Example Landscapes to Support Analysis of Multimodal Optimisation

Thomas Jansen[1] and Christine Zarges[2(✉)]

[1] Department of Computer Science,
Aberystwyth University, Aberystwyth SY23 3DB, UK
t.jansen@aber.ac.uk

[2] School of Computer Science, University of Birmingham,
Birmingham B15 2TT, UK
c.zarges@cs.bham.ac.uk

Abstract. Theoretical analysis of all kinds of randomised search heuristics has been and keeps being supported and facilitated by the use of simple example functions. Such functions help us understand the working principles of complicated heuristics. If the function represents some properties of practical problem landscapes these results become practically relevant. While this has been very successful in the past for optimisation in unimodal landscapes there is a need for generally accepted useful simple example functions for situations where unimodal objective functions are insufficient: multimodal optimisation and investigation of diversity preserving mechanisms are examples. A family of example landscapes is defined that comes with a limited number of parameters that allow to control important features of the landscape while all being still simple in some sense. Different expressions of these landscapes are presented and fundamental properties are explored.

1 Introduction

Most real-world optimisation problems do not have a single best solution but many locally or globally optimal ones. The field of multimodal optimisation deals with tackling such problems and nature-inspired techniques have proven to be very popular and powerful to tackle these types of problems [17].

Over the last decade a rich set of benchmarks for the systematic and sound comparison of different optimisation methods has been developed[1]. Many problems in these benchmarks are multimodal. However, they are usually restricted to real-parameter optimisation problems and not accessible to theoretical analysis.

The authors want to thank the organisers of the Dagstuhl Seminar 15211 'Theory of Evolutionary Algorithms' for encouraging discussions that motivated this work. This article is based upon work from COST Action CA15140 'Improving Applicability of Nature-Inspired Optimisation by Joining Theory and Practice (ImAppNIO)' supported by COST (European Cooperation in Science and Technology).

[1] see, e.g., www.epitropakis.co.uk/cec16-niching/competition and coco.gforge.inria.fr.

J. Handl et al. (Eds.): PPSN XIV 2016, LNCS 9921, pp. 792–802, 2016.
DOI: 10.1007/978-3-319-45823-6_74

There has been some debate on appropriate optimisation goals in multimodal optimisation [2]. On one hand, one could be interested in the global perspective of locating a single (local or global) optimum. On the other hand, practitioners are often aiming at a multi-local perspective, i.e., they want to identify a multitude of different optima, either in a simultaneous or sequential fashion[2]. When considering such a multi-local perspective, *niching* techniques [19] are very common, i.e., techniques that prevent the algorithm from converging to a single solution and thus, enable it to explore multiple peaks of the search space in parallel. Some previous theoretical work consider Ising model problems [5,21] or simple bi-modal example function [6,16] exist. However, no common set of benchmarks suitable for theoretical analysis is available to date.

This lack of suitable benchmark functions is a serious impediment for the development of a theory of multimodal optimisation. In the area of classical optimisation where one is 'only' interested in finding an optimal search point simple example functions have been at the heart of the development of a powerful and useful theoretical framework and a multitude of strong theoretical results. Consider for example the well-known example function OneMax, used as early as 1992 to derive run time results for a simple evolutionary algorithm [15]. It has given rise to a natural generalisation, the class of linear functions [4] which in turn has motivated the introduction of a powerful proof technique: drift analysis [7]. And still today it is the function to consider when introducing novel perspectives [9] or expanding the horizon of theoretical analysis [12]. Clearly, OneMax is not the only useful and important example function but it is one of a relatively small number of example functions, most of which are unimodal (see [8] for a broad overview). Multimodal example functions are rarely considered–one noteworthy exception being TwoMax, a simple bi-model problem that can be seen as the maximum of OneMax and ZeroMax [6].

We address this need by introducing a family of landscapes with a limited number of parameters. We want to allow for the control of important features of problems that are simple enough for theoretical analysis. We explore properties of these 'theoretical' landscapes in the spirit of fitness landscape analysis that usually considers landscapes underlying real-world problems such as satisfiability [18] or are inspired by biology [20].

