbers and EAs if the modes themselves 'lean' towards the global optimum. Garnier and Kallel (2002) pointed out that the complexity of a fitness landscape for a search method is not only related to the number of local optima, but to the distribution of the optima and the topology of the areas between them.

2.2 Ruggedness

The property of ruggedness is considered crucial for a problem's difficulty. The more rugged a fitness landscape, the shorter, more numerous and more heterogeneous the gradients to the local optima. In a rugged landscape, the fitnesses of neighbouring solutions are less correlated and thus, it is harder for a search method to infer a search direction from previous solution quality. When the landscape is smoother and the correlation between the fitnesses of neighbouring solutions is high, there are longer gradients for the solver to follow: The algorithm can 'assume' that a fitter solution will lead to an even fitter neighbour.

Ruggedness is related to the modality of the problem as a small correlation length indicates little correlation between neighbouring solutions, which suggests short gradients, which, in turn, suggests numerous local optima. An optimisation problem is considered easier to solve when the correlation length is high, which means that highly correlated parts of the landscape form easy-to-follow gradients to the optima (Stadler and Schnabl, 1992).

3 Measures for Characterising Fitness Landscapes

A great number of approaches to characterising fitness landscapes exist. For comprehensive surveys see Malan and Engelbrecht (2013); Pitzer and Affenzeller (2012). Various measures for analysing features of fitness landscapes have been introduced. Some measures are used to adapt the search strategy

Evolutionary Computation Just Accepted MS.
doi:10.1162/EVCO_a_00177

on the fly, others are employed to explain the behaviour of the algorithm after the optimisation process, whereas a third class of measures are used to classify different problems into different complexity classes (Jansen, 2001). The common objective is to describe the features of a fitness landscape with a single numerical value. Each technique measures a specific feature, such as ruggedness or modality, hence for a comprehensive characterisation of fitness landscapes more than one measure is required. The following sections introduce a set of measures for charactering different features of fitness landscapes.

3.1 Autocorrelation

Weinberger proposed two metrics to estimate the ruggedness of a fitness landscape: the autocorrelation function and the correlation length (Weinberger, 1990, 1991). The autocorrelation function $\rho(n)$ measures the relationship between the solutions of a random walk and the quality of these solutions. Given a series of fitness values $(f(s_t), ..., f(s_{t+n}))$ of a set of random solutions $(s_t, ..., s_{t+n})$ obtained from a random walk, the autocorrelation function is defined as:

$$\rho(n) = \frac{E[f(s_t)f(s_{t+n})] - E[f(s_t)]E[f(s_{t+n})]}{var(f(s_t))} \tag{1}$$

where $E[f(s_t)]$ is the expected value of $f(s_t)$, and $var(f(s_t))$ is the variance of $f(s_t)$.

Assuming a statistically isotropic landscape where, on average, the fitness landscape has the same topological features in all directions, this measure can be used as an indicator of difficulty for certain problems (Manderick et al., 1991; Merz and Freisleben, 2000a).

The autocorrelation length $\tau = \frac{1}{\rho(1)}$ (Fontana et al., 1993; Stadler, 1996) measures the decay of the autocorrelation function and the ruggedness of the fitness landscape, where values closer to 1 indicate a stronger correlation and values toward zero indicate the absence of correlation. Angel and Zissimopoulos (1998) observed a strong connection between correlation length and the hardness of an optimisation problem for a local search algorithm.

Angel and Zissimopoulos (2001) defined the Autocorrelation Coefficient (AC) for QAP based on the the autocorrelation $\rho(1)$ as follows:

$$\xi = \frac{1}{1 - \rho(1)} \tag{2}$$

Based on the AC ξ and the problem cardinality c, they introduced a measure of problem ruggedness denoted ζ.

$$\zeta = 100 - 400/c(\xi - c/4) \tag{3}$$

The ruggedness measure ζ (Angel and Zissimopoulos, 2001) takes values between 0 and 100, with higher values denoting a weaker autocorrelation and thus a more rugged landscape. They find that the performance of a Simulated Annealing algorithm declines monotonically with rising ζ.

While $\rho(n)$ is usually approximated by applying random walks of length n, Chicano et al. (2012) calculated the AC for QAP exactly based on the matrices. They confirm the inverse relationship between autocorrelation length and number of local optima (induced by shorter gradients) for QAP. They also pointed out that since the autocorrelation length generally increases with the size of the problem, a comparison is only meaningful when the problem instances are of identical size. Although the autocorrelation metrics have most often larger values when the problem instances are larger, this does not mean that larger instances are easier to solve. Due to similar observations, Tayarani-N. and Prügel-Bennett (2014) also opine that the importance of autocorrelation as a diagnostic of problem hardness is often exaggerated.

3.2 Fitness Distance Correlation

Fitness Distance Correlation (FDC) (Jones and Forrest, 1995b) was first conceived with the aim of studying the performance of Genetic Algorithms (GA) against different fitness functions. It measures the correlation between the fitness of a solution and distance to the optimal solution (Jones and Forrest, 1995a). While ruggedness and local optima analysis provide insights into the connectedness and local

Evolutionary Computation Just Accepted MS.
doi:10.1162/EVCO_a_00177

structure of fitness landscapes, FDC was designed to obtain a view about the global structure of the landscape. In the original definition of FDC, the knowledge of the global optimum is a prerequisite.

Given a set of n solutions $(s_1, ..., s_n)$, their corresponding fitnesses $(f(s_1), ..., f(s_n))$, and the set of distances of these solutions to the nearest global optimum $(d(s_1, s^g), ..., d(s_n, s^g))$, the correlation coefficient is computed as follows:

$$r = \frac{\sum_{i=1}^{n}(f(s_i) - \overline{f})(d(s_i, s^g) - \overline{d})}{n\sigma_f \sigma_d} \tag{4}$$

where $\overline{f}$ and $\overline{d}$ are the average fitness and the average distance to the global optimum, and σ_f and σ_d are the standard deviations of fitnesses and distances. Values close to unity are indicative of a positive correlation, i.e. the fitnesses of the solutions increase with increasing proximity to the global optimum. Positive correlation is favourable for minimisation problems, whereas negative correlation (values close to -1) indicates an easy maximisation problem and a hard minimisation problem. Correlation values around 0.0 are considered difficult problems with no correlation between solutions and global optima.

The authors (Jones and Forrest, 1995b) defined three landscapes identified by FDC:

- *Misleading or deceptive fitness landscapes:* The correlation coefficient r has a value greater than or equal to 0.15. In these problems, the population of a GA tends to diverge from the optimum.
- *Difficult fitness landscapes:* The value of correlation coefficient r is between 0.15 and -0.15. In these problems there is little correlation between fitness and distance to the optimum.
- *Straightforward fitness landscapes:* This type of landscapes are considered easy to search since the fitness tends to grow as we approach the optimum. The calculated correlation coefficient r is less than or equal to -0.15.

A weakness of this method is that the global optimum must be known a priori, which is not the case in newly arising problems for which solutions are sought. Its informative value has been the subject of some debate (Schiavinotto and Stützle, 2007; Altenberg, 1997). Authors usually agree on the FDC being a useful measure of problem difficulty for a particular algorithm when the distance metric is aligned with the search step of the algorithm (Jones and Forrest, 1995a; Altenberg, 1997). Despite good initial results on the relationship between FDC values and GA performance, subsequent studies have pointed out that FDC is not a reliable measure (Altenberg, 1997; Quick et al., 1998; Kallel et al., 1999). Specifically, Altenberg (1997) presented a problem diagnosed by the FDC as difficult which was solved easily by a GA. Notwithstanding the substantial criticism, FDC has frequently been used as a diagnistic of problem difficulty (Merz and Freisleben, 2000b; Vassilev et al., 2000; Czech, 2008; Merz, 2004).

3.3 Matrix Dominance of QAP

Following a definition by Vollmann and Buffa (1966) which assumes the variation of the flow matrix to be the deciding factor in the difficulty of a QAP problem, Merz and Freisleben (2000a) proposed the variation coefficient of both matrices A and B, multiplied by 100, as an expression of the difficulty of QAP problems. As both matrices influence the fitness difference of two neighbouring QAP solutions, Merz and Freisleben suggested that a vector containing the coefficients of both matrices should be used as a difficulty measure, which they named the dominance of a QAP instance.

$$dom(X) = 100\frac{\sigma(X)}{\overline{X}} \tag{5}$$

The dominance of the matrices A and B are calculated according to Eq. 5 - the equivalent of the matrix's variation coefficient multiplied by 100.

3.4 Negative Slope Coefficient

Vanneschi et al. (2006) proposed the negative slope coefficient (NSC), designed to define the degree of difficulty a problem poses to a GA, also termed evolvability. It is based on the fitness difference between parent and offspring solutions. Given the existence of n parent solutions x_i, it applies a neighbourhood operator to obtain the child solution v_i. Having created n child solutions, the fitnesses $f(x_i), f(v_i)$ are combined in a vector as x, y values. The x-axis defined by the parents' fitness values is subdivided into k intervals of equal length.

Evolutionary Computation Just Accepted MS.
doi:10.1162/EVCO_a_00177

Each interval forms a bin $C_a \in \{C_0...C_k\}$ which contains the tuples whose parent fitnesses $f(x_i)$ are within the range of the interval. The means of the parent and child fitnesses $\overline{f_a}(x_i)$ and $\overline{f_a}(v_i)$ in C_a are calculated. The relative fitness differences between the means of the successive intervals $\{C_0...C_k\}$ are calculated according to Eq. 6.

$$S_a = \frac{\overline{f_{a+1}}(v) - \overline{f_a}(v)}{\overline{f_{a+1}}(x) - \overline{f_a}(x)} \tag{6}$$

As the intervals are naturally ordered by the fitnesses of the initial solutions such that $\overline{f_a}(x) < \overline{f_{a+1}}(x)$, the denominator is always positive and the outcome depends on the difference of the average offspring fitnesses f', which are summed to the NSC as shown in Eq. 7.

$$NSC = \sum_{a=0}^{k-1} min(0, S_a) \tag{7}$$

The NSC considers only negative values, which means that if better solutions produce better offspring than worse solutions, the NSC is zero and the fitness landscape is considered easy for a GA, whereas if the reverse is the case, the landscape is difficult. This is intuitive as the selection operator preserves better solutions as parents for the next generation. If better parents actually produce worse offspring, the GA is unlikely to be successful at finding optimal results.

