Some Facts about So Called GA-Hardness Measures

Bart Naudts*,[1] and Leila Kallel[2]

[1] Department of Mathematics and Computer Science,
University of Antwerp, RUCA,
Groenenborgerlaan 171, B-2020 Antwerpen, Belgium.
Tel.: +32-3-218-0405
E-mail: *bnaudts@ruca.ua.ac.be*

[2] CMAP – URA CNRS 756, Ecole Polytechnique,
Palaiseau 91128, France.
Tel.: +33-1-69-33-46-11
E-mail: *kallel@cmapx.polytechnique.fr*

January 20, 1998

Abstract

In this paper we take a closer look at a number of so called GA-hardness measures, amongst which epistasis variance and fitness distance correlation are the better known. By generalizing the reference classes of the latter two measures, we are able to overcome their sensitivity to non-linear scaling. This results in the definition of measure which is more realistic with respect to the behavior of a genetic algorithm (GA). We show that the values of epistasis variance, epistasis correlation, fitness distance correlation and the site-wise optimization measure can be unreliable and entirely uncorrelated to convergence quality and convergence speed of the genetic algorithm on the fitness function in question, irrespective of the latter being an easy or a hard one. We give an overview of the relation between the reference classes of the four measures and a number of intuitive GA-easiness classes, in which we also locate our examples.

Keywords: genetic algorithms, GA-hardness measures, epistasis, fitness distance correlation

*Research assistant of the Fund for Scientific Research – Flanders (Belgium) (F.W.O.).

Introduction

In today's GA-community, measures like *epistasis variance*, introduced in (Davidor, 1991a), and *fitness distance correlation*, introduced in (Jones & Forrest, 1995), are commonly used as summary statistics, i.e., used to predict whether a given fitness function will be easy or hard for the genetic algorithm. Although there have been some recent experiments which suggest that the quality of these predictions is not always so reliable (e.g., in (Altenberg, 1997) for fitness distance correlation and in (Rochet, Venturini, Slimane, & Kharoubi, 1998) for epistasis variance), a thorough investigation of the properties and predictive qualities of measures like epistasis variance and fitness distance correlation has not been carried out yet.

In the first section, we try to define what it means for a fitness function to be easy for the genetic algorithm. We discuss an intuitive classification based on convergence quality and convergence speed of the GA.

The first part of the second section starts by discussing epistasis variance, *epistasis correlation*, introduced in (Rochet, Slimane, & Venturini, 1997), and fitness distance correlation. The inability of these measures to deal with constantness and, more importantly, their sensitivity to non-linear scaling motivate the need for a generalization of their reference classes (i.e., the classes they measure a distance from). The second part of the second section is devoted to this generalization. It presents a measure arising from it, called the *site-wise optimization measure* (of which a preliminary version appeared in (Naudts, Suys, & Verschoren, 1997)). The relations between the function classes defined in this section are extensively investigated.

In the third section we present more general function classes which are well known to consist of (or at least, contain) easy functions: the unimodal (or steepest-ascent optimizable) functions and the GA-easy functions (in the sense of schema competitions). We conclude the third section by defining the function class from which we will draw examples which mislead the so called GA-hardness measures discussed in the second section.

The main goal of this paper is to show that the values produced by the four measures of Sect. 2 can be entirely unrelated to the convergence behavior of the GA. After having shown that non-linear scaling troubles the epistasis measures and fitness distance correlation, we present in Sect. 4 three kinds of fitness functions for which none of the measures can predict the convergence behavior. Worse, we find for example epistasis variance giving low values to almost unsolvable problems and higher values to completely trivial ones.

The first part of the fifth section comments on how to draw the samples which are used to approximate the values of the measures for a given fitness function. We propose (for further investigation) two alternatives to random sampling: Metropolis-generated samples and strings sampled from successive generations of a GA run. In the last part of the fifth section, we discuss how the results of the previous sections apply to Reeves and Wright's Anova tables (Reeves & Wright, 1995), which give much more detailed information about the epistatic structure of a fitness function than the epistasis measures do.

It is the intention of this paper to provide ample evidence of the claims made. Some examples and proofs are deferred to the appendix, though, to improve the readability of the text.

1 Notations, definitions, conventions

Throughout this text, $S = \Sigma^L$ will denote the search space, where Σ is a finite alphabet and L the set of sites, with $|L| = \ell$, the string length. Usually $\Sigma = \{0, 1\}$ and $L = \{0, 1, \ldots, \ell - 1\}$. If $\alpha \in \Sigma$, then we use the notation $\alpha^{(n)}$ to denote the concatenation of n copies of α, i.e., $\alpha^{(n)} = \alpha\alpha \ldots \alpha$. A schema (or hyper-plane) is a string in Σ'^L, with $\Sigma' = \Sigma \cup \{\#\}$, the alphabet augmented with a wildcard symbol. The fitness of a schema is defined as the average of the fitness values of the strings belonging to that schema. The order of a schema is the number of non-wildcard symbols in the schema.

The Hamming distance between two strings $s^1, s^2 \in S$ is denoted $\mathrm{d}(s^1, s^2)$. We use the notation $s[i|\alpha]$ to indicate the string s with symbol α put at site i. The string $s \in S$ is a local optimum (w.r.t. the Hamming distance) of the fitness function f with global optimum s^* when $f(s) < f(s^*)$ and there exist no sites i and symbols α such that $f(s[i|\alpha]) > f(s)$. We say that a fitness function f is unimodal, and we write $f \in \mathsf{UNI}$, when it has no local optima. We call it multi-modal when it has at least one local optimum. A fitness function f is called *monotone*, which is denoted by $f \in \mathsf{MON}$, when it satisfies

$$\mathrm{d}(s^1, s^*) < \mathrm{d}(s^2, s^*) \implies f(s^1) > f(s^2) \qquad \text{for all } s^1, s^2 \in S. \tag{1}$$

Throughout this paper we use the same genetic algorithm to observe the hardness of fitness functions. Unless otherwise stated, the alphabet is binary and the string size is 80. The population size remains fixed at 80 strings. We use uniform crossover, with a probability of 0.5 to choose between crossing over or cloning two parents which are selected using linear ranking selection. After cloning or crossover, one of the offspring is randomly selected to undergo mutation and to become part of the next population. We apply bit flip mutation with a rate of $1/3\ell$. We synthesize a run of this genetic algorithm by recording the number of generations needed to detect the optimum of the fitness function. Finally, we present an average number of generations computed of recordings of 21 runs with different randomly sampled initial populations. We also show the standard deviation in the form of error bars.

Although it is beyond the scope of this paper to give a full account of the subject, we want to specify statement " ... are easy for the GA" a bit more. Intuitively, we can make the following hierarchy of *convergence quality* of a given GA and fitness function:

1. The GA detects the optimum of the fitness function independently of the choice of the randomly sampled initial population. The variance on the number of iterations needed to detect the optimum is comparatively small. This behavior is called *stable convergence*.

2. Each run of the GA still detects the optimum of the fitness function, but now the variance on the number of iterations needed for detection is comparatively large. The GA exhibits *unstable convergence*.

3. The GA sometimes detects the optimum of the fitness function after a reasonable number of iterations, but it takes a number of runs because the problem is sensitive to the choice of initial population and the choices made by the algorithm.

   This behavior can be seen as follows: there are a number of easy paths to the global optimum, but if the GA is not in one of them then it gets trapped in a local optimum from which it cannot escape.

4. The GA does not perform better than random search. It cannot detect the optimum of the fitness function in a reasonable length of time, no matter how often it is run and how its parameters are tuned.

Note that this is a rough and intuitive classification. Whether a given fitness function belongs to one convergence quality class or not can heavily depend on the choice of parameters of the GA. We stick to the GA defined above in this section to classify fitness functions in this paper.

Experiments (as in Fig. 1) show that monotone and site-wise optimizable (defined in Sect. 2) fitness functions belong to first convergence quality class. Within this class, *convergence speed* can be used as a refinement in the hierarchy of GA-easiness. This allows us, for example, to distinguish between monotone functions and longpaths for the GA defined in (Kallel & Naudts, 1998) which typically show first convergence quality but extremely low convergence speed (even exponential in string size). (The latter is not to be confused with the longpath defined in (Horn, Goldberg, & Deb, 1994) which is a longpath for a hill climber in Hamming space). In the same way, a partial order can be made within the second convergence quality class, simply by comparing the mean and standard deviations of convergence time for two functions. A better refinement order may be obtained using some statistical test (as the t-test) between two sets of convergence times.

2 Monotonicity

2.1 Epistasis measures and first order functions

2.1.1 Epistasis variance

Davidor (1991a) proposed to compute the epistasis of a binary string $s \in S$ with respect to a fitness function $f : S \to R$ and a sample P as $\varepsilon_P(s) = f(s) - \xi_P(s)$, with

$$\xi_P(s) = \sum_{i \in L} \frac{1}{|\{t \in P\,;\, t_i = s_i\}|} \sum_{\substack{t \in P \\ t_i = s_i}} f(t) - \frac{\ell - 1}{|P|} \sum_{t \in P} f(t). \tag{2}$$

He defined the *epistasis variance* of f w.r.t. P as a scaled version of the Euclidean distance between f and the value ξ_P, i.e.,

$$\varepsilon_P^2(f) = \frac{1}{|P|} \sum_{s \in P} \varepsilon_P^2(s) = \frac{1}{|P|} \sum_{s \in P} (f(s) - \xi_P(s))^2. \tag{3}$$

A fitness function is said to be a first order (or linear, fully separable) function when it is of the form

$$s \mapsto \sum_{i \in L} g_i(s_i), \qquad \text{with } g_i : \Sigma \to R, \quad \text{for all } i \in L. \tag{4}$$

We write $f \in \mathsf{1ORD}$ when f is a first order function. It has been proved independently in (Naudts et al., 1997), in (Rochet, Slimane, & Venturini, 1996) and implicitly also in (Reeves & Wright, 1995) that ξ_P is the first order (or linear) approximation of f. More precisely, ξ is that first order function which is closest to f in Euclidean distance.