It is important to note that there has been some debate on appropriate example functions and optimisation goals in multimodal optimisation [2]. While the research in this paper is inspired by this discussion it goes beyond the initial ideas presented in [2] by introducing three different ways of implementation. One might want to argue that our example functions are inspired by and a generalisation of TwoMax, similar to linear functions being inspired by and a generalisation of OneMax. We think that the set of example functions presented here is a richer and more interesting generalisation. It bears resemblance with 'older' problem classes (e. g., [10,11]) but allows for more control. It is similar to the moving peaks benchmark [1] but it is static, of course.

[2] see, e.g., www.epitropakis.co.uk/ppsn2016-niching.

In the next section we present our main ideas behind our example functions. We describe the properties of some interesting landscapes in Sect. 3. We hint at the richness of the different example functions in Sect. 4.

2 Defining Landscapes and Objective Functions

We define our example functions based on an abstract idea of a landscape. It is important to note that we use landscape in a general, colloquial sense that does not coincide with the technical meaning of a fitness landscape as something that is defined by a neighbourhood graph and function values. We will be considering this latter kind of landscape (calling them *fitness landscape* to emphasise the difference) when we have defined objective functions.

We fix the set of bit strings of length n (equivalently, the Boolean hypercube of dimension n) as our search space. This is a complex, high-dimensional search space. Nevertheless, we *think* of it as a flat landscape where we introduce *peaks* that are defined by their *position*, their *slope* and their *height* (where we will give the height in an indirect way). The objective of an optimisation algorithm operating in this landscape is to identify peaks: a highest peak in exact optimisation, a collection of peaks in multimodal optimisation.

The kind of search heuristics we consider usually conduct search by modifying one or several bit strings they have explored already to get to another, yet unexplored bit string. The modifications tend to change only a limited number of bits and, therefore, it makes sense to use the Hamming distance between two bit strings as metric in our search space. The Hamming distance of x and y, $\mathrm{H}(x, y)$, equals the number of bits that have different values in x and y. Clearly, it is a value between 0 and n. If x is a point in our landscape we currently have and y is a point we want to reach then $\mathrm{H}(x, y) = 0$ indicates that we have reached the target point y. Since we will be considering maximisation it is more convenient to consider $n - \mathrm{H}(x, y)$ instead.

Definition 1. *For $x, y \in \{0,1\}^n$ let $\mathrm{H}(x,y) := \sum_{i=0}^{n-1} |x[i] - y[i]|$ denote the Hamming distance of x and y. We also define $\mathrm{G}(x,y) := n - \mathrm{H}(x,y)$.*

We now introduce our notion of a landscape that is defined by some number of peaks with their parameters that are introduced to the search space. We want to find these peaks and therefore consider the distance to a nearest peak.

Definition 2. *A landscape is defined by the number of peaks $k \in \mathbb{N}$ and the definition of the k peaks (numbered $1, 2, \ldots, k$) where the i-th peak is defined by its position $p_i \in \{0,1\}^n$, its slope $a_i \in \mathbb{R}^+$, and its offset $b_i \in \mathbb{R}_0^+$.*

For a search point $x \in \{0,1\}^n$ we define its closest peak (given by its index i) as $cp(x) := \operatorname*{arg\,min}_{i \in \{1,2,\ldots,k\}} \mathrm{H}(x, p_i)$. In cases where there are multiple i that minimise $\mathrm{H}(x, p_i)$ we define as tie breaking rule that i should additionally maximise $a_i \cdot \mathrm{G}(x, p_i) + b_i$. If this is still not unique an arbitrary i that minimises $\mathrm{H}(x, p_i)$ and among those maximises $a_i \cdot \mathrm{G}(x, p_i) + b_i$ can be selected.

The tie breaking rule we introduce is tailored towards the way we calculate fitness (which we define in Definition 3). Since we are interested in finding peaks it makes sense to concentrate on a higher one if there are multiple nearest peaks. Since we only care about distance and height we do not care about any tertiary criterion.

The general idea of our landscape is that the fitness value of a search point depends on peaks in its vicinity. For the sake of clarification, let us consider the situation for a landscape with only a single peak, i. e., $k = 1$ and the parameters of the peak are p_1, a_1, b_1. The fitness of $x \in \{0,1\}^n$ is given as $a_1 \cdot \mathrm{G}(x, p_1) + b_1$. We see that the peak itself has fitness $a_1 \cdot \mathrm{G}(p_1, p_1) + b_1 = a_1 \cdot n + b_1$. We call $a_1 n + b_1$ the height of the peak p_1.