4 Predictive Diagnostic Optimisation

Predictive Diagnostic Optimisation (PDO) (Moser and Gheorghita, 2012) projects the quality of a solution to be expected after applying SD. The algorithm optimises randomly created solutions locally and calculates the ratio of the initial and final fitnesses. This ratio is later applied to similar solutions to project the fitnesses to expect at the end of SD. To identify similar solutions, the ratio between the original fitness and the fitness after the first LS move is also recorded in the predictor. Once predictors are available, new candidate solutions for local search can be matched to them after the solution has been locally improved once.

4.1 Predictor Definition

LS as illustrated in Fig. 1 is performed starting from an initial random solution s_i. Assuming a minimisation problem, after the *first step* of the LS, the ratio of fitness improvement between the initial solution s_i and the improved solution s'_i is calculated according to Eq. 8.

$$p'_i = \frac{(f(s_i) - f(s'_i))}{f(s_i)} \tag{8}$$

The 'first ratio' of p'_i sets this fitness gain in proportion to the initial fitness of the solution before the move. The 'altitude' of the current solution in the landscape naturally influences its potential for further improvement. Including the 'altitude' in p'_i is assumed to render the predictor suitable for solutions which are at an entirely different 'altitude' but experience a similar relative improvement in the 'first step'.

Once the local optimisation of the solution is complete and no further improvement is possible, the final fitness of the local optimum is used to calculate the ratio between the initial improvement of the first step and the fitness of the local optimum $f(s''_i)$ as shown in Eq. 9.

$$p''_i = \frac{f(s_i) - f(s''_i)}{f(s_i) - f(s'_i)} \tag{9}$$

Both ratios form part of the predictor. When considering a new starting solution j for local optimisation, this solution is evaluated according to the objective function, which yields the value $f(s_j)$. After the first local search step $f(s'_j)$ is obtained, the first ratio is calculated and compared to the first ratio of each available predictor. The closest matching predictor is used to project the expected final fitness $f_e(s''_j)$ using Eq. 10.

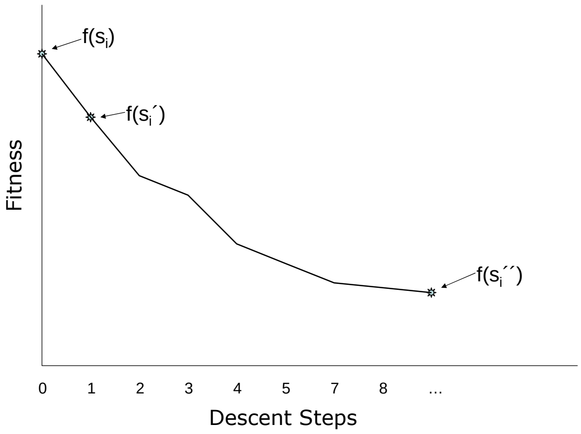

Figure 1: Development of the fitness of a solution during local optimisation over an unknown number of improvement steps. The initial fitness $f(s_i)$, the fitness after the first improvement step$f(s_i')$ and the final fitness $f(s_i'')$ at the local optimum are used for the prediction.

$$f_e(s_j'') = p_i'' * (f(s_j) - f(s_j')) + f(s_j) \tag{10}$$

As the projection of the expected fitness $f_e(s_j'')$ only uses the fitnesses at distinct points in the solution's descent towards the optimum, the number of improvement steps will vary depending on how quickly the 'bottom' of the basin is reached.

4.2 Predictor Discovery

The predictors are created in an initial discovery phase with the goal of finding all predictors necessary to describe all optima encountered in the problem within a tolerance of error. A new predictor is created during this phase whenever the existing predictors do not describe a new local optima with the expected accuracy. The discovery phase ends when no new predictors have been created in a predefined number of iterations.

To test whether a new predictor is needed, a new solution s_i is created uniformly randomly and improved through s_i' to s_i''. The ratios p_i' and p_i'' are calculated according to Eq. 8 and 9. PDO then determines if the new predictor is significantly different from the predictors created in the previous iterations. The similarity between two predictors is established using the Canberra distance (Eq. 11), a numerical measure of the distance between pairs of points in a vector space calculated as follows (Lance and Williams, 1966):

$$C(p_i, p_j) = \frac{|p_i' - p_j'|}{|p_i'| + |p_j'|} + \frac{|p_i'' - p_j''|}{|p_i''| + |p_j''|} \tag{11}$$

where $p_i = (p_i', p_i'')$ and $p_j = (p_j', p_j'')$ are the predictors with their respective ratios of fitness gain. The measure is defined in the range [0,2), with greater values representing more dissimilar predictors. If the distance of the new predictor to all other predictors in the pool is greater than a predefined threshold of $\epsilon = 0.1$, the predictor is added to the pool. The predictor creation phase stops if no new predictors are accepted during 100 iterations.

4.3 Neighbourhood Sampling

To reduce the number of function evaluations required for the local search during the exploration of the neighbourhood, a sampling procedure (Gheorghita et al., 2013) has been devised for the neighbourhood search that precedes each local search move. SD always applies the perturbation with the largest fitness gain. A representative sample is determined by a statistical significance test which achieves a defined

Evolutionary Computation Just Accepted MS.
doi:10.1162/EVCO_a_00177

confidence level by monitoring the accuracy of the sample. The average of the best fitnesses $\overline{s*}$ and the mean square of those best fitnesses $\overline{s*^2}$ is computed for each new sample. The sample size k is increased until the predefined significance level is reached.

$$e = \frac{2Z}{\sqrt{k}} \frac{\sqrt{\overline{f^2}(s*) - \overline{f}^2(s*)}}{\overline{f}s*} \tag{12}$$

where Z is the confidence interval. A higher confidence interval allows for more samples, giving more chances to discover better fitnesses, at the expense of a larger number of function evaluations. In consequence, the LS strategy employed by PDO is an approximation of SD.

4.4 Predictor Application

In the predictor application phase, random candidate solutions are created and improved locally once. The improvement achieved by this first step is used to identify the closest-matching predictor in each local search group by its ratio p_i'. When the local search of the solution is complete, the predictions made by the representative predictors are used to determine the prediction error (PE) as the difference between actual and expected fitness. The preditor count (PC) is representative of the number of basin shapes that are different between a predefined tolerance defined by the Canberra distance. The predictor diversity (PD) is calculated as the variance in Canberra distances between the predictors. As similar basins are represented by a single predictor, this is not the same measure as the fitness variance of the local optima.

5 Experimental Investigation

5.1 Descriptors

The following descriptors were used to characterise the fitness landscape of an optimisation problem instance.

Descriptors 1 to 3 are indicators specific to the PDO methods, although descriptor 3 has not been proposed earlier, and collected separately for each operator (section 5.4). Metrics 4 to 6 can be produced by any algorithm that includes a deterministic LS method. Descriptors 7 to 15 are established measures of search space difficulty introduced in section 3, among which 14 and 15 are specific to the QAP. Item 16 merely combines metrics 14 and 15.

5.2 Quadratic Assignment

The purpose of the experiments is to investigate how well selected descriptors describe the difficulty of problem instances. The descriptors can be evaluated most reliably when using problems whose characteristics are already known to a large extent and whose 'landscape difficulty' conveniently varies to allow for the exploration of different landscape features. The Quadratic Assignment Problem (Koopmans and Beckmann, 1955) (QAP) is a well-known generic problem whose objective function depends on the flow and distance matrices. The search landscape of the QAP can be influenced by manipulating these matrices.

As in this work the QAP is used as a representative of combinatorial problems in general, its specific properties have to be discussed as limitations to the generalisability of the results. Fortunately, the QAP has been studied extensively as a formulation of many practical problems. Tayarani-N. and Prügel-Bennett (2014) noted that QAPs rarely have more than one global optimum and that global optima tend to reside in areas of good fitness, although the second-best optimum is occasionally found at a large distance from the best. In any case, the areas of good fitness around the global optimum are generally too large to search exhaustively.

The QAP is the problem of allocating a set of facilities to a set of locations, with a cost function associated to the distance and flow between the facilities. We are given two n x n input matrices with real elements $H = [h_{ij}]$ and $D = [d_{kl}]$, where h_{ij} is the flow between facility i and facility j and d_{kl} is the distance between location k and location l. Given n facilities, the solution of the QAP is codified as a permutation $\sigma = (x_1, ..., x_n)$ where each x_i (i = 1, . . . , n) represents the facility that is allocated to the i-th location. The fitness of the permutation is given by the following objective function:

1.	Predictor error	PE	The mean square error was used to measure the error of predictors in forecasting the quality of a solution.
2.	Predictor count	PC	The number of predictors created in the training phase, believed to be indicative of the variations in slopes of basins of attraction.
3.	Predictor diversity	PD	The diversity of predictors is measured using the Canberra distance as described in Equation 11, a function used to quantify how similar two predictors are. The overall Canberra distance is averaged over the number of predictors.
4.	Descent Distance	DD	The number of improving steps between the initial and locally optimised best solution found over the trial. The values are averaged over the number of trials (restarts).
5.	Optima Fitness Variation	OFV	The variation in fitness between all local optima discovered during the search.
6.	Optima Distance Variation	ODV	The variation in the number of steps needed to find each local optimum from the inital solutions; the variation of DD.
7.	Fitness-distance correlation	FDC	The FDC between random samples and the known global optimum, Eq. 4.
8.	FDC of Local Optima	FDCLO	The FDC between all local optima discovered and the known global optimum.
9.	Autocorrelation	$\rho(10)$	The Autocorrelation between a random sample and a solution obtained by applying an operator ten times.
10.	Autocorrelation	$\rho(1)$	The Autocorrelation between a random sample and a solution obtained by applying an operator once.
11.	Autocorrelation Coefficient	AUC	The Autocorrelation coefficient ξ, Eq. 2, measured over random walks.
12.	Ruggedness Measure	RC	The ruggedness measure ζ based on the autocorrelation coefficient, Eq. 3.
13.	Negative Slope Coefficient	NSC	The Negative Slope Coefficient, Eq. 6 and 7. The NSC was obtained by uniform random sampling, not the Metropolis-Hastings method suggested in one of the applications (Vanneschi et al., 2004).
14.	Dominance of QAP matrix A	dom(A)	The variation coefficient of the distance matrix.
15.	Dominance of QAP matrix B	dom(B)	The variation coefficient of the flow matrix.
16.	Dominance of QAP both matices	dom (AxB)	The variation coefficient of the combined matrices. As the objective function of QAP multiplies the matrix entries, this value measures the variation coefficient of a matrix obtained by multiplying each entry in matrix A with each entry in matrix B.

$$F(x_1, x_2...x_n) = \sum_{i=1}^{n}\sum_{j=1}^{n} h_{ij} * d_{x_i x_j} \tag{13}$$

The objective is to assign each facility to a location such that the total cost is minimised. The quality of the solution is determined by the absolute position of each index (facility) in the permutation. Some of the problem instances used were procured from the QAP library (Burkard et al., 1991), some from a publicly available problem generator (Li and Pardalos, 1992), and some by altering known problem instances to change the features.