A kind of normalization is required to achieve the equality of the epistasis variance of f and of αf, for $\alpha > 0$. Manela and Campbell (1992) propose to divide the epistasis variance by the variance of the fitness function. Alternatively, as in (Naudts et al., 1997), one can first normalize the fitness function, using the Euclidean norm, before computing the unscaled epistasis variance. This approach guarantees that epistasis variance takes values in the closed interval [0, 1].

In this paper we use the unscaled Euclidean distance and Euclidean normalization. To compute the epistasis variance over the sample P, we write

$$\mathsf{epiv}_P(f) = \frac{\sqrt{\sum_{t\in P}(f(s) - \xi(s))^2}}{\sqrt{\sum_{t\in P} f(t)^2}}. \tag{5}$$

By $\mathsf{epiv}(f) = \mathsf{epiv}_S(f)$ we denote the epistasis variance computed over the *whole* search space. Note that $\mathsf{epiv}(f) = 0$ exactly when f is a first order function.

2.1.2 Epistasis correlation

Epistasis correlation, proposed in (Rochet et al., 1997), computes the (statistical) correlation between a fitness function f and its first order approximation ξ over a sample P:

$$\mathsf{epic}_P(f) = \frac{\sum_{t\in P}(f(t) - \bar{f}_P)(\xi(t) - \bar{\xi}_P)}{\sqrt{\sum_{t\in P}(f(t) - \bar{f}_P)^2}\sqrt{\sum_{t\in P}(\xi(t) - \bar{\xi}_P)^2}}, \tag{6}$$

with

$$\bar{f}_P = \frac{1}{|P|}\sum_{t\in P} f(t), \qquad \bar{\xi}_P = \frac{1}{|P|}\sum_{t\in P}\xi(t). \tag{7}$$

We use $\mathsf{epic}(f)$ to denote the epistasis correlation of f computed over the whole search space. Being a correlation coefficient, $\mathsf{epic}_P(f) = \mathsf{epic}_P(af + b)$ for all $a, b \in R$.

Epistasis correlation values are in the range [0, 1]. A correlation value of 1 indicates that the fitness function is a first order function with the property of

$g_i(\alpha) \neq g_i(\beta)$ for all symbols $\alpha \neq \beta$ (using the notations of Eq. 4). A correlation value of 0 indicates no correlation with first order functions.

Epistasis correlation has the immediate advantage over epistasis variance of also having a correlation plot that is interpretable over the mere distance produced by the other, which is hard to interpret. Unlike epistasis variance with Euclidean normalization, it is, as stated above, translation invariant.

2.1.3 Discussion

While epistasis variance was an attempt at detecting or measuring the epistasis (amount of interaction between sites) of a fitness function, it merely detects the absence of epistasis. This was already pointed out in (Davidor, 1991b); it is obvious that this is also the case for epistasis correlation.

Having accepted their somewhat ambitious name, there is a more technical but not too important problem the epistasis measures suffer from which is caused by the choice of reference class. It is easy to verify that the first order functions contain all constant functions. Consequently, epistasis variance will return the value 0 both for constant fitness functions and first order functions with $g_i(0) \neq g_i(1)$ for all i (using the notations of Eq. 4). This is counterintuitive because the GA deals very differently with the two types of functions. The latter are easily optimized, while constant areas in the fitness landscape decrease the search efficiency of the GA. Consider the special case of the following *needle in a haystack* problem.

In (Naudts et al., 1997) it is proved that maximal epistasis variance is obtained for a function which is zero everywhere but in two bit-complementary strings. Of course, approximations of epistasis variance are very unlikely to find exactly those two strings, and will report zero epistasis variance, because of constantness. We will demonstrate this approximation problem in the appendix by computing the exact and approximated values for the Royal Road functions (see, e.g., (Mitchell, Forrest, & Holland, 1993)). Rather unexpectedly, we also came across an approximation problem with epistasis correlation. It is also reported in the appendix.

The *noise-to-signal epistasis measure* (presented in the appendix, Def. 1) is just an example of another way of measuring the distance between a fitness function and the first order functions. As it turns out, it makes the constantness problem suffered by epistasis measures more explicit, to the extent that this measure is useless in practice.

A serious weakness of the epistasis measures is their sensitivity to non-linear scaling, something a genetic algorithm with ranking selection is entirely insensitive to. We will show that they can take any value of their range when presented with a monotone fitness function. Let $\Sigma = \{0, 1\}$. For each positive $n > 0$, let $l(i) = (i/\ell)^n$ for all $i \in \{0, \ldots, \ell\}$. We construct a monotone fitness function NLM_n by setting $\mathrm{NLM}_n(s)$ equal to $l(1(s))$, with $1(s)$ the number of 1's in s. The higher we take this control parameter n, the more the fitness value of optimum $1\ldots11$ gets isolated from the mass of the fitness values, which results in, e.g., a fitness distance correlation going to 0 for both n and ℓ going to infinity. Figure 1 shows the results of the experiment: both epistasis variance and epistasis correlation take all values between 0 and 1. Note that proportional selection sees a stronger and stronger pressure towards strings with more ones.

We can conclude that the epistasis measures cannot characterize GA-easy functions, i.e., we have that nor high epistasis variance nor low epistasis correlation

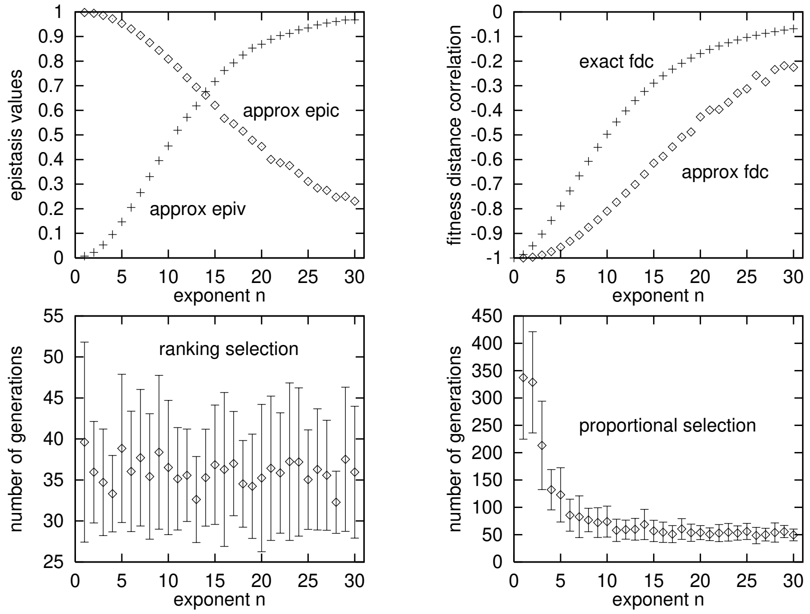

Figure 1: Epistasis variance, epistasis correlation, fitness distance correlation and average number of generations to detect the optimum of the **NLM$_n$ monotone functions**, as defined in Sect. 2.1.3, for n ranging from 1 to 30. The approximations are averaged over 21 values using samples of size 2^{12}. In the bottom left picture, we show the results obtained with the GA described in Sect. 1 (linear ranking selection, uniform crossover, ordinary mutation). In the bottom right picture, proportional (roulette wheel) selection is used. Because of the small population size (80 individuals), the low crossover rate of 0.5 and a mutation rate which is kept all the time at $1/3\ell$, there is not enough selection pressure for the cases $n = 1$ and $n = 2$ to converge (quality class 3 convergence). Note that the GA behaves better with the values of the measures getting worse.

implies GA-hardness. Other examples have lead Manela and Campbell (1992), amongst others, to come to the same conclusion.

2.2 Fitness distance correlation and linear functions

When the *fitness distance correlation* (Jones & Forrest, 1995) of a fitness function f equals -1, then f is necessarily of the form $s \mapsto C_1 - C_2\,\mathrm{d}(s, s^*)$, where C_1 and C_2 are constants and s^* is the global optimum of f.

Notationally, we use $\mathsf{fdc}(f)$ to denote the fitness distance correlation value of a fitness function f. It is easy to see that these linear monotone functions are contained in the first order functions, the reference class of the epistasis measures.

A first and obvious drawback of fitness distance correlation is that one of the global optima has to be known. Kallel and Schoenauer (1996) have proposed a way to overcome this, by approximating the correlation with respect to the k best values found in the approximation sample.

The fitness–distance plot of the whole search space provides a lot more information than the correlation value. But this information can be extremely misleading, mainly because two points which are close together in the plot do not have to be close in Hamming space. In (Kallel & Naudts, 1998) we study this phenomenon in greater detail.

Fitness distance correlation is also sensitive to non-linear scaling. Figure 1 shows exact and approximated values for the NLM_n functions. A first suggestion to cure this sensitivity is to perform a ranking on the fitness values before computing the correlation coefficient. The limitation of this solution is that it is only useful when there are not more different fitness values than there are distances to the global optimum. An optimal correlation would then be obtained for all functions of the form $s \mapsto C_1 - C_2\,\mathrm{d}(s, s^*)^n$, for example.

Figure 1 also shows that the exact and approximated fitness distance correlations disagree for exponents $n > 1$ and sample sizes of the order of 2^{12}. Moreover, the values of epistasis correlation and fitness distance correlation for NLM_n are similar. The same phenomenon occurs in example 2 of the appendix, where values for the Royal Road functions are computed.