It remains to be determined how we deal with multiple peaks. There are different ways this can be handled and there is no correct or incorrect way of doing it. It depends on what you want to achieve. We consider three different options and briefly discuss what we have in mind for the different versions.

Definition 3. *Let $k \in \mathbb{N}$ and k peaks $(p_1, a_1, b_1), (p_2, a_2, b_2), \ldots, (p_k, a_k, b_k)$ be given. We define the following three objective functions (also called fitness functions).*

- $f_1(x) := a_{cp(x)} \cdot \mathrm{G}\big(x, p_{cp(x)}\big) + b_{cp(x)}$, *called the nearest peak function*
- $f_2(x) := \max\limits_{i \in \{1,2,\ldots,k\}} a_i \cdot \mathrm{G}(x, p_i) + b_i$, *called the weighted nearest peak function*
- $f_3(x) := \sum\limits_{i \in \{1,2,\ldots,k\}} a_i \cdot \mathrm{G}(x, p_i) + b_i$, *called the all peaks function*

The nearest peak function, f_1, has the fitness of a search point x determined by the closest peak. The fitness is given as discussed above, $a_i \cdot \mathrm{G}(x, p_i) + b_i$, and the peak i that determines the slope a_i and offset b_i is the closest peak, $i = \mathrm{cp}(x)$. It implements a very local point of view where the height of other peaks is ignored even if their height is very much higher and they are only a little farther.

The weighted nearest peak function, f_2, takes the height of peaks into account. It considers $a_i \cdot \mathrm{G}(x, p_i) + b_i$ for all k peaks and uses the peak that yields the largest value to determine the function value. This implies that peaks with bigger height determine the function value in a larger area of the search space in comparison to smaller peaks.

The all peaks function, f_3, takes into account $a_i \cdot \mathrm{G}(x, p_i) + b_i$ for all peaks simultaneously and simply adds them up. Note that $f_3(x)/k$ yields the average influence of all peaks and in this sense we can view f_3 as an 'averaged' fitness landscape. Since many randomised search heuristics are rank-based [3] the difference between $f_3(x)$ and $f_3(k)/k$ is inconsequential.

We use the following visualisation of fitness landscapes resulting from the above definitions: We project the n-dimensional Boolean hypercube onto a 2-dimensional plane and connect direct Hamming neighbours by edges. We use a third dimension for the resulting fitness values, indicated by both height and colour (where blue indicates low fitness and red high fitness). An example for f_1

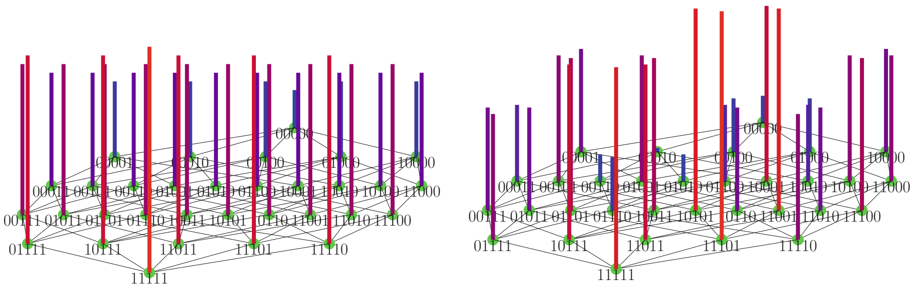

Fig. 1. Visualisation of fitness landscapes: f_1 with $n = 5$, $k = 1$, $p_1 = 1^n$, $a_1 = 1$ and $b_1 = 0$ (left) and f_3 with $p_1 = 11111$, $p_2 = 11001$, $p_3 = 10101$, $a_1 = a_2 = a_3 = 5$ and $b_1 = b_2 = b_3 = 0$ (right) (Color figure online)

with $n = 5$, $k = 1$, $p_1 = 1^n$ (the all-ones bit string), $a_1 = 1$ and $b_1 = 0$, which is identical to the well-known OneMax function, is shown in Fig. 1 (left).