The instances used for evaluation in this work are listed in Table 14 in the appendix. The alipa, slipa and grid instances were created using the Li-Pardalos generator (Li and Pardalos, 1992) parametrised with the maximum values to use for each matrix (e.g. max1:50 is a maximum of 50 for Matrix 1). The instances were chosen/created ensuring the optimal solution is known.

5.3 Properties of Difficult QAP

5.3.1 Sparsity

The sparsity of the matrix is the percentage of zeros contained in the matrix. High matrix sparsity creates plateaus in the fitness landscape that inhibit the progress of a local search algorithm. Most of the problem instances in the QAP library originate in real-world problems, like hospital design (Krarup and Pruzan, 1978), backboard wiring (Steinberg, 1961), self-testable sequential circuits (Eschermann and Wunderlich, 1990), where many constraints are present. The sparsity of real-world instances varies strongly with values over 50%. The instances created using the problem generator (Li and Pardalos,

Evolutionary Computation Just Accepted MS.
doi:10.1162/EVCO_a_00177

1992) were designed to have a low sparsity.

5.3.2 Coefficient of Variation

The variation coefficient is an indicator of the variability in the matrices, independent of the size and distribution's range. This coefficient is calculated as the ratio of the standard deviation σ to the mean μ of the matrix entries as shown in Eq. 14.

$$c_v = \frac{\sigma}{\mu} \tag{14}$$

Consider two vectors $d_1 = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$ and $d_2 = \{1, 2, 3, 1, 2, 3, 1, 2, 3, 10\}$. The first vector d_1 has a standard deviation of 3.03 and a mean of 5.5, resulting in a variation coefficient of 0.55, whereas d_2 has a standard deviation of 2.5 and a mean of 2.8, resulting in a variation coefficient of 0.9. Matrices of distribution d_2 are harder to optimise, because all local optima are significantly worse than the one global optimum. Unless the search algorithm finds the global optimum, the performance of the optimiser is bound to appear poor.

5.3.3 Skewness

The skewness of matrices indicates the degree of asymmetry of the matrix entries in QAP. The skewness of a matrix can be expressed as the third moment of the mean normalised by the standard deviation as follows:

$$\gamma = \frac{\sum_{i,j=1}^{n} (y_{i,j} - \overline{y})^3}{(n-1)\sigma^3} \tag{15}$$

where n is the size of the matrix, $y_{i,j}$ is the matrix element located at row i and column j, $\overline{y}$ is the average of the matrix entries and σ is the standard deviation of the matrix. If the skewness is zero, the distribution of the matrix is symmetrical. The scale of skewness is not bounded, but the distribution is considered far from symmetrical if its skewness is above 1.0 or below -1.0.

5.3.4 Matrix Properties and Problem Difficulty

The variation coefficient of the matrices of QAP instances has been identified as a major contributor to the difficulty the instances pose to a heuristic by Merz and Freisleben (2000a). Sparsity clearly complicates the search through the introduction of plateaus. Ultimately, the difficulty of the problem is reflected in the quality of the solutions a gradient-exploiting heuristic can find or how long it takes for the heuristic to discover these solutions. The correlations between the solution quality found by a GA and the matrix properties can give an indication of the influence variation, skewness and sparsity of a QAP problem's matrices have on the difficulty of the problem.

Table 1: Correlations between the quality of the best solutions produced by a GA and the properties of the matrices.

	Sparsity	Variation	Skewness
GA	-0.57	-0.59	-0.42

The steady-state GA implementation used to establish the correlation has a population size of 100. After random initialisation, the distance-preserving crossover and mutation operators (Merz and Freisleben, 2000a) are applied for 100000 iterations, with the distance set to 5. Table 1 shows the Spearman correlations between the matrix properties and the mean best quality achieved by a GA on 30 repeat trials at solving the instances. For the purposes of this test, the highest scoring matrix (distance or flow) was chosen for the correlation. In the presence of two matrices, the values are by no means authoritative, but they show a clear negative correlations between all properties and the quality a GA can achieve. Skewness appears to have the least influence on problem difficulty, a notion that coincides with our preliminary experiments.

Evolutionary Computation Just Accepted MS.
doi:10.1162/EVCO_a_00177

5.4 Local Search

The problem instances were optimised using 30 repeat trials each applying (an approximation of) SD to uniformly randomly created solutions as described in section 4. The SD procedure stops when no further improvement is found by sampling the neighbourhood. The descriptor values as well as the best solution qualities were recorded.

The type of operator used for perturbations of solutions in iterative algorithms has been found to influence the success of the search (Moser and Mostaghim, 2010). QAP solutions are generally perturbed by transpositions (swapping assignments). Variation in a landscape can be evoked by changing the number of transpositions that form a move. To investigate the effect of different fitness landscapes, the following operators were applied in separate experimental trials.

2-opt operator: a simple local search algorithm defined as a pair-exchange neighbourhood which consists of all permutations obtained by applying a transposition of two elements in the original permutation.

3-opt operator: an extension of 2-opt that performs two pair-exchanges. The second pair-exchange is applied to the permutation resulting from the first pair exchange.

k-opt operator: for a fixed k, where k-opt is the natural generalisation of 2-opt and 3-opt, performs up to k pair exchanges to the original permutation with the condition that the intermediary transpositions display a fitness improvement (Lin and Kernighan, 1973). k is established by the immediate fitness deterioration compared to the k-1 pair-exchange. It is a conditional move based on improving exchanges, and the exit criterion is the fitness deterioration. It was designed to overcome fitness barriers caused by epistasis. In the experiments used for this work, a k-opt move swapped, on average, 2.19 assignments, with a standard deviation of 0.02 between the averages for the different instances.

k-opt-d: k-opt with deterioration was introduced to address the problem of entrapment in local optima. The operator performs a variable number of pairwise swaps, accepting deteriorating transpositions (as sub-moves) as long as the move improves on the current fitness. It is an extension of k-opt designed to overcome substantial fitness barriers, such as benches or peaks on the slope of a 2-opt landscape, as well as plateaus. In the experiments used for this work, a k-opt-d move swapped, on average, 2.41 assignments, with a standard deviation of 0.32 between the averages for the different instances.

The search operators were applied in separate trials to create different fitness landscapes. The average number of steps needed to reach the 'bottom of the basin' are reported here, averaged over the means for each problem instance. Table 2 shows the variations in the lengths of the descents to the optima.

Table 2: Number of times the operator was applied to a solution before the local optimum was reached. The minimum and maximum step averages list the instances where they occurred.

	2-opt	3-opt	k-opt	k-opt-d
Mean	25	17	21	22
Std dev	18.4	10.5	15.4	15.8
Min	rand-50-neg_a			
	3.5	3.3	3.1	2.9
Max	grid1	grid1	grid6	grid3
	73	41	61	63

The differences in the numbers of steps illustrate the significant differences in basin depth between the instances as well as the change in the number of steps incurred by increasing the size of the steps. As k-opt performs, on average, 2.19 transpositions and k-opt-d 2.41, it is not surprising that 3-opt performs the smallest number of steps to reach the optimum.

Evolutionary Computation Just Accepted MS.
doi:10.1162/EVCO_a_00177

5.5 Establishing Problem Difficulty

Appraising the expressiveness of a diagnostic measure of difficulty, it is necessary to have a priori information about the difficulty of the benchmark problem and problem instances used in the appraisal. The ultimate motivation for fitness landscape analysis is to establish the landscape features that render a problem difficult for certain stochastic solvers that attempt to exploit the landscape gradients in their search for the global optimum. It would therefore appear meaningful to adopt the quality achieved by a stochastic solver as a reference measure for the assumed problem difficulty.

Given the global optima are know for the instances we used, the quality achieved is normalised by the known global optimum. Nonetheless, the approach is not without its flaws; it is sensitive to the general elevation of the fitness landscape. The higher the number and fitnesses of the local optima, the better the solver appears to perform. Given that the comparison is between instances of the same problem, whose properties are relatively well known, we assume that this will not affect the results significantly.

As the descriptor values were collected during trials of PDO, we chose the quality produced by the algorithm while collecting the data as a reference for the assumed difficulty for maximum authenticity. The 30 repeat trials were allowed an equal number of $1,000,000$ function evaluations. The results for each of the operators are shown in Table 3. All operators except 3-opt produce almost identical quality with identical variation. This can be explained by the fact that k-opt and k-opt-d use, on average, only slightly more than a single transposition per move.

Table 3: Means and standard deviations of the fitness of the best solution achieved using PDO with each operator as well as an EA. The best results of the problem instances were averaged over 30 trials and reported as a ratio of the known best fitness.

	2-opt	3-opt	k-opt	k-opt-d	EA
Best fitness	0.92	0.89	0.92	0.92	0.87
Std dev	0.09	0.11	0.09	0.08	0.12

PDO uses uniform random initialisation of solutions to optimise. This is a very basic form of Iterated Local Search (ILS). Many more sophisticated variations exist Stützle (2006), but these are likely to explore the search space in a less uniform way, introducing a deliberate bias towards better regions of the search space. Such bias has been avoided in PDO in an attempt to favour a balanced characterisation of the search landscape at the expense of search potential. Because of the rudimentary search strategy, PDO might be suspected of producing a result quality that is not representative of the quality of popular and successful heuristics such as evolutionary algorithms (EA). The problem this entails is twofold: Firstly, since PDO's result quality is used as a reference value in the investigation, the validity of the results of this study might be in jeopardy. Secondly, if the method produced significantly worse results than a good heuristic solver for QAP, the only benefit would be its ability to produce values for some search space characterisation metrics.

To test the quality of the PDO search results, we optimised all instances also using an EA with the QAP-specific operators suggested by Merz and Freisleben (2000a). The steady-state implementation uses both distance-preserving crossover and distance-preserving mutation to create a new individual, which replaces the worst in a population of 100 individuals if its quality is better. Both the EA and PDO are allowed 100000 function evaluations, and the trials are repeated 30 times for statistical significance.

Unexpectedly, the EA is outperformed by the PDO approach regardless of operator (Table 3).