An immediate conclusion is that the fitness distance correlation cannot reliably be related to GA convergence behavior. The examples of Sect. 4 will give more evidence to this fact.

2.3 The site-wise optimization measure: just a bit beyond monotonicity

The reason why both the epistasis measures and fitness distance correlation are sensitive to non-linear scaling is to be found in their reference classes. In this section, we construct a class which is a generalization of both the first order functions and the linear monotone functions, and we propose an associated measure which is insensitive to scaling by strictly monotone increasing fitness functions.

In the appendix we prove that f is a first order function if and only if for all $\beta \in \Sigma$, $\alpha \in \Sigma \setminus \{\beta\}$ and $i \in L$, we have that $f(s[i|\beta]) - f(s[i|\alpha])$ is a constant independent of $s \in S$. This implies that a first order function can be characterized by exactly $\ell(|\Sigma| - 1) + 1$ parameters. Now, for the purposes of GA behavior prediction, why would we need that the fitness value difference is always constant? Would it not be sufficient for a genetic algorithm with ranking selection that the difference always has the same sign? Taking this into account, we propose a reference class denoted SWD, the class of *site-wise decidable* functions, satisfying Eq. 8. One reads the equation as: for each site there exists a symbol that we can substitute in every string to possibly improve the fitness value of the string:

$$\exists s^* \in S \cdot \forall t \in S, \forall i \in L \cdot f(t[i|s_i^*]) \geq f(t). \tag{8}$$

It is easy to verify that all first order functions are site-wise decidable. One obvious generalization of fitness distance correlation's reference class one could come up with is the class of the monotone functions, replacing the strict linearity condition, by the one-directional condition "if one comes closer to the optimum, the fitness has to increase".

Before we investigate the relationship between site-wise decidable and monotone functions, let us first consider the *site-wise optimization* algorithm. It has been introduced as the bit-setting optimization algorithm in (Wilson, 1991).

Input: a fitness function $f : S \to R$ and a string $s \in S$.

Algorithm: Construct a string $m \in S$ by performing the following steps for each site $i \in L$:

- Construct the set

$$A_i = \{ \alpha \in \Sigma \ ; \ f(s[i|\alpha]) \geq f(s[i|\beta]) \ \text{for all } \beta \in \Sigma \}$$

- If $s_i \in A_i$, then assign $m_i := s_i$. Otherwise, assign m_i to an arbitrary element of A_i.

Output: The string m, denoted SWO(f, s).

Let us call a function f *site-wise optimizable* (denoted $f \in$ SWO) if the output of this algorithm is optimal regardless of the input string. It follows that f is site-wise

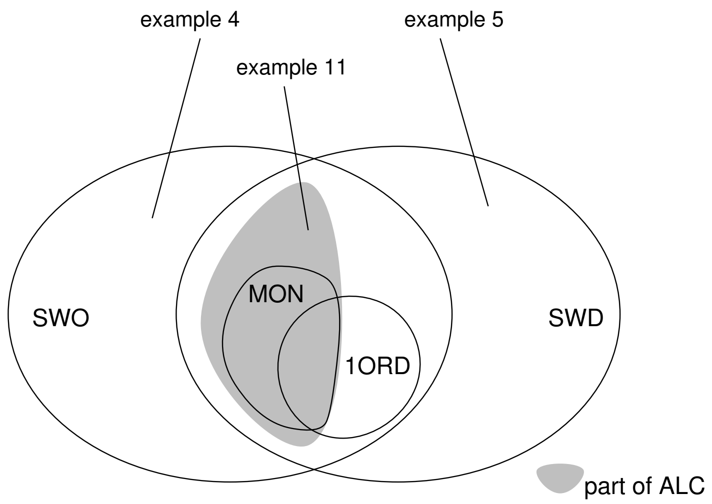

Figure 2: The position of the first order and monotone functions within the classes of site-wise optimizable and site-wise decidable functions. Propositions 7, 8, 9, 10 and examples 4, 5, 11 of the appendix prove the inclusions of this diagram. Note that the absence of local constantness also separates the site-wise optimizable functions with one global optimum from those with more (i.e., the shaded area also represents $\mathsf{SWO} \cap \mathsf{UGO}$).

optimizable and has a unique global optimum if and only if there is a $m \in S$ such that the output of the site-wise optimization algorithm is m regardless of the input string s.

The close relationship between site-wise optimizable and site-wise decidable functions is best understood by introducing yet another property that fitness functions might satisfy

$$\forall s \in S \cdot \forall i \in L \cdot \exists \alpha \in \Sigma \cdot \forall \beta \in \Sigma \setminus \{\alpha\} \cdot f(s[i|\alpha]) > f(s[i|\beta]). \tag{9}$$

Let us call this property the *absence of local constantness* (ALC). Figure 2 shows the relations between most of the function classes introduced so far. The proofs of the inclusions can be found in the appendix.

We propose the site-wise optimizable fitness functions with a unique global optimum as a generalization of the reference classes of epistasis and fitness distance correlation measures. They do not possess local constantness, which caused the approximation problems of epistasis measures. Moreover, they are only a bit more general than the monotone functions.

We will now construct a distance measure between a fitness function and the class of site-wise optimizable functions with unique global optimum, using the fact that these functions can be optimized by the site-wise optimization algorithm. (This measure was first described in (Naudts et al., 1997), where it was denoted τ, the bit decidability measure).

Let us define the *width* of a population (or sample) P to be the average Hamming distance between its members, i.e.,

$$w(P) = \frac{1}{|P|^2} \sum_{p,q \in P} \mathrm{d}(p, q). \tag{10}$$

In the appendix we show that the width of the search space S equals $\ell/|\Sigma|$. If we denote by $\mathrm{SWO}(f, P)$ the population formed by applying the site-wise optimization algorithm on the members of P individually, then $w(P) = 0$ for all samples P if and only if f has a unique global optimum and is site-wise optimizable. A natural way to define the *site-wise optimization measure* is now $\mathsf{swo}(f) = \mathsf{swo}_S(f)$, where

$$\mathsf{swo}_P(f) = \frac{w(\mathrm{SWO}(f, P))}{w(P)}. \tag{11}$$

The range of the (exact, not approximated) site-wise optimization measure is the closed interval [0, 1]. A value of 1 is obtained, amongst others, by constant fitness functions. This has a trivial interpretation: there is no information for the site-wise optimization algorithm to map a string s to anything else than s itself, i.e., the values of the sites cannot be decided at all. As a second improvement on the epistasis measures and fitness distance correlation, we have that the site-wise optimization measure is invariant to non-linear fitness function scaling, i.e., $\mathsf{swo}(g(f)) = \mathsf{swo}(f)$ for all strictly monotone increasing functions g. As a consequence, $\mathsf{swo}(\mathrm{NLM}_n) = 0$ for all n.

Exact site-wise optimization values for the Royal Road functions are easily obtained, see (Naudts et al., 1997). They only depend on the block length (and *not* on the number of blocks, which is in proportion to the string length). The approximation quality is very good.

3 GA-Easiness Beyond Monotonicity

3.1 Introduction

In the previous section we described a number of so called GA-hardness measures in detail. Recall that it is our main objective to show that the values produced by these measures can be entirely unrelated to the convergence behavior of the genetic algorithm. We have already demonstrated that fitness distance correlation and the epistasis measures can take any value of their range when presented with monotone fitness functions which all share the same convergence behavior (convergence quality class 1 according to the classification of Sect. 1). In order to find other such function classes, and to test the site-wise optimization measure, we will have to look beyond the monotone fitness functions.

An other reason for looking beyond monotonicity is that even the class of monotone fitness functions is too small to play the role of a reference class: there exist many other fitness functions for which the GA shows quality class 1 behavior.

In Sect. 3.2 and 3.3, we present two larger fitness function classes related to GA-easiness: the unimodal (or steepest-ascent optimizable) functions and the schema-GA-easy (or non-deceptive) functions. Section 3.4 presents a subclass of the unimodal functions, called the *restricted steepest-ascent optimizable* functions. This subclass will be used to draw examples from which again fool the predictive quality of all the measures of the previous section.

From this point on, we will assume that all fitness functions are in UGO, i.e., they have a unique global optimum. Figure 3 sketches the relationship between the site-wise optimizable functions and the three function classes introduced in the rest of this section.

3.2 Steepest-ascent optimization and unimodal functions

The steepest-ascent optimization algorithm (see, e.g., (Wilson, 1991)) is the standard implementation of a hill climber on the space Σ^L. It is defined analogously to the site-wise optimization algorithm:

Input: a fitness function $f : S \to R$ and a string $s \in S$.

Algorithm: Modify the string s by repeatedly performing the following steps:

- Record, for all sites i and symbols α, the triples

$$(i, \alpha, f(s[i|\alpha]) - f(s))$$

- Assign $s_i = \alpha$ where i and α are chosen such that the third component of the triple (i, α, x) reaches the highest possible value (this is the steepest ascent).

– Stop when no triple with strictly positive third component has been found.

Output: The string s, denoted $\mathrm{SAO}(f, s)$.

Note that this algorithm always stops after a finite number of changes in s. We call a fitness function $f : S \to R$ *steepest-ascent optimizable* if the application of this algorithm leads to the optimum of f regardless of the input string.

It is easy to verify that a fitness function f is steepest-ascent optimizable if and only if it is unimodal. One also finds that all site-wise optimizable functions are steepest-ascent optimizable, as in (Wilson, 1991). The same does not apply to site-wise decidable functions, which is shown in the appendix. To complement Wilson's example of a steepest-ascent but not site-wise optimizable function, the appendix also shows examples of fitness functions in $(\mathsf{UNI} \setminus \mathsf{SWO}) \cap \mathsf{ALC}$ and $(\mathsf{UNI} \setminus \mathsf{SWO}) \setminus \mathsf{ALC}$.