We will discuss the differences between the three different fitness functions in more detail in the next two sections. We will also discuss which properties of the different fitness functions are of particular interest.

3 Properties

All three objective functions yield the same fitness landscapes for $k = 1$. They are all OneMax-like, i.e., p_1 is the single local and global optimum, fitness strictly decreases with increasing Hamming distance to p_1 and all points with equal Hamming distance to p_1 have the same fitness value. Consequently, we restrict ourselves to the more interesting case of $k > 1$.

When analysing fitness landscapes a variety of criteria can be considered (see, e.g., [20] for an overview). In this paper, we are particularly interested in the number of local and global optima and their locations in the search space. We additionally consider the so-called basin of attraction of a local optimum, i.e., the set of search points that are guaranteed to lead to it when using a simple hill-climber such as Random Local Search (RLS, Algorithm 1), and use as a measure for its size the probability that this happens when starting from a search point selected uniformly at random (u.a.r.).

Algorithm 1. Random Local Search (RLS)

Choose $x \in \{0,1\}^n$ u.a.r.
repeat
  Create offspring $y := x$. Select $i \in \{0, \ldots, n-1\}$ u.a.r. and flip bit $y[i]$.
  if $f(y) \geq f(x)$ **then** $x := y$
until *forever*

Additionally, we are interested in the influence of different parameters of landscapes (see Definition 2). This includes particularly the number of peaks k, their positions and heights (as defined by their slope and offset). Note, the peaks that we use to define a landscape do not necessarily correspond to a local optimum of the resulting fitness landscape (see Sect. 4.2).

4 Results

We provide some first insights into properties of our proposed set of example functions by considering a number of properties that are similar to properties of known example functions and are, we hope, of some general interest. The results in these sections hint at properties of different instantiations of our families of example functions that could be starting point for useful analysis of different randomised search heuristics. We examine a generalisation of the well-known TwoMax function [6] for all three fitness functions from Definition 3 in Sect. 4.1. Section 4.2 is dedicated to the comparison of f_1 and f_2. Looking at randomly distributed peaks we compare the influence of the slope as it manifests itself in f_1 and f_2. Looking at f_3 in Sect. 4.3 we consider an important property by means of a specific configuration of peaks.

4.1 Generalisation of TwoMax

As a starting point, we consider a landscape with two peaks $p_1 = 0^n$ and $p_2 = 1^n$ and see that for f_1 with offsets $b_1 = b_2 = 0$ and slopes $a_1 = a_2 = 1$ this is identical to the well-known bi-modal example function $\text{TwoMax}(x) := \max\left\{\sum_{i=1}^{n} x[i],\ n - \sum_{i=1}^{n} x[i]\right\}$. We examine all three fitness functions and different settings for the two offsets and slopes. In the following, let $|x|_1$ denote the number of 1-bits in x and $|x|_0$ the number of 0-bits.

It is easy to see that f_1 has exactly the two local maxima p_1 and p_2. Offsets and slopes influence only the fitness values but not the basins of attractions.

Theorem 1. *Let $p_1 = 0^n$ and $p_2 = 1^n$ with arbitrary $a_1, a_2 \in \mathbb{R}^+$, $b_1, b_2 \in \mathbb{R}_0^+$. The fitness landscape defined by f_1 has exactly two local maxima, p_1 and p_2, with fitness $a_1 \cdot |x|_0 + b_1$ and $a_2 \cdot |x|_1 + b_2$, respectively. RLS reaches p_1 with probability $1/2$ and p_2 otherwise.*

Proof. As discussed in Sect. 2, the fitness is only determined by the closest peak. It follows immediately, that the two peaks are both locally optimal and that each search point is in the basin of attraction of its closest peak. Plugging all parameters into Definition 3 yields the first statement. Let $\mathcal{B}_i$ denote the basin of attraction of p_i. For the second statement we need to prove that the RLS starts in $\mathcal{B}_1$ or $\mathcal{B}_2$ with equal probability. From the above, we see that all x with $|x|_0 > n/2$ are in $\mathcal{B}_1$ while all x with $|x|_1 > n/2$ are in $\mathcal{B}_2$. As both sets of points are of equal size RLS starts in either of them with equal probability. Points with $|x|_1 = |x|_0 = n/2$ have equal distance to p_1 and p_2 and belong to neither basis of attraction. Given such a point x, we know that RLS flips a 1-bit with probability 1/2 and a 0-bit otherwise. Thus, after one step, we are in one of the two previous cases. □