Table 4: Correlations between the quality of the best solutions produced by an EA and the solutions produced by PDO with the operators applied in this work.

	2-opt	3-opt	k-opt	k-opt-d
EA quality	0.96	0.98	0.96	0.95

Table 4 shows the Spearman correlation coefficients between the results produced with the PDO method, using each of the operators described in section 5.4 and the EA. Interestingly, the EA results correlate the strongest with the 3-opt operator, which applies the largest search move. Overall, the correlations are so strong that we can assume the results regarding the expressiveness of the difficulty measures are representative of the difficulty a problem poses to a stochastic solver well suited to the problem and not only the PDO method itself.

5.6 Correlation between Descriptors and Quality

To establish a correlation between the available difficulty measures on the one hand and the solution quality on the other, we computed the Spearman correlation between the descriptors and the PDO/EA qualities respectively.

Table 5 shows that the prediction error correlates most significantly with the PDO solution quality. The range [0.4 - 0.6] is defined as a medium Spearman correlation, and the values for the PE are at the lower limit of a strong correlation for all operators. Only in the case of k-opt-d is the correlation at the upper limit of the medium bracket.

The dominance of matrices A has the second-strongest correlation, while the flow matrices B hardly correlate at all. Not surprisingly, the dominance of the combined matrices lies between that of matrices A and B and is the forth strongest, following the the FDC of the local optima. The correlations are very similar across the operators and none of the other descriptors correlate more than weakly with the achieved solution quality.

Table 5: Correlations between the quality of the best solutions produced and the measures applied to determine problem difficulty.

	PDO Quality				EA Quality			
	2-opt	3-opt	k-opt	k-opt-d	2-opt	3-opt	k-opt	k-opt-d
PE	-0.61	-0.64	-0.61	-0.59	-0.72	-0.73	-0.73	-0.73
PC	-0.21	-0.30	-0.22	-0.18	-0.30	-0.37	-0.24	-0.23
PD	-0.20	-0.23	-0.20	-0.15	-0.32	-0.30	-0.25	-0.24
DD	0.18	0.20	0.18	0.14	0.24	0.27	0.23	0.24
OFV	-0.18	-0.18	-0.18	-0.21	-0.14	-0.15	-0.15	-0.14
ODV	0.24	0.24	0.27	0.29	0.35	0.31	0.38	0.46
FDC	-0.21	-0.22	-0.21	-0.19	-0.22	-0.22	-0.22	-0.22
FDCLO	-0.48	-0.49	-0.50	-0.53	-0.53	-0.54	-0.54	-0.59
$\rho(10)$	0.28	0.20	0.11	0.22	0.35	0.25	0.17	0.31
$\rho(1)$	0.20	0.16	0.21	0.19	0.27	0.24	0.28	0.30
AUC	0.20	0.16	0.21	0.19	0.27	0.24	0.28	0.30
RC	0.08	0.21	0.07	0.05	0.14	0.18	0.11	0.11
NSC	0.16	-0.22	-0.12	-0.05	0.15	-0.21	-0.1	-0.09
dom(A)	-0.55	-0.56	-0.55	-0.52	-0.63	-0.63	-0.63	-0.63
dom(B)	-0.01	-0.16	-0.11	-0.01	-0.22	-0.22	-0.22	-0.22
dom(AxB)	-0.44	-0.48	-0.43	-0.43	-0.53	-0.53	-0.53	-0.53

The correlation between the PE and the quality produced by an EA is even stronger than its correlation with PDO quality, and the operator used to produce the PE hardly influences the relationship. PC and PD also correlate strongly with the EA quality, refuting the suggestion that these measures are only indicative of the challenge a solver poses to PDO or, a local-search-based solver in general.

The dom(A) and dom(B) metrics were among four properties of QAP used in a study by Smith-Miles (2008) to predict the performance of an ILS algorithm, a Taboo Search and Max-Min Ant System. The study reports a strong correlation between expected and actual quality, which could not be attributed to the dom(A) and dom(B) values according to the correlations shown above. A reliable prediction of algorithm performance could only be explained by the other factors included (cardinality and sparsity) or the specific problem instances chosen.

Evolutionary Computation Just Accepted MS.
doi:10.1162/EVCO_a_00177

All measures reported appear to correlate more strongly with the EA quality, except FDC which is almost equally correlated. The correlations with EA quality do not vary by operator in the case of FDC and the matrix-based measures. FDC uses the Hamming distance for the distance calculation and is therefore independent of the operators.

The NSC values in Table 5 are based on uniform random sampling. The MH sampling suggested in one of the studies (Vanneschi et al., 2004) produced different values for each operator (2-opt: -0.25; 3-opt: -0.12; k-opt: -0.17; k-opt-d: 0.05 for PDO), but overall, it was not able to increase the correlation of NSC with the quality in a meaningful way.

6 Factor Analysis

Principal Components Analysis (PCA) and Exploratory Factor Analysis (EFA) are means of analysing patterns of association between variables by examining the covariance between these variables. The underlying assumption is that the larger the differences between values of the same variable, the more helpful the variable is in determining differences between incidences of the underlying data and the more accurately it can describe the incidences.

The landscape descriptors examined in this work were all devised to describe aspects of fitness landscapes with the intention of determining a problem's difficulty as experienced by a gradient-based solver. By analysing the collinearity of the descriptors, we hope to identify what properties of the fitness landscapes are captured by the descriptor variables, and ultimately, whether they can help determine the type and degree of difficulty a problem poses.

The main goal of PCA is to reduce the number of variables needed to describe the data adequately. Unlike EFA, it does not attempt to distinguish between common and specific variation in the variables. The variation a variable shares with all other variables in the model is assumed to be meaningful and descriptive of the dependent variables, whereas specific variation is exclusive to the variable and can be attributed to error or other circumstances which are not helpful in building a model of the data. EFA attempts to remove specific variation and build a model based on common variance alone. The proportion of common variation can only be approximated, and it is meaningful to conduct PCA as a first step to identify the potential number of common factors to extract from the data.

6.1 Determinimg the Number of Factors

PCA aligns the first component, which is a combination of varying contributions of the existing variables, such that this combination covers the largest variance found in the variables. The subsequent components are formed along lines perpendicular to all previously defined component alignments, ensuring that a specific part of a variable's variance is included in only one of the components. By definition, the first component is the most important in that it explains the largest variance in the data.

From the possible descriptors, we excluded the dom(A) and dom(B) factors. A preliminary PCA on all 16 variables confirmed that the variation of matrix B (dom(B)) is insignificant as a contributor and dom(A) contributes in almost exactly the same proportions as dom(AxB). Given the 14 remaining variables, we allowed for a total of 14 components initially. The resulting eigenvalues (variance covered) of the components and the cumulative proportions of variance covered are shown in Table 6.

Nine components can explain at least 97% of the variance in the data. Using all nine would be considered overfitting the model, as it is customary to exclude components with eigenvalues less than unity. Rather than aiming at building an expressive model, we are attempting to combine variables to extract meaningful landscape features and identify the variables that contribute to them. Also, PCA does not consider dependent variables and the correlation between the components extracted and the quality achieved by the solver have to be analysed in a separate step. Therefore we might consider five or six factors pending further investigation.

6.2 Exploring Factors

The variance of the variables of any data set is comprised of common and specific variances. Variance specific to a variable i (denoted δ_i) may be due to error or fluctuations unrelated to the model. Common variance (denoted χ) is the part of the variables' fluctuations that is shared between the variables and can therefore be explained as part of the model. Unlike PCA, EFA attempts to exclude specific variance

Table 6: Eigenvalues λ (total variance covered) of components and cumulative percentages (%) (of the sums of the eigenvalues) of variance covered by the components.

	2-opt		3-opt		k-opt		k-opt-d	
	λ	%	λ	%	λ	%	λ	%
C1	5.87	42	5.80	41	5.70	41	5.69	41
C2	2.08	57	2.06	56	2.01	55	2.11	56
C3	1.28	66	1.38	66	1.43	65	1.36	65
C4	1.20	74	1.12	74	1.21	74	1.17	74
C5	1.02	82	1.02	81	1.06	82	1.07	81
C6	0.75	87	0.79	87	0.80	87	0.76	87
C7	0.73	92	0.69	92	0.53	91	0.60	91
C8	0.42	95	0.52	96	0.44	94	0.46	94
C9	0.35	98	0.33	98	0.37	97	0.33	97
C10	0.15	99	0.12	99	0.18	98	0.19	98
C11	0.06	99	0.11	100	0.14	99	0.13	99
C12	0.05	100	0.04	100	0.07	100	0.10	100
C13	0.02	100	0.02	100	0.03	100	0.02	100
C14	0.02	100	0.01	100	0.02	100	0.02	100

from the model it builds of the data. Table 7 lists the approximated specific variance δ of each of the descriptors.

Table 7: Proportion of variances specific to the variables (descriptors) δ when 6 factors are extracted.

	2-opt	3-opt	k-opt	k-opt-d
PE	0.28	0.68	0.39	0.43
PC	0.04	0.02	0.04	0.01
PD	0.00	0.00	0.00	0.00
DD	0.07	0.00	0.12	0.09
OFV	0.85	0.97	0.87	0.93
ODV	0.01	0.00	0.03	0.00
FDC	0.84	0.68	0.89	0.86
FDCLO	0.00	0.31	0.00	0.32
$\rho(10)$	0.06	0.20	0.16	0.19
$\rho(1)$	0.00	0.00	0.00	0.00
AUC	0.05	0.03	0.04	0.00
RC	0.00	0.00	0.00	0.00
NSC	0.94	0.81	0.72	0.83
dom(AxB)	0.51	0.56	0.49	0.47

If we used five factors on the 2-opt data, PE would have a unique variance of $\delta = 0.42$ (not shown), whereas with six factors (Table 7) it is reduced to 0.28. Since PE has a significant correlation with result quality, it is interesting to see what other variables the difference of 0.14 correlates to, so we allow six factors.

Table 7 shows that the variables OFV, FDC and NSC have very high values of δ, hence we cannot expect them to contribute to the six factors in a major way - each of them describes largely a separate feature.

We extracted the six factors with an oblique rotation. Rotating factors (components) aims at increasing the fit with the group of variables whose covariance contributes the most to the factor. Oblique rotation does not enforce separation between the factors, and the factors may share the same variation of a variable.