Note that it is far from sure that all steepest-ascent optimizable functions belong to the first convergence quality class, although many are. They certainly are contained in the optimal convergence quality class of simulated annealing (see, e.g., (van Laarhoven & Aarts, 1987)), which is essentially a more robust (w.r.t. local optima) version of the steepest-ascent optimization algorithm.

It is worth remarking that there also exist easy functions at the other end of the modality spectrum. Horn and Goldberg (1995) show that there exists easy yet maximally multi-modal fitness functions.

3.3 Schema-GA-easy functions

The term *GA-easy*, which we refer to as *schema-GA-easy*, originated in the study of deception and schema competitions, started mainly in (Goldberg, 1987). Relevant in this context are, amongst others, (Whitley, 1991), (Das & Whitley, 1993), (Liepins & Vose, 1991) and (Grefenstette, 1993). Briefly, a fitness function f is called schema-GA-easy whenever the winners of the competition of any order k less than ℓ are contain the global optimum of f.

It is straightforward to prove that all monotone fitness functions are schema-GA-easy. An investigation of the statements $\mathsf{SWD} \subset \mathsf{schema\text{-}GA\text{-}easy}$ and $\mathsf{SWO} \subset \mathsf{schema\text{-}GA\text{-}easy}$ requires a more precise definition of schema-GA-easiness: we could, e.g., state that no competitions can remain undecided, or that the winners can only lead to one unique optimum instead of many different ones. In this way, all first order functions could be seen as schema-GA-easy, or only the subset $\mathsf{1ORD} \cap \mathsf{ALC}$. For the remainder of this text, we say that a function is schema-GA-easy when all competitions (of any order) are decided in favor of the same optimum. In the appendix we prove that $\mathsf{SWO} \cap \mathsf{UGO} \subset \mathsf{schema\text{-}GA\text{-}easy}$.

It is shown in (Wilson, 1991) that schema-GA-easy fitness functions are not necessarily steepest-ascent optimizable. A function which is not schema-GA-easy is called *deceptive* of some sort and order. The appendix shows a simple example of a deceptive function which is steepest-ascent optimizable.

Vose and Wright (1995) present the extreme case of a (fully) non-deceptive function which is difficult for the GA, showing that schema-GA-easiness does not imply GA-easiness. Depending on the choice of mutation rate, this function belongs

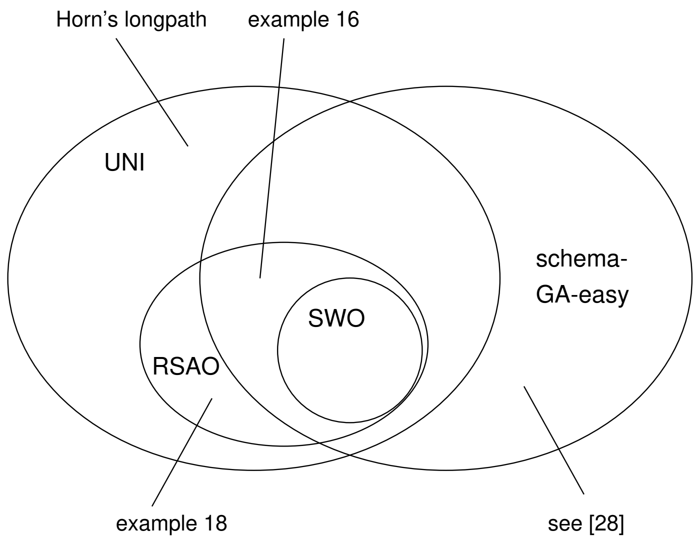

Figure 3: Pictorial description of the relationship between four classes related to GA-easiness in the case of functions with unique global optimum. In the intersection of the unimodal and schema-GA-easy functions we find the class of site-wise optimizable functions, itself containing the monotone functions. The restricted steepest ascent optimizable functions are trivially contained in the unimodal functions. Propositions 14, 17 and examples 16, 18 in the appendix prove the inclusions of this diagram. The longpath problem presented in (Horn, Goldberg, & Deb, 1994) is an example of a unimodal function which is not restricted steepest ascent optimizable nor schema-GA-easy.

to quality classes 2 or 3.

3.4 Restricted steepest-ascent optimizable functions

The restricted steepest-ascent optimization algorithm is a restricted version of the steepest-ascent optimization algorithm in the sense that, once the value at a site has been modified, it cannot be modified again. It follows that this algorithm performs not more than ℓ steps. It can be considered as a compromise between site-wise and steepest-ascent optimization.

We call a fitness function f restricted steepest-ascent optimizable, and we write $f \in \mathsf{RSAO}$, if and only if this algorithm reaches to optimum independently of the input string s. Obviously, SWO $\subset$ RSAO $\subset$ UNI.

All restricted steepest-ascent optimizable functions we have come across so far, and certainly those in Sect. 4.1 and 4.3, showed high convergence speed and optimal convergence quality. We believe this is a general characteristic of the class.

4 Instructive examples

We use the popular name *onemax* to denote the problem of optimizing the number of ones in the bitstring, i.e., the problem of maximizing $f : s \mapsto \sum_{i \in L} s_i$. We define *zeromax* analogously.

4.1 Deceptive mixtures of onemax and zeromax

By mixing a onemax and a number of zeromax problems we obtain a class of deceptive yet restricted steepest-ascent optimizable functions.

Let $\Sigma = \{0, 1\}$, $L = \{0, \ldots, \ell - 1\}$ and let n, with $0 < n < \ell$, be a control parameter. We construct a fitness function f_n by first putting an $\ell - 1$-site zeromax problem on the schema $0\#^{\ell-1}$, followed by an $\ell - 2$-site zeromax problem translated by 2ℓ on the schema $10\#^{\ell-2}$. We continue adding translated zeromax problems until an $\ell - n$-site zeromax problem, translated by $2(n-1)\ell$, is put on the schema $1^{n-1}0\#^{\ell-n}$. Finally, we put an $\ell - n$-site onemax problem on the schema $1^n\#^{\ell-n}$ and translate it by $2n\ell$.

As an example, the function f_4 is sketched in Fig. 4. In short,

$$f_4(0s_1 \dots s_{\ell-1}) = \text{ number of zeros in } s_1 \dots s_{\ell-1}$$

$$f_4(10s_2 \dots s_{\ell-1}) = \text{ number of zeros in } s_2 \dots s_{\ell-1}$$

$$f_4(110s_3 \dots s_{\ell-1}) = \text{ number of zeros in } s_3 \dots s_{\ell-1}$$

$$f_4(1110s_4 \dots s_{\ell-1}) = \text{ number of zeros in } s_4 \dots s_{\ell-1}$$

$$f_4(1111s_4 \dots s_{\ell-1}) = \text{ number of ones in } s_4 \dots s_{\ell-1}$$

Figure 5 shows that the functions f_n belong to the first convergence quality class; the GA can be said to converge fast on them when linear ranking selection is applied. Proportional selection exerts not enough pressure for the GA to converge. Yet, fitness distance correlation for all the values of n and the site-wise optimization measure for the first values of n are blinded by the large proportion of completely irrelevant deception. Both epistasis measures predict a near to first order function for low values of n, and stabilize to some – rather good – intermediate value for higher values of n. Note that none of the measures sees the convergence speed dropping with n increasing.

4.2 Nearest neighbor interaction functions

A one-dimensional nearest neighbor interaction (NNI) function f with cyclic boundary conditions is determined by 3ℓ constants, denoted by h_i (the second order components), and k_i and l_i (the first order components), for $i \in L$. For the sake of notation, we will assume that $h_\ell \equiv h_0$ and $s_\ell \equiv s_0$. The fitness value of a string

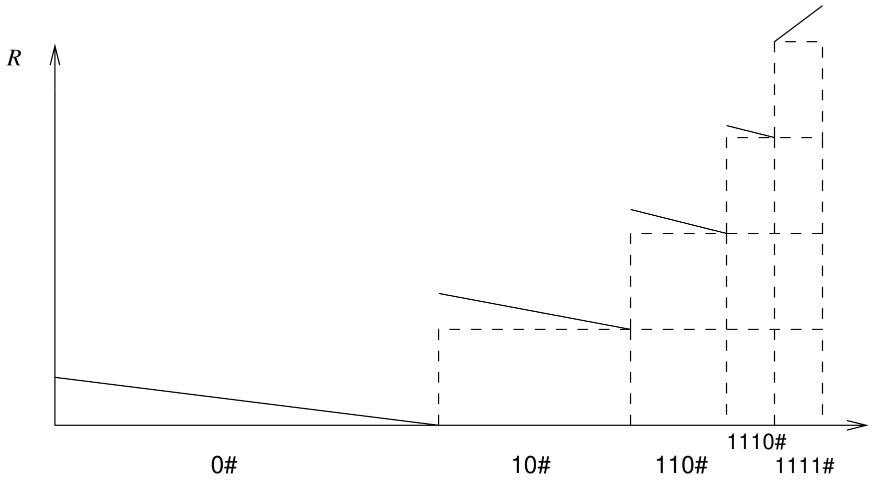

Figure 4: Pictorial description of the deceptive mixture of a onemax and 4 zeromax problems. The mixture of a onemax and four nearest neighbor problems (as in Sect. 4.3) is obtained by substituting the zeromax functions and adjusting the height of the translations according to the optimum of the nearest neighbor functions.