Things are different for f_2 as larger peaks have influence in a larger area of the search space in comparison to smaller peaks and thus will have a larger basin of attraction. We remark that our choice of p_1 and p_2 implies that two search points with the same number of 0-bits have equal fitness value. Thus, we can derive a bound on $|x|_0$ that determines the boundary of the basins of attractions of p_1 and p_2.

Theorem 2. *Let $p_1 = 0^n$ and $p_2 = 1^n$ with arbitrary $a_1, a_2 \in \mathbb{R}^+$, $b_1, b_2 \in \mathbb{R}_0^+$ and consider the fitness landscape defined by f_2. The basin of attraction of p_1 contains all search points x with $|x|_0 > a_2/(a_1+a_2) \cdot n + (b_2 - b_1)/(a_1+a_2)$.*

Proof. According to Definition 3, the fitness of a search point x is determined by p_1 if $a_1 \cdot (n - \mathrm{H}(x, 0^n)) + b_1 > a_2 \cdot (n - \mathrm{H}(x, 1^n)) + b_2$. We see that $|x|_1 = \mathrm{H}(x, 0^n)$ and thus, $|x|_0 = n - \mathrm{H}(x, 0^n)$. Similarly, we have $|x|_1 = n - \mathrm{H}(x, 1^n)$. We get $a_1 \cdot |x|_0 + b_1 \; > \; a_2 \cdot (n - |x|_0) + b_2$ which is equivalent to $|x|_0 > a_2/(a_1+a_2) \cdot n + (b_2 - b_1)/(a_1 + a_2)$ and see that all x with this property are in the basin of attraction of p_1. □

We see that RLS is initialised in the basin of attraction of p_1 with probability $1 - o(1)$ if $(a_2 n + b_2 - b_1) / (a_1 + a_2) = n/2 - \omega(\sqrt{n})$.

For f_3 all peaks have an influence on a search point's fitness. This leads to a very different structure of the fitness landscape.

Theorem 3. *Let $p_1 = 0^n$ and $p_2 = 1^n$ with arbitrary $a_1, a_2 \in \mathbb{R}^+$, $b_1, b_2 \in \mathbb{R}_0^+$. If $a_1 \neq a_2$, the fitness landscape defined by f_3 has a unique global optimum. If $a_1 > a_2$, this global optimum is p_1. Otherwise it is p_2.*

If $a_1 = a_2$, all search points have the same fitness $a_2 \cdot n + b_1 + b_2$.

Proof. According to Definition 3, the fitness of a search point x is

$$f_3(x) = (a_1 G(x, 0^n) + b_1) + (a_2 G(x, 1^n) + b_2) = (a_1 - a_2) \cdot |x|_0 + a_2 \cdot n + b_1 + b_2.$$

We see that $a_1 = a_2$ implies $f_3(x) = a_2 \cdot n + b_1 + b_2$, which is independent of x, proving the second statement. For $a_1 > a_2$ the fitness increases with increasing number of zeros and thus, p_1 is the unique global optimum. Similarly, it decreases with increasing number of zeros if $a_1 < a_2$. □

4.2 Comparing f_1 and f_2

The fitness landscapes defined as f_1 and f_2 are similar in nature. For both fitness landscapes the fitness is defined by only one of the peaks: for f_1 it is always the nearest peak; for f_2 the slope and offset of the peaks are taken into account so that 'higher' peaks can 'overrule' closer but smaller peaks. We formalise this by considering the set of local optima.