Tables 8 and 9 show the *factor loadings* or correlations between the factors and the descriptor variables. The most significant correlations are marked with a dark background, the darker the more impor-

Table 8: Factor loadings (correlations between factors and variables) for the **2-opt** and **3-opt** operators, total variance explained (TVE) and correlations of factor scores with quality (QCorr).

	2-opt						3-opt					
Variable	F1	F2	F3	F4	F5	F6	F1	F2	F3	F5	F6	F4
PE	-0.05	0.42	0.03	0.10	-0.18	0.50	-0.20	0.32	-0.14	0.08	-0.2	0.12
PC	-0.03	0.93	0.01	0.09	-0.07	-0.01	0.05	1.00	0.05	0.05	-0.01	0.01
PD	-0.05	0.90	-0.01	0.04	0.17	0.03	-0.05	0.96	-0.01	-0.17	0.01	0.04
DD	0.98	-0.12	0.03	0.13	0.05	0.03	1.01	-0.01	0.00	-0.05	-0.04	-0.15
OFV	-0.13	-0.21	0.04	0.39	0.08	-0.20	-0.15	-0.10	-0.02	0.03	0.03	-0.13
ODV	0.97	0.06	0.15	-0.11	-0.07	-0.03	0.99	0.00	0.13	0.02	0.05	0.06
FDC	-0.02	0.08	-0.14	0.06	-0.15	0.29	-0.10	0.24	-0.37	0.27	-0.03	0.18
FDCLO	0.03	0.10	-0.04	0.94	-0.03	0.01	-0.08	0.62	-0.09	0.28	-0.05	-0.29
$\rho(10)$	0.79	-0.16	0.07	0.02	-0.05	-0.18	0.86	0.00	0.08	0.04	0.09	0.04
$\rho(1)$	0.52	0.02	-0.18	-0.11	-0.29	-0.41	0.61	-0.18	-0.19	0.06	0.39	-0.02
AUC	0.90	0.03	-0.29	-0.08	0.04	0.05	0.91	0.00	-0.25	0.02	-0.10	0.09
RC	0.02	0.01	0.99	-0.04	0.02	0.01	-0.01	0.02	1.00	0.02	0.00	0.01
NSC	0.03	-0.06	-0.19	0.06	0.18	0.06	-0.11	0.29	0.10	-0.01	0.38	0.08
dom(AxB)	0.04	0.19	0.06	-0.10	0.65	-0.07	-0.05	0.17	-0.06	-0.62	-0.02	0.02
TVE	26%	14%	9%	8%	5%	4%	29%	19%	9%	4%	3%	1%
QCorr	-0.03	0.02	0.04	-0.13	0.00	-0.23	0.13	-0.10	0.25	0.10	0.03	-0.03

tant the contribution of the variable to the factor. From these contributions we can infer what properties of the landscape the factors possibly describe. The factors are displayed in the order of the proportion of variance they explain (TVE), but their numbering follows our interpretation of the features they represent. To determine how well the factors express the difficulty of the problem, we calculated the factor scores for each of the problem instances based on the factor loadings shown. The resulting factor scores were used to establish the correlations with the quality the algorithm found (QCorr).

Factor F1 covers the largest proportion of the variance (TVE) except in the case of k-opt-d. In all cases, it has a very large contribution from the descent distance to the best local optimum found (DD). ODV contributes greatly to this factor, suggesting that when the descent to the global optimum is long, the distances to the local optima tend to vary more. Autocorrelation describes how consistent the development of a solution's fitness is over a number of changes and high autocorrelation expresses longer gradients. The contribution of the $\rho(10)$, $\rho(1)$ and AUC metrics seems to express that longer descents to the global optimum and large variances in the local optima in this case coincide with longer gradients. Even though the ruggedness measure RC is also based on $\rho(1)$, it does not contribute to F1. All contributors in factor F1 are related to the number of moves along a slope, which suggests that factor F1 describes the LS gradients of the basins of a problem. $\rho(10)$ measures a longer sequence of steps, making it more suitable for measuring gradients that are longer than a single step, and it contributes more to F1 than $\rho(1)$. The correlation with the quality achieved, however, is practically non-existent except with k-opt-d. Apparently the LS is not susceptible to variations in basin descents unless deteriorating moves are allowed. Conceivably the negative -0.22 correlation with k-opt-d indicates that varying gradients are better solved with an approach that is capable of jumping basins.

Factor F2 describes the second-largest part of the variance, except with k-opt-d, where it becomes more expressive than F1. F2 is dominated by the predictor-related measures, foremost the predictor count (PC) and diversity (PD). PE has a medium contribution to this factor, to a weaker extent with 3-opt and k-opt-d. The reason may be that there is less correlation between the fitness after the first step and the fitness after the entire local search with the 3-opt and k-opt-d operators, because they combine more transpositions (an average 2.41 in the case of k-opt-d) and therefore the final fitnesses at the local optima may be less distinctive – a smoothing effect of large moves is to be expected because they will stop short of making a final small improvement.

In the cases of 3-opt and k-opt-d, FDCLO and FDC contribute to F2, while OFV provides a weak contribution with 2-opt and k-opt. Based on the inclusion of the predictor count PC and the predictor diversity, this factor most likely describes the variability in the gradients. Varying gradients are usually described with the term ruggedness. The RC measure is intended to measure this feature, and it contributes to factor F2 in a small way.

Table 9: Factor loadings (correlations between factors and variables) for the **k-opt** and **k-opt-d** operators, total variance explained (TVE) and correlations of factor scores with quality (QCorr).

	k-opt						k-opt-d					
Variable	F1	F2	F3	F4	F5	F6	F2	F1	F3	F6	F5	F4
PE	-0.12	0.51	0.03	0.08	-0.22	-0.32	0.42	-0.10	-0.05	-0.47	0.16	0.13
PC	0.01	0.97	0.00	0.07	-0.06	0.05	1	-0.09	-0.04	0.07	0.06	0.03
PD	-0.02	0.93	-0.02	0.03	0.17	-0.02	0.93	0.02	0.03	-0.10	-0.22	0.00
DD	0.90	-0.15	0.09	0.13	0.00	0.06	-0.07	0.84	0.11	0.15	0.04	-0.02
OFV	-0.10	-0.28	0.00	0.39	0.08	0.08	-0.08	0.01	0.06	0.06	0.05	-0.27
ODV	0.94	0.05	0.10	-0.18	-0.09	0.02	-0.01	0.43	0.18	0.20	0.06	0.56
FDC	-0.09	0.18	-0.09	0.12	-0.16	0.02	0.20	-0.18	-0.24	0.02	0.20	0.05
FDCLO	0.03	0.11	-0.02	0.94	-0.04	-0.02	0.61	0.16	-0.03	-0.08	0.25	-0.36
$\rho(10)$	0.81	-0.04	-0.05	0.03	0.11	0.20	0.05	0.71	0.07	0.25	-0.06	0.09
$\rho(1)$	0.44	-0.05	-0.23	-0.15	-0.21	0.51	-0.04	0.22	-0.20	0.69	0.05	0.16
AUC	0.70	-0.04	-0.56	-0.06	0.01	-0.15	-0.10	0.79	-0.36	-0.07	0.04	0.13
RC	0.1	-0.01	1.00	-0.02	0.03	-0.06	-0.02	-0.01	0.98	-0.06	0.04	0.03
NSC	-0.54	0.03	-0.22	-0.07	0.04	0.01	0.22	-0.15	0.17	-0.02	0.15	-0.07
dom(AxB)	0.02	0.15	0.04	-0.11	0.66	-0.04	0.13	0.00	-0.10	-0.01	-0.69	-0.02
TVE	24%	16%	10%	8%	4%	3%	18%	16%	9%	6%	5%	4%
QCorr	-0.03	0.02	0.06	-0.20	-0.03	0.17	0.04	-0.22	-0.04	0.36	0.02	-0.05

Factor F3 is clearly dominated by the ruggedness factor RC. AUC and $\rho(1)$ also play minor roles in this factor, and when the landscape is formed by larger steps, FDC. RC is a factor of the cardinality of the problem and $\rho(1)$ expresses the consistency of the fitness development with a single move. The concept of ruggedness, as we understand it, describes a landscape that has gradients which are diverse in length as well as steepness. Whether RC can express this adequately, or whether F3 is simply an expression of the number of optima in the landscape remains an open question.

Given oblique rotation, the factors are not independent. Factor F3 and F1 have a considerable correlation (0.25 for 2-opt and 0.51 for k-opt). If we assume F1 to describe the length of the gradients, we would expect a relationship to exist, as long gradients lead to fewer optima.

Factor F4 is dominated almost exclusively by FDCLO. It plays a very minor role with 3-opt and k-opt-d. 2-opt and k-opt have similar step sizes (k-opt often defaults to 2 transpositions), hence similar properties are to be expected. FDCLO depends on a variation in basin depths for a meaningful FDC of local optima. It is likely that larger moves such as 3-opt 'smooth' the landscape, as they stop short of making a last small improving move. With 2-opt and k-opt F4 also has a relatively strong correlation with F1 (0.36 and 0.33) – longer gradients make for more distinct local optima, which raises the significance of FDCLO.

Factor F5 is dominated by the matrix variation dom(AxB). Depending on the operator, there is interplay with $\rho(1)$ or FDC and FDCLO, even PE. Matrix variation can lead to varying lengths of gradients or ruggedness – a multitude of optima of varying gradients and fitnesses. The very small contributions of some other variables on F5 may mean that it includes all of these aspects and therefore describes very many features at the same time. It has a medium correlation with F1 (-0.47 with 2-opt, 0.03 in the case of

3-opt, 0.36 with k-opt and 0.48 in the case of k-opt-d).

Factor F6 combines contributions from PE and $\rho(1)$, which may describe the 'predictability' of the fitness development, the homogeneity of the landscape. It has a small correlation with F1 except in the case of 3-opt (between 0.16 and 0.28).

The correlations of the factors with the optimum quality, assumed representative of the problem difficulty, are mostly negligible. F6, dominated by PE, has a correlation of 0.23 with 2-opt, whereas F3, dominated by RC, is correlated with quality to a 0.25 degree. K-opt shows a quality correlation of 0.20 for F4 (FDCLO) and k-opt-d for F1 (0.22). The variables when combined into factors are found to correlate rather poorly with the quality achieved by the solver. This may mean that we cannot assume the factors to represent distinct landscape features, or the landscape features to have a clear influence on the difficulty of the problem.

Table 10: Component loadings (correlations between components and variables) for the **2-opt** and **3-opt** operators, total variance explained (TVE) and correlation with quality of best solution produced (QCorr).