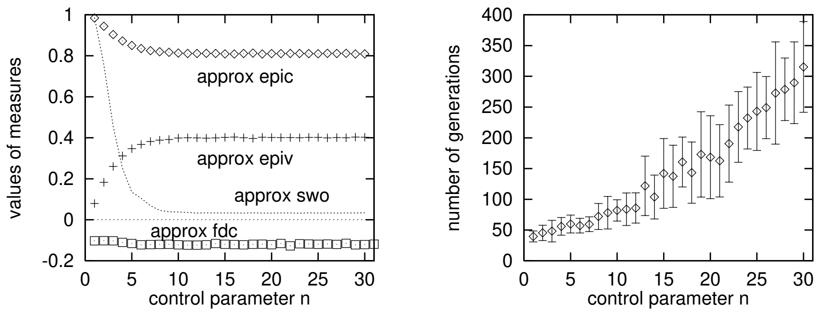

Figure 5: Epistasis variance, epistasis correlation, fitness distance correlation, site-wise optimization values and number of generations to optimum for the **deceptive mixtures of a onemax and *n* zeromax problems**. The string size is 80. The approximations are averaged over 21 values using a random sample of size 2^8 for swo and of 2^{12} for the other measures. See Sect. 1 for a description of the GA. The same GA with proportional (roulette-wheel) selection fails to detect the optimum within a reasonable length of time due to a too low selection pressure.

$s \in S$ is given by

$$f(s) = \sum_{i \in L} h_i(1 - s_i)s_{i+1} + \sum_{i \in L}(k_i s_i + l_i(1 - s_i)). \qquad (12)$$

The origin of these functions lies in statistical physics. The functions correspond to the instances of the anti-ferromagnetic one-dimensional nearest neighbor interaction model, whose energy functional is

$$H = -\sum_{i \in L} J_i \sigma_i \sigma_{i+1} - \sum_{i \in L} \tau_i \sigma_i. \qquad (13)$$

Here the interaction constants J_i take non-positive values, the external magnetic field strengths τ_i take arbitrary values and the spin variables σ_i take values in $\{-1, 1\}$. Functions of this form have been extensively studied; an overview can be found in (Mezard, Parisi, & Virasoro, 1987). Closer to our subject, Prügel-Bennett and Shapiro (1997) have made a theoretical study of the dynamics of the GA on simple random Ising systems, which are theoretically harder to deal with than the class we are considering, yet are close enough to our class to transfer the results.

It is important to realize that NNI problems can be optimized in linear time by a *divide and conquer* algorithm. It is also possible to calculate their optimum using analytical and numerical techniques from statistical physics.

We define a subclass of NNI problems by setting the first order component to zero and the values of the interaction component h_i to $1 + (a(i+1) \bmod b)$, for positive $a, b \in \mathbb{N}$. Functions in this class range from fast and perfect convergence – quality class 1 and high speed (see Sect. 1) – over fast but occasional convergence (quality class 3) to slow and occasional convergence (also quality class 3). Figure 6 shows the values of the four measures for a selection of 29 NNI problems from the class parametrized by a and b. It is clear that none of the measures can make a distinction between the different classes of quality of convergence. The influence of the selection operator on the performance of the GA is drastic. Instances whose optimum is occasionally found with proportional selection are found to be trivial for a GA with ranking selection, and vice versa.

We can make the NNI problems extremely hard for both the GA and simulated annealing by applying a technique we call *balancing*. We want to achieve that for each site i, the equation $h_{i-1} + 2k_i = h_i + 2l_i$ is satisfied. This can easily be accomplished by changing the first order component of the function. Translated to the traditional notations from statistical physics, balanced NNI's correspond to instances of the nearest neighbor interaction model without external magnetic field, i.e., with energy functional $H = -\sum_i J_i \sigma_i \sigma_{i+1}$. The GA is unable to detect the optimum of any of the balanced versions of the NNI problems presented in Fig. 6, no matter how long or how often it is run. We refer to (Jäckle, Stinchcombe, & Cornell, 1991) and (Prügel-Bennett & Shapiro, 1997) for the reasons behind this.

In the case of balanced NNI problems not shown in the figure, all measures except epistasis variance show extremely high values (in distance to their respective reference class), a fact which is not difficult to prove. Let us outline the proof for the case of the site-wise optimization measure. The expected contribution of a site of a given string equals half of the second order component of the previous site plus the complete first order component of the current site. This is because the probability that there is a 0 at the previous site equals the probability of there being a 1, in which case the second order component can never be included. If a problem is balanced, we thus have that the expected contribution of a site is independent of the value at that site. In the sample containing the strings modified by the site-wise optimization

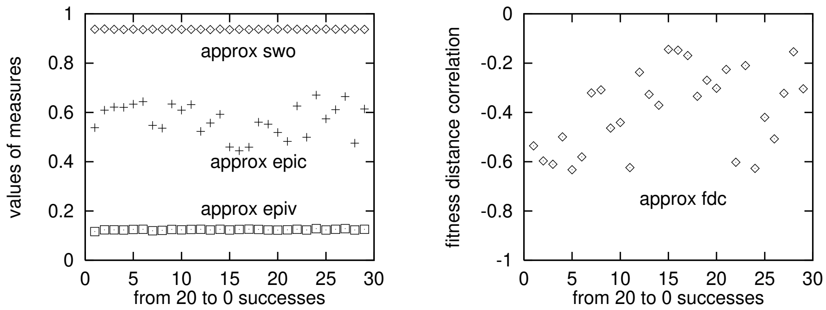

Figure 6: Epistasis variance, epistasis correlation, site-wise optimization values and fitness distance correlation for a **selection of 29 unbalanced nearest neighbor interaction problems** on 80 sites, as defined in Sect. 4.2. The problems are ordered according to the number of runs which reached the optimum. The success rate is optimal for the six leftmost problems, and is zero for the two rightmost. The approximations are averaged over 21 values using a random sample of size 2^8 for swo and of 2^{12} for the other measures. We refer to Sect. 1 for a description of the GA. Applying the same GA but with proportional selection results in a different ordering. This does not change the conclusions about the measures, however.

algorithm we then have that there are as many strings with a 0 on a given site as there are strings with a 1 on that site, which results in a site-wise optimization value of 1.

4.3 Mixtures of onemax and nearest neighbor interaction functions

The construction of these mixtures is similar to that of the deceptive mixtures defined in Sect. 4.1, and Fig. 4 can be used as a guide to it. In this section, we substitute the zeromax problems by balanced nearest neighbor interaction problem with $a = 5$ and $b = 7$. To ensure that the mixtures are restricted steepest ascent optimizable, we have to adjust the translations such that the fitness ranges of each schema involved in the definition are disjoint. This is not a problem since we can easily compute the optimum of the balanced problem on $\ell - n$ sites.

The substitution of the zeromax problems by the NNI problems (Fig. 7) only effects the values of the site-wise optimization measure, which stabilize to a high value of 0.983 with n increasing. Recall that in the previous case (mixture of a onemax and n zeromax problems in Fig. 5), when the control parameter n becomes greater than, say, 6, the site-wise optimization measure cannot be fooled anymore by the deceptive components.

Figure 7 also shows that the new mixtures still belong to the first convergence quality class (as defined in 1); the GA can be said to converge fast on them when linear ranking selection is applied. Proportional selection exerts not enough pressure for the GA to converge. Fitness distance correlation and the site-wise optimization measure are blinded by the ruggedness of the landscape which is completely ignored by the GA concentrating on the translations.

Again, as in the case of Sect. 4.1, the epistasis measures that stabilize to some – rather good – intermediate value. Also similar is that none of the measures sees the convergence speed dropping with n increasing.

Note that the epistasis variances obtained for the mixture of onemax and NNI problems (around 0.4) in Fig. 7 are even higher than those obtained for the NNI problems (around 0.1) in Fig. 6, although the former problems are a lot easier for a GA to solve than the latter ones (quality 1 vs quality 3 and 4). The same phenomenon of misleading prediction is observed with fitness distance correlation. The two other measures, epistasis correlation and the site-wise optimization measure, hardly see any difference between the two cases.

5 Related work

This section relates the results of the previous sections to another attempt at improving the epistasis measure, and proposes some guidelines for further work.

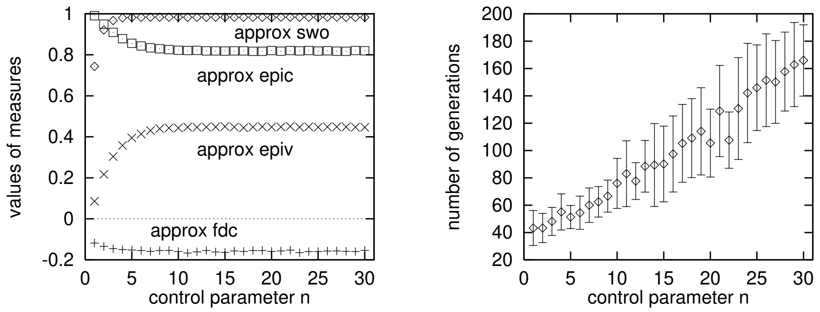

Figure 7: Epistasis variance, epistasis correlation, fitness distance correlation, site-wise optimization values and number of generations to optimum for **mixtures of a onemax and *n* balanced nearest neighbor interaction functions**. The string size is 80. The approximations are averaged over 21 values using a random sample of size 2^8 for swo and of 2^{12} for the others. See Sect. 1 for a description of the GA. As with the other mixtures, the same GA with proportional (roulette-wheel) selection fails to detect the optimum within a reasonable length of time due to a too low selection pressure.

5.1 The Experimental Design perspective

Reeves and Wright's Anova tables (Reeves & Wright, 1995) contain a lot more information about the epistatic structure of a fitness function than the epistasis measures do. In short, these tables also provide, apart from the deviation from the first order functions, also information about the higher order interactions present in the fitness function.