Theorem 4. *For f_1 and f_2 the set of local maxima is a subset of the peak locations $\{p_1, p_2, \ldots, p_k\}$. If the minimum Hamming distance between two peaks is at least 3 then the set of local maxima for f_1 is the set of peaks and, for f_2, the set of local maxima is a subset of the set of local maxima of f_1.*

Proof. If a point x is not a peak it has a Hamming neighbour with smaller Hamming distance to the peak that defines the function value of x. This proves that x cannot be a local maximum. Now, consider f_1 for a set of peaks that have minimum Hamming distance 3. Each Hamming neighbour y of a peak p has p as its nearest neighbour because the other peaks have Hamming distance at least 2 from y. This implies that $f_1(y) < f_1(p)$ and since this holds for each Hamming neighbour y we have that p is a local optimum. Finally, consider a peak p_i that is local maximum for f_2. We want to prove that p_i is also a local maximum for f_1. If the nearest other peak has Hamming distance at least 3 we are done. Consider a peak p_j with Hamming distance 1. We have that p_i is not a local optimum for f_1 if $f_1(p_j) > f_1(p_i)$ holds. But in this case $f_2(p_j) > f_2(p_i)$, too, so p_i is not a local maximum for f_2, either. Finally, consider a peak p_j with Hamming distance 2. Again, we have that p_i is not a local optimum if $f_1(p_j) > f_1(p_i)$ holds. But in the same way this implies $f_2(p_j) > f_2(p_i)$ and p_i is a local maximum for f_1. $\square$

Clearly, the question if the set of local optima for f_1 and f_2 differ for a given set of peaks depends on the parameters of the peaks. We consider the case of peaks with random positions to show a remarkable phase transition with respect to the other parameters, slope and offset. While the relative slope difference a_i/a_j can be arbitrarily large (measured in n) it turns out that constant bounds on the smallest and largest relative difference determine if f_1 and f_2 have completely equal or almost completely different local optima.

Theorem 5. *Let an at most polynomial number $k = n^{O(1)}$ of peaks (p_1, a_1, a_2), (p_2, a_2, b_2), $\ldots$, (p_k, a_k, b_k) with $a_1, a_2, \ldots, a_k \in \mathbb{R}^+$, $b_1, b_2, \ldots, b_k \in \mathbb{R}_0^+$ and $b_i \leq a_i$ for all $i \in \{1, 2, \ldots, k\}$ be given where the peak positions $p_1, p_2, \ldots, p_k$ are chosen independently, uniformly at random from $\{0, 1\}^n$. Let the minimum and maximum relative slope differences be $m := \min\limits_{i \neq j \in \{1,2,\ldots,k\}} a_i/a_j$ and $M := \max\limits_{i \neq j \in \{1,2,\ldots,k\}} a_i/a_j$. There exist constants $0 < c_1 < c_2 < 1$ such that if $m > c_2$ the set of local optima of f_1 and f_2 are equal to $\{p_1, p_2, \ldots, p_k\}$ with probability $1 - o(1)$ and if $M < c_1$ there are peak parameters with this value of M such that the set of local optima of f_1 and f_2 have only one element in common with probability $1 - o(1)$.*

Proof. We first show that the peaks are all in linear Hamming distance of each other with overwhelming probability. Consider two arbitrary peaks p_i and p_j. Considering p_i fixed, the expected number of bits equal in p_i and p_j when choosing $p_j \in \{0, 1\}^n$ uniformly at random equals $n/2$. Application of Chernoff bounds [14] and application of a simply union bound yields that for all pairs of peak positions p_i, p_j with $i \neq j$ we have $\Pr(\mathrm{H}(p_i, p_j) \in [(1 - \varepsilon)n/2, (1 + \varepsilon)n/2]) = 1 - e^{-\Omega(n)}$. We consider only the situation where this is the case.

We have $f_1(p_i) = a_i \cdot n + b_i$ and $f_2(y) = a_i \cdot (n - 1) + b_i$ for any Hamming neighbour y of p_i. We want to show that $f_2(p_i) = f_1(p_i)$ and $f_2(y) = f_1(y)$ holds which implies that p_i is a local optimum of f_2. We consider only p_i since the case y is very similar. We have $f_2(p_i) = \max\limits_{j \in \{1,\ldots,k\} \setminus \{i\}} \{a_i \cdot n + b_i, a_j \cdot (n - \mathrm{H}(p_i, p_j)) + b_j\}$.