	2-opt					3-opt				
	C1	C2	C3	C4	C5	C1	C3	C2	C5	C6
PE	0.95	0.05	-0.01	0.01	0.00	1.00	-0.01	0.01	-0.03	0.00
PC	0.02	0.95	0.00	0.02	0.02	0.01	-0.01	0.95	0.04	0.08
PD	0.03	0.90	0.01	0.00	0.00	0.03	0.05	0.95	-0.02	-0.03
DD	0.03	-0.14	0.00	0.00	0.03	-0.05	-0.09	0.00	-0.03	0.06
OFV	0.00	-0.01	0.00	0.00	0.01	0.00	0.00	0.00	-0.01	0.00
ODV	0.00	0.06	0.01	0.00	0.15	-0.03	-0.03	0.05	0.09	-0.09
FDC	-0.01	0.00	0.00	1.00	-0.01	-0.01	-0.02	0.02	-0.03	0.00
FDCLO	0.03	0.10	0.00	0.01	-0.05	0.00	0.01	0.02	-0.01	0.98
$\rho(10)$	-0.15	-0.14	0.01	-0.01	0.07	0.07	0.09	-0.10	0.15	0.01
$\rho(1)$	-0.28	0.06	-0.05	-0.04	-0.20	-0.11	0.09	-0.20	-0.18	-0.03
AUC	0.07	0.03	0.00	-0.02	-0.30	-0.05	-0.04	0.06	-0.32	-0.05
RC	0.00	0.02	-0.02	-0.03	0.99	-0.04	0.01	0.02	0.97	-0.02
NSC	-0.01	0.00	1.00	0.00	-0.01	-0.01	0.99	0.03	-0.01	0.01
dom(AxB)	-0.03	0.03	0.00	-0.01	-0.01	-0.01	-0.01	0.00	-0.01	0.00
TVE	8%	30%	8%	9%	14%	9%	8%	31%	15%	8%
QCorr	-0.35	0.21	0.18	0.11	0.10	-0.42	-0.18	0.16	0.16	-0.13

Table 11: Correlations of the five most significant components for the **2-opt** and **3-opt** operators.

2-opt						3-opt					
	C1	C2	C3	C4	C5		C1	C3	C2	C5	C6
C1	1	0.66	0.03	0.26	-0.05	C1	1	0.08	0.43	0.03	-0.35
C2	0.66	1	0.21	-0.14	-0.02	C3	0.08	1	0.16	-0.16	-0.15
C3	0.03	0.21	1	0.04	-0.13	C2	0.43	0.16	1	0.11	-0.62
C4	0.26	-0.14	0.04	1	-0.11	C5	0.03	-0.16	0.11	1	-0.02
C5	-0.05	-0.02	-0.13	-0.11	1	C6	-0.35	-0.15	-0.62	-0.02	1

6.3 Components that Correlate with Quality

While EFA is generally considered more appropriate for the purpose of extracting patterns from the variance of a set of variables, in this case it seems that its strategy of removing the unique variance also removes correlation with the quality of the solution, suggesting that it is the unique variance of the variables that is most correlated with quality.

To include the specific variance, we extracted all 9 components that account for 98% of the total variance with oblique rotation, then correlated the components to the quality of the solutions, reporting the 5 components with the highest quality correlation. Allowing so many components removes the need to combine variation when the variables do not correlate very strongly. Consequently, the components focus almost a single variable each. The inclusion of the specific variance also contributes to this outcome.

Table 12: Component loadings (correlations between components and variables) for the **k-opt** and **k-opt-d** operators, total variance explained (TVE) and correlation with quality of best solution produced (QCorr).

	k-opt					k-opt-d				
	C1	C2	C6	C4	C7	C1	C2	C4	C7	C6
PE	0.88	0.11	0.03	0.05	0.00	0.93	0.07	0.03	-0.01	0.01
PC	0.04	0.95	0.06	0.02	-0.01	0.04	0.93	0.03	-0.01	0.09
PD	0.04	0.90	0.03	-0.01	-0.01	0.07	0.89	0.00	-0.01	0.04
DD	0.01	-0.19	0.17	-0.03	-0.04	0.01	-0.22	-0.01	-0.02	0.23
OFV	0.01	-0.01	0.01	0.00	1.00	-0.01	-0.01	0.00	1.00	0.01
OFD	0.04	0.03	-0.20	-0.05	-0.01	0.04	0.04	-0.03	0.01	-0.24
FDC	0.01	-0.01	0.00	1.00	0.00	0.00	-0.01	1.00	0.00	-0.01
FDCLO	0.03	0.11	0.92	0.02	0.04	0.06	0.24	0.03	0.07	0.80
$\rho(10)$	-0.18	0.02	0.03	0.05	-0.02	-0.19	0.14	0.00	-0.06	-0.04
$\rho(1)$	-0.30	0.06	-0.17	0.01	0.05	-0.26	-0.01	0.01	0.02	-0.08
AUC	0.11	-0.09	-0.08	-0.01	0.00	0.06	-0.12	-0.03	0.00	0.04
RC	0.03	-0.01	-0.03	-0.01	0.01	-0.01	-0.04	-0.03	0.01	0.01
NSC	0.02	-0.02	0.00	0.00	-0.01	0.01	-0.02	0.00	0.00	0.00
dom(AxB)	-0.03	0.04	-0.04	0.00	0.01	-0.01	0.02	0.00	0.00	-0.02
TVE	8%	26%	7%	8%	8%	8%	28%	8%	8%	9%
QCorr	-0.35	0.19	-0.16	-0.14	0.11	-0.33	0.20	-0.14	0.09	-0.09

Table 13: Correlations of the five most significant components for the **k-opt** and **k-opt-d** operators.

	k-opt						k-opt-d				
	C1	C2	C6	C4	C7		C1	C2	C4	C7	C6
C1	1	0.54	0.37	0.19	-0.04	C1	1	0.49	0.23	-0.03	0.33
C2	0.54	1	0.51	-0.23	-0.02	C2	0.49	1	0.22	-0.04	0.41
C6	0.37	0.51	1	0.22	0.21	C4	0.23	0.22	1	-0.12	0.15
C4	0.19	-0.23	0.22	1	-0.12	C7	-0.03	-0.04	-0.12	1	0.19
C7	-0.04	-0.02	0.21	-0.12	1	C6	0.33	0.41	0.15	0.19	1

Tables 10 and 12 show the 5 components included for their strong correlations with the quality of the result (QCorr). Component C1, which is comprised almost exclusively of the PE, has the strongest quality correlation (0.35), followed by component C2 (0.21) which is dominated by both the PD and PC measures. Only in the case of 3-opt does C2 rank behind C3 on the correlation with quality. C2 is a close equivalent of factor F2, whose quality correlation is almost nonexistent. The influence on problem difficulty seems to be due largely to unique variance.

C3 is based entirely on NSC and takes second place with 3-opt, but comes only ninth with k-opt and k-opt-d. NSC alone correlates a little more strongly with PDO-quality than the quality of EAs for which it was designed (Table 5). Varying the number of changes made in a single move appears to affect this operator which measures the fitness improvement of a single move and its dependency on the original fitness level. Its behaviour is unusual as 2-opt and k-opt are similar moves, but 3-opt is

Evolutionary Computation Just Accepted MS.
doi:10.1162/EVCO_a_00177

larger. C4 behaves as expected: It ranks fourth with both 2-opt and k-opt, but only eighth with 3-opt. It is dominated by FDC and correlates only by $0.11 - 0.14$ with the quality.

The RC loads strongly on C5 with 2-opt and 3-opt, but it ranks below the fifth component on k-opt and k-opt-d. C5 has significant contributions from AUC and $\rho(10)$, but the quality correlation of RC alone is stronger in the case of 3-opt. With larger steps in the search space the gradients become shorter, which increases the ruggedness and appears to indicate that RC indeed describes ruggedness.

With 3-opt, FDCLO-based C6 has a 0.13 correlation with quality, while none of the remaining components achieve a 10% correlation. With all operators, the C1 and C2 components have a medium correlation – rather expectedly, as they are all predictor-based metrics. The NSC-based C3 also correlates with C2 in a minor way in the case of 2-opt. The strongest correlation in the 3-opt results exists between C2 and the FDCLO-based C6. The same relationship is documented in factor F2 on the 3-opt data. It also applies with k-opt (0.51), where it is stronger than in the case of 2-opt (0.41; not shown - C6 ranks sixth with 2-opt). The correlation between FDCLO and the predictor metrics appears to become stronger with increasing size of operator move, possibly because large moves 'smooth' the landscape and remove the noise (small variations) in the fitnesses of neighbouring solutions. A similar pattern appears to apply with FDC, visible in the correlations between C2 and C4.

7 Findings Summary

7.1 Landscape Features

Based on the combinations of descriptors identified by the EFA, if appears that the following features can be described using the metrics selected:

- *Length of gradients*; descriptors DD, ODV, AUC, $\rho(10)$, $\rho(1)$ (k-opt/k-opt-d also NSC)
- *Basin/gradient depth* (variation of fitness of local optima); descriptors PC, PD, PE, (3-opt/k-opt-d also FDCLO, FDC)
- *Number of optima* (gradient length); descriptors RC, AUC, (3-opt/k-opt-d also FDC)
- *Fitnesses of global optima*; descriptor FDCLO, OFV (k-opt-d also ODV)

The features are listed in the order of the proportion of the total variance they capture, which is unrelated to the difficulty of the problem. The clear correlations between some variables seem to make it easy to identify some features of the landscape. However, most of the possible features of a landscape are related - long gradients lead to fewer optima, smoothness or homogeneity lead to uniform elevations and small fitness variations of local optima. It is hardly possible to identify any features definitively by the metrics.

However, the combination of DD, OD and AUC clearly appears to describe the lengths of the gradients. Unlike the basin depth, the gradient length is 'genotypic': It describes the number of steps it takes the LS to get to the optimum, rather than the fitness of the optima, which might be considered 'phenotypic'. Basin depth appears to express the fitnesses of the local optima and the differences between them. RC in combination with AUC may describe the number of the optima, but the combination of these two metrics might also reflect the fitness difference between two neighbouring solutions, including the expected variation of the fitness differences.

The fourth item is difficult to interpret; FDCLO measures whether local optima closer to the global best are more similar in fitness to the global best than more distant ones. There appears to be a relationship between variability in basin depth and fitness correlation of local optima with the fitness of the global optimum. Evidently, when there is no variation in the fitnesses of local optima, FDCLO will not be able to express much about the properties of a problem.