While this technique gives more quantitative information about the epistatic structure of the fitness function, and certainly has to be preferred over epistasis variance or correlation for that matter, the question remains whether there is a correspondence between the epistasis information and the convergence behavior of the genetic algorithm. As can be deduced from our experiments with nearest neighbor interaction problems, it is not the amount of interaction which influences the convergence. Rather, one has to search for the presence of symmetry or frustration within the interaction. The latter phenomena already occur in second order interaction problems. We refer to (Naudts & Verschoren, 1998) for a more detailed discussion of second order interaction.

Interaction does not necessarily have to have a negative effect on the convergence behavior of the GA neither. We refer to the NLM_n functions defined in Sect. 2.1.3: the more interactions we add (i.e., the higher the exponent n), the faster the GA with proportional selection converges.

We conclude that quantitative information about higher order interactions is not a priori related to the convergence behavior of the GA.

5.2 Random sampling versus Metropolis sampling

It is very easy to make up fitness functions which yield close to minimal observed values for the measures, yet are very hard to optimize for the GA. Consider the onemax problem modified by inserting a balanced nearest neighbor interaction problem in the subspace $1^{(n)}\#^{(\ell-n)}$, for some n and ℓ large enough. Random samples of size less than 2^n will typically only carry information about the onemax component of the fitness function, which results in minimal observed values for all measures presented in this paper (and any other measures based on random samples).

Using the Metropolis algorithm, which is the basic algorithm behind simulated annealing, it is possible to construct samples of the search space with the property that the mean fitness value of the sample equals a given value f_β (where β is the inverse temperature; it suffices to know that theoretically there is a one-to-one correspondence between temperature and average values). In this way, we can obtain information about the structure of the fitness landscape as the GA sees it when the average fitness value of its population has already surpassed the average fitness value of the function.

Note that this technique also allows for an estimation of the number of strings which share a given fitness value. From these data, the *density of states* (Rosé, Ebeling, & Asselmeyer, 1996) of a fitness function is computed. It is presented as a absolute optimization hardness measure.

Similar to the technique of obtaining information at a higher fitness value level is the technique of computing the values of the measures using the information contained in successive generations of a GA run. This typically gives better insight in the structure of the search space than, e.g., the exact values of the measures, as the latter can be biased by a large and completely irrelevant part of the search space.

Conclusions

We have shown in this paper that the values produced by epistasis variance, epistasis correlation and fitness distance correlation cannot reliably be related to the convergence behavior of the genetic algorithm. Moreover, even if we perform the necessary steps to make these measures more robust with respect to non-linear scaling, we can still find fitness functions which are very easy for the GA and whose site-wise optimization values can be anywhere in the range of this measure. We conclude that these measures do not deserve the name GA-hardness measures, mainly because they are too weak to detect a host of functions which are easy for the GA. Actually, epistasis variance and correlation, fitness distance correlation and the site-wise optimization reliably detect first order functions resp. linear and monotone functions, when presented with the relevant information. The behavior of the GA on any of these fitness functions is identical, well understood and can be predicted. Presenting the relevant information to the measures is of primordial importance.

The key problem in designing a summary statistic is to find a characterization of a broad class of fitness functions on which the GA behaves similarly. Clearly, monotonicity is too small to satisfy this requirement. Note that finding a good definition of "similar behavior" is also a major problem, which we will try to address in the near future.

The main results of this paper hold for GAs with ranking and proportional selection schemes. The tremendous influence of selection pressure on the convergence behavior of the GA is a strong limitation on the potential predictive quality of any a priori measure.

We have tried to base our conclusions on mathematical proofs in the appendix. Otherwise, the examples used were as representative as possible of the classes of problems in study, and the results obtained were independent of string length and GA parameters used.

Acknowledgements

The authors are grateful to their PhD supervisors, Alain Verschoren at the University of Antwerp and Marc Schoenauer at the Ecole Polytechnique. Their continuing support has made this joint work possible.

References

Altenberg, L. (1997). Fitness distance correlation analysis: an instructive counterexample. In T. Bäck (Ed.), *Proceedings of the 7th international conference* *on genetic algorithms* (p. 57-64). Morgan Kaufmann Publishers.

Das, R. & Whitley, D. (1993). The only challenging problems are deceptive: Global search by solving order-1 hyperplanes. In S. Forrest (Ed.), *Proceedings of the 5th international conference on genetic algorithms* (p. 166-173). Morgan Kaufmann Publishers.

Davidor, Y. (1991a). Epistasis variance: a viewpoint on GA-hardness. In G. Rawlins (Ed.), *Foundations of genetic algorithms* (p. 23-35). Morgan Kaufmann Publishers.

Davidor, Y. (1991b). *Genetic algorithms and robotics: a heuristic strategy for optimization.* Signapore: World Scientific Publishing Co. (Pte.) Ltd.

Goldberg, D. (1987). Simple genetic algorithms and the minimal deceptive problem. In D. L. (Ed.), *Genetic algorithms and simulated annealing* (p. 74-88). Morgan Kaufmann Publishers.

Grefenstette, J. (1993). Deception considered harmful. In L. Whitley (Ed.), *Foundations of genetic algorithms 2* (p. 75-92). Morgan Kaufmann.

Horn, J. & Goldberg, D. (1995). Genetic algorithms difficulty and the modality of fitness landscapes. In L. Whitley & M. Vose (Eds.), *Foundations of genetic algorithms 3* (p. 243-269). Morgan Kaufmann Publishers.

Horn, J., Goldberg, D., & Deb, K. (1994). Long path problems. In Y. Davidor, H.-P. Schwefel, & R. Manner (Eds.), *Proceedings of the 3rd conference on parallel problems solving from nature* (p. 149-158). Springer Verlag.

Jäckle, J., Stinchcombe, R., & Cornell, S. (1991). Freezing of nonequilibrium domain structures in a kinetic Ising model. *J. Stat. Phys.*, *62*, 425-433.

Jones, T. & Forrest, S. (1995). Fitness distance correlation as a measure of problem difficulty for genetic algorithms. In L. Eshelman (Ed.), *Proceedings of the 6th international conference on genetic algorithms* (p. 184-192). Morgan Kaufmann Publishers.

Kallel, L. & Naudts, B. (1998). *A longpath problem for the GA.* (Submitted)

Kallel, L. & Schoenauer, M. (1996). *Fitness distance correlation for variable length representations* (Tech. Rep. No. 363). Ecole Polytechnique: CMAP.

Liepins, G. & Vose, M. (1991). Deceptiveness and genetic algorithm dynamics. In G. Rawlins (Ed.), *Foundations of genetic algorithms* (p. 36-52). Morgan Kaufmann Publishers.

Manela, M. & Campbell, J. (1992). Harmonic analysis, epistasis and genetic algorithms. In R. Manner & B. Manderick (Eds.), *Proceedings of the 2nd conference on parallel problems solving from nature* (p. 57-64). North Holland.

Mezard, M., Parisi, G., & Virasoro, M. (1987). *Spin glass theory and beyond.* Singapore: World Scientific Press.

Mitchell, M., Forrest, S., & Holland, J. (1993). The Royal Road for genetic algorithms: Fitness landscapes and GA performance. In F. Varela & P. Bourgine (Eds.), *Proceedings of the first european conference on artificial life-93* (p. 245-254). MIT Press/Bradford Books.

Naudts, B., Suys, D., & Verschoren, A. (1997). Epistasis as a basic concept in formal landscape analysis. In T. Bäck (Ed.), *Proceedings of the 7th international conference on genetic algorithms* (p. 65-72). Morgan Kaufmann Publishers.

Naudts, B. & Verschoren, A. (1998). SGA search dynamics on second order functions. In J.-K. Hao, E. Lutton, E. Ronald, M. Schoenauer, & D. Snyers (Eds.), *Artificial evolution 97.* Springer Verlag.

Prügel-Bennett, A. & Shapiro, J. (1997). The dynamics of a genetic algorithm for simple Ising systems. *Physica D*, *104*, 75-114.

Reeves, C. & Wright, C. (1995). An experimental design perspective on genetic algorithms. In L. Whitley & M. Vose (Eds.), *Foundations of genetic algorithms 3* (p. 7-22). Morgan Kaufmann Publishers.

Rochet, S., Slimane, M., & Venturini, G. (1996). Epistasis for real encoding in genetic algorithms. In V. Narasimhan & L. Jain (Eds.), *Proceedings of the ieee anziis 96.*

Rochet, S., Slimane, M., & Venturini, G. (1997). *A critical study of epistasis measures.* (In preparation)

Rochet, S., Venturini, G., Slimane, M., & Kharoubi, E. E. (1998). A critical and empirical study of epistasis measures for predicting GA performances: a summary. In J.-K. Hao, E. Lutton, E. Ronald, M. Schoenauer, & D. Snyers (Eds.), *Artificial evolution 97.* Springer Verlag.

Rosé, H., Ebeling, W., & Asselmeyer, T. (1996). The density of states - a measure of the difficulty of optimization problems. In H.-M. Voigt, W. Ebeling,

I. Rechenberg, & H.-P. Schwefel (Eds.), *Proceedings of the* 4^{th} *conference on parallel problems solving from nature* (Vol. 1141, p. 208-217). Springer Verlag.

van Laarhoven, P. & Aarts, E. (1987). *Simulated annealing: theory and applications.* Dordrecht, The Netherlands: Kluwer Academic Press.

Vose, M. & Wright, A. (1995). Stability of vertex fixed points and applications. In L. Whitley & M. Vose (Eds.), *Foundations of genetic algorithms 3* (p. 103-114). Morgan Kaufmann Publishers.

Whitley, D. (1991). Fundamental principles of deception in genetic search. In G. Rawlins (Ed.), *Foundations of genetic algorithms* (p. 221-241). Morgan Kaufmann Publishers.