Thus, we want to prove that $a_i \cdot n + b_i > a_j \cdot (n - \mathrm{H}(p_i, p_j)) + b_j$ holds. Remember that we have $b_j \leq a_j$ and $\mathrm{H}(p_i, p_j) \geq ((1-\varepsilon)/2)n$. Thus, it suffices if $a_i \cdot n > a_j \cdot n \cdot (((1+\varepsilon)/2) + (1/n))$ holds. With $a_i/a_j > ((1+\varepsilon)/2) + (1/n)$ this is the case so that choosing any $c_2 > (1+\varepsilon)/2$ suffices (because $a_i/a_j \geq m$).

On the other hand, we are also in the situation where $\mathrm{H}(p_i, p_j) < (1+\varepsilon)n/2$. We have $a_i n + b_i \leq (1+1/n)a_i n$ and $a_j \cdot (n - \mathrm{H}(p_i, p_j)) + b_j \geq a_j \cdot n \cdot ((1-\varepsilon)/2)$. Thus, if $a_i/a_j < ((1-\varepsilon)/2)/(1+1/n)$ we have that $f_2(p_i)$ is determined by the peak (p_j, a_j, b_j). Clearly, any constant $c_1 < (1-\varepsilon)/2$ suffices (because $a_i/a_j < M$). It is not hard to see that we can set the peak slopes in a way that f_2 is defined by the same peak (p_j, a_j, b_j) making p_j the only local (and thus also global) optimum. $\square$

4.3 Considering Properties of f_3

As a third example we consider an important property of f_3. For this we look at a landscape on $n = 5d$ bits with three clustered peaks $p_1 = 1^n$, $p_2 = 1^{2d}0^{2d}1^d$ and $p_3 = 1^d0^d1^d0^d1^d$, $a_1 = a_2 = a_3$ and arbitrary b_1, b_2 and b_3. Note, that the three peaks have pairwise equal Hamming distance $\mathrm{H}(p_i, p_j) = 2d$. We first observe that the fitness landscape based on f_3 has a unique global optimum that coincides with the centre of mass of the three peaks. An example for $d = 1$ is shown in Fig. 1 (right).

Theorem 6. *Let $p_1 = 1^n$, $p_2 = 1^{2d}0^{2d}1^d$ and $p_3 = 1^d0^d1^d0^d1^d$, $a_1, a_2, a_3 \in \mathbb{R}^+$ with $a_1 = a_2 = a_3$ and arbitrary $b_1, b_2, b_3 \in \mathbb{R}_0^+$. The centre of mass of the three peaks, i.e., $1^{3d}0^d1^d$, is the unique global optimum of the fitness landscape defined by f_3.*

Proof. Recall that $f_3(x) := \sum_{i \in \{1,2,\ldots,k\}} a_i \cdot (n - \mathrm{H}(x, p_i)) + b_i$. We first observe that the offsets b_i do not have an influence on the ranking of search points as $b_1 + b_2 + b_3$ is added to the fitness of all search points. Thus, we can ignore the b_i in the following. As $a_1 = a_2 = a_3$, search points maximising $\sum_{i \in \{1,2,\ldots,k\}} n - \mathrm{H}(x, p_i)$ will be assigned the maximal fitness value. It is easy to see that these are exactly the points that minimise the average Hamming distance to the given peaks. Using this, the first statement follows directly from the proof of Theorem 1 in [13] and we obtain the centre of mass by performing a simple majority vote for each bit position. $\square$

We remark that the above approach can be used to determine the set of global maxima for arbitrary sets of peaks. If $a_1 = a_2 = a_3$, we first obtain the set of search points with maximal fitness value by performing a simple majority vote for each bit position. Note, that in case of ties, search points with both bit values are assigned maximal fitness. For example, let us consider the above peaks with $d = 1$, i.e., $p_1 = 11111$, $p_2 = 11001$ and $p_3 = 10101$, and $p_4 = 00000$. We see that we have a tie for the 2nd and 3rd bits. Thus, we have four search points with maximal fitness value: 11101, 11001, 10101 and 10001. Given the set of search points with maximal fitness values we can then easily determine the set of global maxima.

The approach can also be generalised to peaks with different slopes by using a weighted majority vote where each a bit in p_i is assigned weight a_i. Let $W_0 = \sum_{i \text{ with } p_i[j]=0} a_i$ and $W_1 = \sum_{i \text{ with } p_i[j]=1} a_i$. We set the j-th bit to 0 if $W_0 > W_1$ and to 1 if $W_1 > W_0$. Ties are handled as discussed above.