7.2 Diagnostic Value of Descriptors

Fitness landscape characterisation ultimately aims at a diagnosis how hard a problem is for a solver, possibly with the goal of identifying the best approach to solve it. Correlating the properties with the quality therefore appears meaningful. The aim of this paper was to combine descriptors both to identify landscape features and to find combinations of descriptors that may correlate with the search quality more reliably than a single descriptor could. Unfortunately, some of the individual descriptors correlate more with the quality than any of the factors.

Evolutionary Computation Just Accepted MS.
doi:10.1162/EVCO_a_00177

It was observed that PE with its borderline strong correlation with quality is the most promising among the descriptors explored. The second most correlated with quality is FDCLO. PE expresses how indicative the first improving step is of the rest of the local search. It appears similar in this respect to NSC. However, NSC accounts for the absolute 'elevation' of the solution at the time of the fitness improvement. Apparently, this knowledge is counterproductive. As long as we 'know' where the search is likely to end, it is irrelevant at what fitness we start the search. There is, however, a loading of both PE and NSC on some factors, indicating similarity of features measured.

Many heuristic solvers are population-based. At each iteration, they discard the 'worst' part of the population. This indicates a preference for large improvements early on in the search. When local search is applied, often the most fit solutions are chosen for the local optimisation. This may explain why metrics that consider the 'first step' of improvement are most reflective of expected solution quality. The fact that NSC does not meet this expectation may be due to the fact that is includes 'absolute' fitness values rather than the step improvement only.

7.3 Differences between Landscapes

The four operators differ in the magnitude of the change made to the solution. This varies the level of detail in the search landscape. The classic 2-opt move represents the smallest possible change to make to a permutation. The k-opt and k-opt-d operators appear to make mostly 2-opt moves interspersed with some larger steps. 3-opt is therefore the largest move proposed and it appears to have a 'smoothing' effect that eliminates the finer details of the 2-opt landscape.

Judging by the quality obtained, k-opt-d is the most 'optimisable' landscape. The average quality produced is on a par with 2-opt, but with a lower standard deviation. This appears to corroborate Stützle's observation that small local search steps lead to stagnation in local optima, and that accepting occasional moves to worse solutions can remedy this (Stützle, 2006).

Fortunately, the variability in step size does not decrease the descriptors' correlation with the search result quality. In the case of NSC, k-opt and k-opt-d are the only landscapes it can express. Given its emphasis on negative quality development, NSC can be seen as a 'red flag' that indicates when the landscape may be difficult. It is expressive only in the case of a problematic landscape.

PE appears slightly more correlated to quality when the search moves are larger. This is easy to understand, as finer landscape structures lead to small variations in basins which are near impossible to predict. Fortunately, this seems no real impediment to the reliability of the PE indicator with the landscapes of 2-opt and k-opt-d.

8 Conclusion

The scalar descriptors of landscapes that are currently known do not describe distinct features in the search landscape very well. Table 7 and the component loadings demonstrate that few descriptors covary sufficiently to give useful hints regarding the features the variables describe. Only the DD, ODV and autocorrelation-related measures appear to have a truly collinear variance. Also, the combinations of variables in factors correlate less significantly with quality than the individual descriptors.

The most reliable indicator of the quality to be expected from the optimisation of QAP problems appears to be PE. Although it correlates with PC and PD, the descriptors seem to describe slightly different aspects of gradients. PC and PD reflect the number of different basin shapes that are present, whereas PE describes the predictability of the gradients based on the first step.

The results suggest that the shape of the gradients (variations in steepness) have more influence on the quality achieved by a heuristic solver than the variations in the lengths of the gradients. If the initial steps leading to a deeper basin entail larger fitness improvements than those leading to shallower basins, the solving algorithm receives a clear indication which optimum to explore. Most current heuristics are specifically designed to exploit this information.

The correlations between the variables and the possible features have been explored exclusively with QAP data. A comparison with analogous data from other problems is necessary to corroborate these findings. In the second instance, an investigation into the suitability of metrics to identify the most appropriate solver is desirable.

Evolutionary Computation Just Accepted MS.
doi:10.1162/EVCO_a_00177

References

Altenberg, L. (1997). Fitness distance correlation analysis: An instructive counterexample. In *In Proceedings of the Seventh International Conference on Genetic Algorithms*, pages 57–64. Morgan Kaufmann.

Angel, E. and Zissimopoulos, V. (1998). Autocorrelation coefficient for the graph bipartitioning problem. *Theoretical Computer Science*, 191:229–243.

Angel, E. and Zissimopoulos, V. (2001). On the landscape ruggedness of the quadratic assignment problem. *Theoretical Computer Science*, 263(12):159 – 172. Combinatorics and Computer Science.

Barnett, L. (2001). Netcrawling-optimal evolutionary search with neutral networks. In *Evolutionary Computation, 2001. Proceedings of the 2001 Congress on*, volume 1, pages 30–37 vol. 1.

Burkard, R., Karisch, S., and Rendl, F. (1991). QAPLIB - a quadratic assignment problem library. *European Journal of Operational Research*, 55(1):115–119.

Chicano, F., Daolio, F., Ochoa, G., Verel, S., Tomassini, M., and Alba, E. (2012). Local optima networks, landscape autocorrelation and heuristic search performance. *CoRR*, abs/1210.4021.

Czech, Z. (2008). Statistical measures of a fitness landscape for the vehicle routing problem. In *IEEE International Symposium on Parallel and Distributed Processing*, pages 1–8.

Daolio, F., Vérel, S., Ochoa, G., and Tomassini, M. (2011). Local optima networks of the quadratic assignment problem. *CoRR*, abs/1107.4161.

Eschermann, B. and Wunderlich, H. (1990). Optimized synthesis of self-testable finite state machines. In *20th International Symposium on Fault-Tolerant Computing (FFTCS 20)*.

Fontana, W., Stadler, P., Bauer, E., Griesmacher, T., Hofacker, I., Tacker, M., Tarazona, P., Weinberger, E., and Schuster, P. (1993). RNA folding and combinatory landscapes. *Physical Review E*, 47:2083–2099.

Garnier, J. and Kallel, L. (2001). How to detect all maxima of a function. In *Theoretical aspects of evolutionary computing*, Natural Computing Series, pages 343–370. Springer Berlin Heidelberg.

Garnier, J. and Kallel, L. (2002). Efficiency of local search with multiple local optima. *SIJDM: SIAM Journal on Discrete Mathematics*, 15:122–141.

Gheorghita, M., Moser, I., and Aleti, A. (2013). Characterising fitness landscapes using predictive local search. In *Proceeding of the 15th Annual Conference Companion on Genetic and Evolutionary Computation Conference Companion*, pages 67–68.

Horn, J. and Goldberg, D. (1994). Genetic algorithm difficulty and the modality of fitness landscapes. In *FOGA*, pages 243–269.

Jansen, T. (2001). On classifications of fitness functions. In *Theoretical Aspects of Evolutionary Computing*, pages 371–386.

Jones, T. and Forrest, S. (1995a). Fitness distance correlation as a measure of problem difficulty for genetic algorithms. In *Proceedings of the Sixth International Conference on Genetic Algorithms*, pages 184–192.

Jones, T. and Forrest, S. (1995b). Genetic algorithms and heuristic search. Technical report, Santa Fe Institute.

Kallel, L., Naudts, B., and Schoenauer, M. (1999). On functions with a given fitness-distance relation. In *Proceedings of the IEEE Congress on Evolutionary Computation*, volume 3, pages 1910–1916.

Koopmans, T. and Beckmann, M. (1955). *Assignment Problems and the Location of Economic Activities*. Cowles Foundation for Research in Economics, Yale University.

Krarup, J. and Pruzan, P. M. (1978). Computer-aided layout design. In *Mathematical Programming in Use*, volume 9, pages 75–94. North-Holland Publishing Company.

Lance, G. and Williams, T. (1966). Computer programs for hierarchical polythetic classification ('similarity analyses'). *The Computer Journal*, 9(1):60–64.

Li, Y. and Pardalos, P. (1992). Generating quadratic assignment test problems with known optimal permutations. *COAP*, 1(2):163–184.

Evolutionary Computation Just Accepted MS.
doi:10.1162/EVCO_a_00177

Lin, S. and Kernighan, B. (1973). An effective heuristic algorithm for the travelling salesman problem. *Operations Research*, 21:498–516.

Malan, K. M. and Engelbrecht, A. P. (2013). A survey of techniques for characterising fitness landscapes and some possible ways forward. *Information Sciences*, 241:148–163.

Manderick, B., de Weger, M., and Spiessens, P. (1991). The genetic algorithm and the structure of the fitness landscape. In *ICGA*, pages 143–150.

Merz, P. (2004). Advanced fitness landscape analysis and the performance of memetic algorithms. *Evolutionary Computation*, 12(3):303–325.

Merz, P. and Freisleben, B. (2000a). Fitness landscape analysis and memetic algorithms for the quadratic assignment problem. *IEEE-Evolutionary Computation*, 4(4):337–352.

Merz, P. and Freisleben, B. (2000b). Fitness landscapes, memetic algorithms and greedy operators for graph bipartitioning. *Evolutionary Computation*, 8:61–91.

Moser, I. and Gheorghita, M. (2012). Combining search space diagnostics and optimisation. In *IEEE Congress on Evolutionary Computation*, pages 897–904.

Moser, I. and Mostaghim, S. (2010). The automotive deployment problem: A practical application for constrained multiobjective evolutionary optimisation. In *Evolutionary Computation (CEC), 2010 IEEE Congress on*, pages 1–8.

Pitzer, E. and Affenzeller, M. (2012). A comprehensive survey on fitness landscape analysis. *Studies in Computational Intelligence*, 378:161–191.

Quick, R., Rayward-Smith, J., and Smith, G. (1998). Fitness distance correlation and ridge functions. In *Parallel Problem Solving from Nature*, volume 1498 of *Lecture Notes in Computer Science*, pages 77–86. Springer.

Radcliffe, N. and Surry, P. (1995). Fundamental limitations on search algorithms:evolutionary computing in perspective. In *Computer Science Today*, volume 1000 of *Lecture Notes in Computer Science*, pages 275–291. Springer-Verlag.

Schiavinotto, T. and Stützle, T. (2007). A review of metrics on permutations for search landscape analysis. *Computers and Operations Research*, 34:3143–3153.

Smith-Miles, K. (2008). Towards insightful algorithm selection for optimisation using meta-learning concepts. In *Neural Networks, 2008. IJCNN 2008. (IEEE World Congress on Computational Intelligence). IEEE International Joint Conference on*, pages 4118–4124.