Wilson, S. (1991). GA-easy does not imply steepest-ascent optimizable. In R. Belew & L. Booker (Eds.), *Proceedings of the 4th international conference on genetic algorithms* (p. 85-89). Morgan Kaufmann Publishers.

Appendix

Definition 1 (Noise-to-signal epistasis measure). Define, for all samples P, sites $i \in L$ and symbols $\alpha \neq \beta \in \Sigma$,

$$\sigma(P, i, \alpha, \beta) = \frac{1}{|P|} \sum_{s \in P} |f(s[i|\alpha]) - f(s[i|\beta])| \,, \tag{14}$$

$$n(P, i, \alpha, \beta) = \frac{1}{|P|-1} \sum_{s \in P} \big(|f(s[i|\alpha]) - f(s[i|\beta])| - \sigma(P, i, \alpha, \beta)\big)^2, \tag{15}$$

$$\nu(P, i, \alpha, \beta) = \begin{cases} \dfrac{\sqrt{n(P, i, \alpha, \beta)}}{\sigma(P, i, \alpha, \beta)} & \text{if } \sigma(P, i, \alpha, \beta) \neq 0, \\ 0 & \text{otherwise,} \end{cases} \tag{16}$$

$$\mathsf{nts}_P(f) = \frac{2}{\ell|\Sigma|(|\Sigma|-1)} \sum_{i \in L} \sum_{\alpha < \beta \in \Sigma} \nu(P, i, \alpha, \beta). \tag{17}$$

The *noise-to-signal epistasis measure* of a fitness function f can then be defined as $\mathsf{nts}(f) = \mathsf{nts}_S(f)$. It follows trivially that $\mathsf{nts}(f) = 0$ if and only if f is a first order function. Note that this measure is insensitive to translation and linear scaling, i.e., $\mathsf{nts}(af + b) = \mathsf{nts}(f)$, for any $a, b \in R$. The choice made in Equation 16 is responsible for the bad behavior on near constant functions.

Example 2 (Constantness problem of epistasis measures). The slightly generalized Royal Road functions (Mitchell et al., 1993) are defined for $\Sigma = \{0, 1\}$ and $L = \{0, \ldots, 2^n - 1\}$, with $n > 1$. Define, for each $n > 1, m \leq n$ and $0 \leq i < 2^{n-m}$, the schema

$$H_i^{n,m} = \#^{(2^m i)} 1^{(2^m)} \#^{(2^n - 2^m(i+1))}. \tag{18}$$

Then we construct, for all $m \leq m' \leq n$, the Royal Road function

$$\mathrm{RR}(n, m, m')(s) = \sum_{k=m}^{m'} 2^k |\{i \ ; \ s \in H_i^{n,k}\}| \qquad \text{for all } s \in S. \tag{19}$$

Given the fact that epistasis variance with L^2-normalization cannot evaluate a constant zero function, we translate the Royal Road functions by 0.01 before carrying out the approximations. This introduces a small but negligible difference between the exact and approximated values. Close to constantness, exact and approximated values of the normalized epistasis variance of the Royal Road functions cease to agree. This is shown in Fig. 8. Note that the figure also shows a disagreement between the exact and approximated epistasis correlation values. Although the algorithm returns zero when it finds the variance of the fitness function or the variance of its linear approximation equal to zero, it does not do so when these values are close to zero (unless the fitness function is normalized, one cannot know what *close to zero* means). In this case, the product of the two variances, which is of order 10^{-31}, closely resembles the covariance, resulting in a correlation coefficient of 1.

Proposition 3. Let f be a fitness function. Then

$$f \in \mathsf{1ORD} \iff \text{if } \beta \in \Sigma, \text{ then } f(s[i|\beta]) - f(s[i|\alpha]) \text{ is independent of } s \in S \text{ for all } i \in L \text{ and } \alpha \in \Sigma \setminus \{\beta\}.$$

Proof. Let f be in 1ORD, i.e., there exist $g_i : \Sigma \to R$, with $i \in L$, such that $f(s) = \sum_{i \in L} g_i(s_i)$ for all $s \in S$. Let β be in Σ. Then for any $s \in S$, any $i \in L$ and $\alpha \in \Sigma \setminus \{\beta\}$,

$$f(s[i|\beta]) - f(s[i|\alpha]) = g_i(\beta) - g_i(\alpha), \tag{20}$$

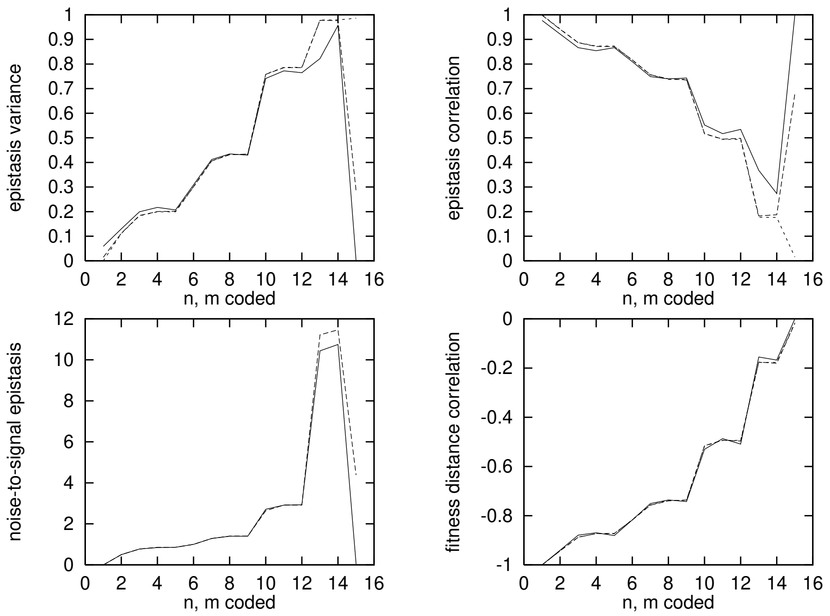

Figure 8: Epistasis variance, epistasis correlation, noise-to-signal epistasis and fitness distance correlation of Royal Road functions with 16 sites. The full line indicates the sample size 2^8 approximations, the dashed line the sample size 2^{12} approximations, and the dotted line the exact values. Note that the latter has not been computed for the noise-to-signal epistasis measure. On the x-axis, the 15 Royal Road functions are ordered as follows (values for m and m'): (0,0), (0,1), . . . , (0,4), (1,1), (1,2), . . . , (1,4), . . . , (4,4).

which is independent of s. To prove the other implication, we will first construct the g_i maps. Choose $\beta \in \Sigma$, $j \in L$ and $s \in S$, let $b = \beta\beta \ldots \beta$, and set

$$\delta_i^\alpha = f(s[i|\alpha]) - f(s[i|\beta]) \tag{21}$$

for all $i \in L$ and $\alpha \in \Sigma \setminus \{\beta\}$. Then define

$$g_j(\beta) = f(b) \tag{22}$$

$$g_j(\alpha) = f(b) + \delta_j^\alpha \qquad \text{for all } \alpha \in \Sigma \setminus \{\beta\}, \tag{23}$$

$$g_i(\beta) = 0 \qquad \text{for all } i \in L \setminus \{j\}, \tag{24}$$

$$g_i(\alpha) = \delta_i^\alpha \qquad \text{for all } \alpha \in \Sigma \setminus \{\beta\} \text{ and } i \in L \setminus \{j\}. \tag{25}$$

We prove that $f(s) = \sum_{i \in L} g_i(s_i)$ by induction on the Hamming distance between s and b. The base of the induction, Hamming distance zero, follows immediately from the construction. Let us now assume that $f(t) = \sum_{i \in L} g_i(t_i)$ for all t at Hamming distance $k < \ell$ from b, and take $s \in S$ such that $d(b, s) = k + 1$. Then we pick a $k \in L$ such that $s_i \neq \beta$, and we have

$$\begin{aligned}
\sum_{i \in L} g_i(s_i) &= f(b) + \sum_{i \in L \cdot s_i \neq \beta} \delta_i^{s_i} && (26)\\
&= \big(f(b) + \sum_{i \in L \cdot\, i \neq k \wedge s_i \neq \beta} \delta_i^{s_i}\big) + \delta_k^{s_k} && (27)\\
&= f(s[k|\beta]) + \delta_k^{s_k} && (28)\\
&= f(s),
\end{aligned}$$

which completes the proof. $\square$

Example 4 (SWO $\not\subseteq$ SWD). Let $\Sigma = \{0, 1\}$ and $L = \{0, 1, 2\}$, and choose values for f such that

$$f(000) = f(100) = f(110) = f(111), \tag{29}$$

$$f(011) < f(001) < f(101) < f(010) < f(000). \tag{30}$$

Clearly f is site-wise optimizable:

$$\mathrm{SWO}(f, 010) = \mathrm{SWO}(f, 001) = \mathrm{SWO}(f, 011) = 100 \tag{31}$$

and $\mathrm{SWO}(f, 101) = 110$. But it is not site-wise decidable, for $f(010) < f(000)$ and $f(101) < f(111)$.

Example 5 (SWD $\not\subseteq$ SWO). Let $\Sigma = \{0, 1\}$ and $L = \{0, 1\}$, and choose values for f such that $f(00) > f(01) > f(10) = f(11)$. The conditions for site-wise decidability are satisfied, but $\mathrm{SWO}(f, 11) = 01$, which is suboptimal.

Proposition 6. SWD $\cap$ ALC $\subset$ SWO.