References

1. Branke, J.: Memory enhanced evolutionary algorithms for changing optimization problems. In: Proceedings of CEC, pp. 1875–1882. IEEE Press (1999)
2. Doerr, B., Hansen, N., Igel, C., Thiele, L.: Theory of evolutionary algorithms (Dagstuhl seminar 15211). Dagstuhl Rep. **5**(5), 57–91 (2016)
3. Doerr, B., Winzen, C.: Ranking-based black-box complexity. Algorithmica **68**(3), 571–609 (2014)
4. Droste, S., Jansen, T., Wegener, I.: A rigorous complexity analysis of the (1 + 1) evolutionary algorithm for linear functions with Boolean inputs. In: Proceedings of ICEC, pp. 499–504. IEEE Press (1998)
5. Fischer, S., Wegener, I.: The one-dimensional Ising model: mutation versus recombination. Theor. Comput. Sci. **344**(2–3), 208–225 (2005)
6. Friedrich, T., Oliveto, P.S., Sudholt, D., Witt, C.: Analysis of diversity-preserving mechanisms for global exploration. Evol. Comput. **17**(4), 455–476 (2009)
7. He, J., Yao, X.: Drift analysis and average time complexity of evolutionary algorithms. Artif. Intell. **127**, 57–85 (2001)
8. Jansen, T.: Analyzing Evolutionary Algorithms. The Computer Science Perspective. Springer, Heidelberg (2013)
9. Jansen, T., Zarges, C.: Performance analysis of randomised search heuristics operating with a fixed budget. Theor. Comput. Sci. **545**, 39–58 (2014)
10. Jong, K.D., Spears, W.M.: An analysis of the interacting roles of population size and crossover in genetic algorithms. In: Schwefel, H.-P., Manner, R. (eds.) Proceedings of PPSN, pp. 38–47. Springer, Heidelberg (1990)
11. Kennedy, J., Spears, W.M.: Matching algorithms to problems: an experimental test of the particle swarm and some genetic algorithms on the multimodal problem generator. In: Proceedings of WCCI, pp. 78–83. IEEE Press (1998)
12. Kötzing, T., Lissovoi, A., Witt, C.: (1+1) EA on generalized dynamic onemax. In: Proceedings of FOGA, pp. 40–51. ACM Press (2015)
13. Moraglio, A., Johnson, C.G.: Geometric generalization of the Nelder-Mead algorithm. In: Cowling, P., Merz, P. (eds.) EvoCOP 2010. LNCS, vol. 6022, pp. 190–201. Springer, Heidelberg (2010)
14. Motwani, R., Raghavan, P.: Randomized Algorithms. Cambridge University Press, Cambridge (1995)
15. Mühlenbein, H.: How genetic algorithms really work: mutation and hillclimbing. In: Proceedings of PPSN, pp. 15–26. Elsevier (1992)
16. Oliveto, P.S., Sudholt, D., Zarges, C.: On the runtime analysis of fitness sharing mechanisms. In: Bartz-Beielstein, T., Branke, J., Filipič, B., Smith, J. (eds.) PPSN 2014. LNCS, vol. 8672, pp. 932–941. Springer, Heidelberg (2014)
17. Preuss, M.: Multimodal Optimization by Means of Evolutionary Algorithms. Springer, Heidelberg (2015)
18. Prügel-Bennett, A., Tayarani-Najaran, M.: Maximum satisfiability: anatomy of the fitness landscape for a hard combinatorial optimization problem. IEEE Trans. Evol. Comput. **16**(3), 319–338 (2012)

19. Shir, O.M.: Niching in evolutionary algorithms. In: Rozenberg, G., Bäck, T., Kok, J.N. (eds.) Handbook of Natural Computing, pp. 1035–1070. Springer, Heidelberg (2012)
20. Stadler, P.: Fitness landscapes. Biol. Evol. Stat. Phys. **585**, 183–204 (2002)
21. Sudholt, D.: Crossover is provably essential for the Ising model on trees. In: Proceedings of GECCO, pp. 1161–1167. ACM Press (2005)