Stadler, P. (1995). Towards a theory of landscapes. In *Complex Systems and Binary Networks*, volume 461 of *Lecture Notes in Physics*, pages 77–163.

Stadler, P. (1996). Landscapes and their correlation functions. *Journal of Mathematical Chemistry*, 20:1–45.

Stadler, P. (2002). Fitness landscapes. *Applied Mathematics and Computation*, 117:187–207.

Stadler, P. and Schnabl, W. (1992). The landscape of the traveling salesman problem. *Physics Letters A*, 161:337 – 344.

Steinberg, L. (1961). The backboard wiring problem: a placement algorithm. *SIAM Review*, 3:37–50.

Stützle, T. (1999). *Local search algorithms for combinatorial problems - analysis, improvements, and new applications*, volume 220 of *Dissertations in Artificial Intelligence*. Infix.

Stützle, T. (2006). Iterated local search for the quadratic assignment problem. *European Journal of Operational Research*, 174(3):1519–1539.

Tayarani-N., M.-H. and Prügel-Bennett, A. (2014). On the landscape of combinatorial optimization problems. *Evolutionary Computation, IEEE Transactions on*, 18(3):420–434.

Vanneschi, L., Clergue, M., Collard, P., Tomassini, M., and Vérel, S. (2004). Fitness clouds and problem hardness in genetic programming. In Deb, K., editor, *Genetic and Evolutionary Computation GECCO 2004*, volume 3103 of *Lecture Notes in Computer Science*, pages 690–701. Springer Berlin Heidelberg.

Vanneschi, L., Tomassini, M., Collard, P., and Vérel, S. (2006). Negative slope coefficient: A measure to characterize genetic programming fitness landscapes. In Collet, P., Tomassini, M., Ebner, M., Gustafson, S., and Ekárt, A., editors, *Genetic Programming*, volume 3905 of *Lecture Notes in Computer Science*, pages 178–189. Springer Berlin Heidelberg.

Evolutionary Computation Just Accepted MS.
doi:10.1162/EVCO_a_00177

Vassilev, V., Fogarty, T., and Miller, J. (2000). Information characteristics and the structure of landscapes. *Evolutionary Computation*, 8(1):31–60.

Vollmann, T. E. and Buffa, E. S. (1966). The facilities layout problem in perspective. *Management Science*, 12(10):B–450–B–468.

Watson, J. P. (2010). *An Introduction to Fitness Landscape Analysis and Cost Models for Local Search*, volume 146 of *International Series in Operations Research & Management Science*. Springer US.

Weinberger, E. (1990). Correlated and uncorrelated fitness landscapes and how to tell the difference. *Biological Cybernetics*, 63:325–336.

Weinberger, E. D. (1991). Local properties of Kauffman's N-k model: A tunably rugged energy landscape. *Physical Review A*, 44(10):6399–6413.

Wright, S. (1932). The roles of mutation, inbreeding, crossbreeding and selection in evolution. *Proceedings of the Sixth International Congress of Genetics*, 1:356–66.

9 APPENDIX - Table of Problem Instances

Properties	Sparsity		Variation		Skewness		Origin	Comments
	M 1	M 2	M 1	M 2	M 1	M 2		
esc32a	85.55	18.75	2.81	0.69	3.14	0.23	QAPlib	symmetric
esc32b	78.91	18.75	2.08	0.69	1.95	0.23	QAPlib	symmetric
esc32c	74.41	18.75	2	0.69	2.08	0.23	QAPlib	symmetric
esc32d	82.42	18.75	2.35	0.69	2.48	0.23	QAPlib	symmetric
esc32h	72.46	18.75	1.88	0.69	2.22	0.23	QAPlib	symmetric
esc64a	96.83	10.94	5.71	0.59	6.47	0.14	QAPlib	symmetric
esc128	99.24	6.25	11.54	0.52	11.79	0.08	QAPlib	symmetric
kra30a	3.33	63.33	0.49	1.5	0.08	1.24	QAPlib	symmetric
kra30b	3.33	63.33	0.5	1.5	0.12	1.24	QAPlib	symmetric
kra32	67.77	3.13	1.64	0.49	1.43	0.1	QAPlib	symmetric
lipa30a	6.56	3.33	0.32	0.43	-0.79	-0.16	QAPlib	asymmetric
lipa30b	6.78	3.33	0.64	0.43	0.08	-0.16	QAPlib	asymmetric
lipa40a	4.94	2.5	0.28	0.42	-0.99	-0.2	QAPlib	asymmetric
lipa40b	5.19	2.5	0.64	0.42	0.05	-0.2	QAPlib	asymmetric
lipa50a	3.96	2	0.25	0.41	-1.16	-0.22	QAPlib	asymmetric
liap50b	4.12	2	0.62	0.41	0.01	-0.22	QAPlib	asymmetric
lipa60a	3.31	1.67	0.23	0.42	-1.3	-0.18	QAPlib	asymmetric
lipa60b	3.11	1.67	0.6	0.42	-0.01	-0.18	QAPlib	asymmetric
lipa70a	2.84	1.43	0.21	0.42	-1.44	-0.1	QAPlib	asymmetric
lipa70b	3.02	1.43	0.6	0.42	-0.01	-0.1	QAPlib	asymmetric
lipa80a	2.48	1.25	0.19	0.42	-1.56	-0.1	QAPlib	asymmetric
lipa80b	2.58	1.25	0.6	0.42	-0.01	-0.1	QAPlib	asymmetric
lipa90a	2.21	1.11	0.18	0.42	-1.67	-0.09	QAPlib	asymmetric
lipa90b	2.16	1.11	0.6	0.42	-0.01	-0.09	QAPlib	asymmetric
nug30	3.33	34.89	0.53	1.12	0.3	1.23	QAPlib	symmetric
ste36a	2.78	73.46	0.56	4	0.4	12.91	QAPlib	backboard wiring
ste36b	2.78	73.46	1.01	4	1.41	12.91	QAPlib	backboard wiring
tai30b	3.56	46.56	0.85	3.24	-0.03	4.46	QAPlib	symmetric
tai64c	95.87	1.56	4.82	1.28	4.61	2.55	QAPlib	symmetric
tho30	3.33	51.78	0.59	1.38	0.45	1.11	QAPlib	symmetric
c2rand_a	0	0	0.33	0.34	0.02	0.09	cust gen	urand $[0, 2)$ + 1
c2rand_b	71.25	0	21.15	0.33	0.32	0.02	cust gen	urand $[0, 2)$ + 1
cr_a	0.25	0	2.83	0.34	8.49	0.09	cust gen	98% const (1) and 2% urand $[0, 50)$
cr_b	71.25	0.25	21.15	2.83	0.32	8.49	cust gen	98% const (1) and 2% urand $[0, 50)$
exp2rand_a	50.75	0	1245.63	0.34	-6.69	0.09	cust gen	98% urand $[0, 2)$ and 2% neg. exp
exp2rand_b	71.25	50.75	21.15	1245.63	0.32	-6.69	cust gen	98% urand $[0, 2)$ and 2% neg. exp
g_a	67.25	0	25.5	0.34	-0.04	0.09	cust gen	grand
g_b	71.25	67.25	21.15	25.5	0.32	-0.04	cust gen	grand
gab_a	67.25	0	1.54	0.34	1.28	0.09	cust gen	grand abs
gab_b	71.25	67.25	21.15	1.54	0.32	1.28	cust gen	grand abs
gau50xcst_a	0	0	17.11	0.34	-0.1	0.09	cust gen	grand * 50, 0 replaced by 1
gau50xcst_b	71.25	0	21.15	17.11	0.32	-0.1	cust gen	grand * 50, 0 replaced by 1
r5g_a	71.25	0	21.15	0.34	0.32	0.09	cust gen	urand $[0, 5)$ * grand
r5g_b	71.25	74.75	21.15	-8.92	0.32	-1.86	cust gen	urand $[0, 5)$ * grand
rand50-neg_a	0.25	0	10.95	0.34	-8.4	0.09	cust gen	98% const and 2% urand $[-50, 0]$
rand50-neg_b	71.25	0.25	21.15	10.95	0.32	-8.4	cust gen	98% const and 2% urand $[-50, 0]$

continued on next page

Properties	Sparsity		Variation		Skewness		Origin	Comments
	M A	M B	M A	M B	M A	M B		
alipa1	1.99	10.64	0.17	-0.82	-1.78	-0.01	LP gen	asymm maxA:2 maxB:-10
alipa2	1.99	2.66	0.17	-0.62	-1.78	0	LP gen	asymm maxA:2 maxB:-50
alipa3	1.99	33.79	0.17	-280.58	-1.78	0.01	LP gen	asymm maxA:2 maxB:-3
alipa4	1.99	1	0.17	0.35	-1.78	-0.17	LP gen	asymm maxA:2 maxB:2
alipa5	1.99	1	0.17	0.53	-1.78	-0.02	LP gen	asymm maxA:2 maxB:10
alipa6	1.99	1	0.17	0.57	-1.78	-0.01	LP gen	asymm maxA:2 maxB:50
slipa1	1.99	1	0.17	0.42	-1.78	-0.05	LP gen	symm maxA:100 maxB:100
slipa2	1.99	1	0.17	0.42	-1.78	-0.06	LP gen	symm maxA:50 maxB:50
slipa3	1.99	1	0.17	0.42	-1.78	-0.05	LP gen	symm maxA:80 maxB:150
slipa4	1.99	1	0.17	0.39	-1.78	-0.09	LP gen	symm maxA:10 maxB:10
slipa5	1.99	1	0.17	0.42	-1.78	-0.05	LP gen	symm maxA:10 maxB:90
slipa6	1.99	1	0.17	0.35	-1.78	-0.19	LP gen	symm maxA:2 maxB:2
grid1	1.99	1	0.6	0.51	0.01	0.37	LP gen	grid maxA:100 maxB:100
grid2	3.13	1	0.61	0.51	0.01	0.37	LP gen	grid maxA:50 maxB:100
grid3	6.39	1	0.62	0.51	0.01	0.37	LP gen	grid maxA:20 maxB:100
grid4	11.23	1	0.65	0.51	0.02	0.37	LP gen	grid maxA:10 maxB:100
grid5	21.23	1	0.72	0.51	0.03	0.37	LP gen	grid maxA:5 maxB:100
grid6	50.99	1	1.02	0.51	0.04	0.37	LP gen	grid maxA:2 maxB:100