Proof. Assume that there exist $s^1, s^2 \in S$ such that $m^1 = \mathrm{SWO}(f, s^1)$ and $m^2 = \mathrm{SWO}(f, s^2)$ are different. Let i be one of the sites in which they differ. According to the the algorithm, we have (Eq. 9 is responsible for the strict inequalities)

$$f(s^1[i|m_i^1]) > f(s^1[i|m_i^2]) \qquad \text{and} \tag{32}$$

$$f(s^2[i|m_i^2]) > f(s^2[i|m_i^1]), \tag{33}$$

which contradicts the fact that f is site-wise decidable. □

Proposition 7. SWO $\cap$ ALC $\subset$ UGO.

Proof. By induction on ℓ, the number of sites. Assume first that $\ell = 2$, e.g., $L = \{0, 1\}$. Suppose f is multi-modal, with optima at s^1 and s^2. The ALC membership forces the optima to be at Hamming distance 2 from each other, i.e., $s^1_0 \neq s^2_0$ and $s^1_1 \neq s^2_1$. We also have that $f(s^1_0 s^2_1)$ is strictly less than $f(s^1)$. Hence $\mathrm{SWO}(f, s^2_0 s^1_1) = s^1_0 s^2_1$ cannot be optimal, which is in contradiction with $f \in \mathsf{SWO}$. This proves the base of the induction.

The induction step is proved similarly. We are given that all fitness functions $g \in \mathsf{SWO} \cap \mathsf{ALC}$ defined on ℓ sites are unimodal. Let $f \in \mathsf{SWO} \cap \mathsf{ALC}$ be defined on $\ell + 1$ sites, and assume it is multi-modal. There can only be two optima, because all optima have to be at Hamming distance $\ell + 1$ from each other. (This follows from the induction hypothesis.) Let s^1 and s^2 be these optima, with $s^1_i \neq s^2_i$ for all $i \in L$. Fix $i \in L$. Again we have $f(s^1[i|s^2_i]) < f(s^1)$, as $f \in \mathsf{ALC}$, and we have a contradiction because $\mathrm{SWO}(f, s^2[i|s^1_i]) = s^1[i|s^2_i]$ which is suboptimal. (The latter equality follows from the induction hypothesis: if the value at site i equals s^1_i, then the algorithm maps the value of site $j \in L \setminus \{i\}$ to s^1_j.) $\square$

Proposition 8. $\mathsf{SWO} \cap \mathsf{ALC}^C \subset \mathsf{UGO}^C$.

Proof. Let f be in SWO and not in ALC. It follows from the latter fact that there exist $s \in S$, $i \in L$ and $\alpha \neq \beta \in \Sigma$ such that

$$f(s[i|\gamma]) \leq f(s[i|\alpha]) = f(s[i|\beta]) \tag{34}$$

for all $\gamma \in \Sigma$. As a consequence, $\mathrm{SWO}(f, s[i|\alpha])$ will have value α at site i, and $\mathrm{SWO}(f, s[i|\beta])$ value β at that site. The outcomes are optimal and different. $\square$

Example 5 shows that there exist unimodal $f \in \mathsf{ALC}^C$.

Proposition 9. $\mathsf{MON} \subset \mathsf{SWD}$.

Proof. Let s^* be the unique global optimum of the monotone fitness function f. Let $t \in S$ and $i \in L$. We have to show that $f(t[i|s_i^*]) \geq f(t)$. When $s_i^* = t_i$, this is the case. The $s_i^* \neq t_i$, then $\mathrm{d}(t[i|s_i^*], s^*) < \mathrm{d}(t, s^*)$, from which $f(t[i|s_i^*]) \geq f(t)$ follows. $\square$

Proposition 10. $\mathsf{MON} \subset \mathsf{SWO}$.

Proof. Let s^* be the unique global optimum of the monotone fitness function f. Let $t \in S$ and $i \in L$. We know that $\mathrm{d}(s[i|s_i^*], s^*) < \mathrm{d}(s[i|\beta], s^*)$ for all $\beta \neq s_i^*$, hence $f(s[i|s_i^*]) > f(s[i|\beta])$ for all $\beta \neq s_i^*$, from which follows that the site-wise optimization algorithm will pick s_i^* as the best symbol. $\square$

Example 11 **($\mathsf{SWO} \cap \mathsf{UGO} \nsubseteq \mathsf{MON}$).** Let $\Sigma = \{0, 1\}$ and $L = \{0, 1, 2\}$. Take the following values for f:

$$
\begin{array}{ll}
f(000) = 8 & f(100) = 6 \\
f(001) = 4 & f(101) = 2 \\
f(010) = 7 & f(110) = 5 \\
f(011) = 3 & f(111) = 1
\end{array}
$$

One verifies that $f \in \mathsf{SWO} \cap \mathsf{UGO}$, but f is not monotone since $f(001) < f(110)$.

Proposition 12. The width $w(S)$ of the search space equals $\ell/|\Sigma|$.

Proof. Let $\alpha \in \Sigma$ and $i \in L$. There are exactly $|\Sigma|^{\ell-1}$ strings in S with a value of α at site i. Exploiting this symmetry, we restrict our calculation to the average

Hamming distance between $|\Sigma|^{\ell-1}$ α's and $(|\Sigma|-1)|\Sigma|^{\ell-1}$ other symbols, and multiply the result by $\ell|\Sigma|$. This yields

$$w(S) = \ell|\Sigma|\frac{|\Sigma|^{\ell-1}(|\Sigma|-1)|\Sigma|^{\ell-1}}{|\Sigma|^{2\ell}} = \frac{\ell}{|\Sigma|},$$

as required. ☐

Proposition 13. For any fitness function f, we have $0 \leq \mathsf{swo}(f) \leq 1$.

Proof. We have to show that no population P of $|\Sigma|^\ell$ elements can have a width strictly greater than $\ell/|\Sigma|$. Again we exploit the symmetry in the definition of the width of the population, by calculating the highest value one single site can contribute to the total width. Let x_α be the number of α's at site i, formally

$$x_\alpha = |\{s \in P\ ;\ s_i = \alpha\}| \qquad \text{for all } \alpha \in \Sigma \text{ and } i \in L. \tag{35}$$

Obviously $\sum_{\alpha\in\Sigma} x_\alpha = |\Sigma|^\ell$, and we have to optimize

$$\sum_{\alpha\in\Sigma} x_\alpha\Big(\sum_{\beta\in\Sigma\setminus\{\alpha\}} x_\beta\Big) = 2\sum_{\alpha<\beta\in\Sigma} x_\alpha x_\beta. \tag{36}$$

Using Lagrange multipliers we obtain the system of equations

$$2\sum_{\beta\in\Sigma\setminus\{\alpha\}} x_\beta + \lambda = 0 \qquad \text{for all } \alpha \in \Sigma, \tag{37}$$

$$\sum_{\alpha\in\Sigma} x_\alpha = |\Sigma|^\ell, \tag{38}$$

from which $x_\alpha = |\Sigma|^{\ell-1}$ follows for all $\alpha \in \Sigma$. This concludes our proof. ☐

Proposition 14. $\mathsf{SWO} \subset \mathsf{UNI}$

Proof. Let $f \in \mathsf{SWO}$. This means that, under f, the site-wise optimization algorithm transforms each string $s \in S$ to one of the global optima. Assume that f

contains a local optimum $m \in S$. Then necessarily m is transformed onto itself by the algorithm, a contradiction. $\square$

Example 15 (SWD $\not\subseteq$ UNI)**.** Let $\Sigma = \{0, 1\}$ and $L = \{0, 1, 2\}$. Assign arbitrary fitness values to f such that

$$f(000) > f(001) = f(010) = f(100)$$
$$f(100) > f(011) = f(101) = f(110) = f(111)$$

The function f is site-wise decidable (filling in 0 does not decrease the fitness value) and 111 is a local optimum. Note that Ex. 5 shows that (SWD $\cap$ UNI) $\setminus$ SWO is not empty.

Example 16 ((RSAO $\setminus$ SWO) $\cap$ ALC $\neq \emptyset$)**.** As in (Wilson, 1991), but slightly modified for ALC membership. Let $\Sigma = \{0, 1\}$ and $L = \{0, 1, 2\}$. Assign arbitrary fitness values to f such that

$$f(111) > f(110) > f(101) > f(011) > f(100) >$$
$$f(010) > f(000) > f(001).$$

To avoid ALC membership, take Wilson's example. We can tune the values of this function to obtain or avoid schema-GA-easy membership.

Proposition 17. SWO $\cap$ UGO $\subset$ schema-GA-easy

Proof. Let $f \in$ SWO $\cap$ UGO, and let s^* be the unique optimum of f. Consider a schema competition of order $n < \ell$, with the wildcard symbols at positions $I = \{i_1, \ldots, i_n\} \subset L$. We have to show that the schema $H(s^*)$, with wildcard symbols at positions in I and symbols of s^* at positions in $L \setminus I$ (i.e., $H(s^*)_j = s^*_j$ if

$j \notin I$), wins the competition, in favor of the schemas $H(s)$ which have the symbols of $s \in S$ at the non-wildcard positions. Let $t \in S$. Then, because of site-wise decidability and absence of local constantness, the string formed by putting the symbols of t at the wildcard positions of $H(s)$ has a strictly smaller fitness value that the string formed by putting the symbols of t at the wildcard positions of $H(s^*)$. This implies that the schema $H(s^*)$ wins the competition (average of pointwise strictly smaller sequence of numbers). $\square$

Example 18 (UNI $\nsubseteq$ schema-GA-easy)**.** Let $\Sigma = \{0, 1\}$ and $L = \{0, 1\}$. Assign the following fitness values: $f(00) = 5$, $f(01) = 1$, $f(10) = 4$ and $f(11) = 3$. The function is deceptive and steepest-ascent optimizable.