Optimization and Information Processing : NFL Results

Dirk Wiesmann

FB Informatik, LS 11, Univ. Dortmund, 44221 Dortmund, Germany
wiesmann@ls11.cs.uni-dortmund.de

Abstract. In this paper we show how no free lunch (NFL) results can be obtained by means of information theory. We derive two features to identify subsets of functions for which a NFL result holds. These subsets can be rather small compared to the set of all functions $f : \mathcal{P} \to W$, for finite sets $\mathcal{P}$ and W. Comparable results are already known, but this paper offers a didactic alternative to impart knowledge about NFL results.

1 Introduction

It has long been claimed that evolutionary algorithms have a good performance over all problems. Although the performance of problem-specific algorithms may be better on a small subset of problems, evolutionary algorithms are supposed to outperform special algorithms on much larger sets of problems [8]. Because the design of problem-specific algorithms requires extensive time and domain knowledge, it seems that evolutionary algorithms offer a very good cost-benefit ratio. In contrast to these assumptions, it is common practice to design evolutionary algorithms with problem-specific representations and operators [2, 9, 5]. It has been difficult to resolve this contradiction.

A first step was made by Wolpert and Macready [10]. They formalized the discussion on the performance of evolutionary algorithms. They used the following (no free lunch) scenario to compare the performance of optimization algorithms. The objective function is drawn randomly from the set $F = \{f : \mathcal{P} \to W\}$. The sets $\mathcal{P}$, and W are finite and W is completely ordered. The aim of the optimization algorithm is to find some $x \in \mathcal{P}$ such that $f(x) = y_{\text{opt}} \in W$ is maximal (or minimal). For every algorithm A the performance measure $\text{perf}(A, f)$ is the number of different search points $x \in \mathcal{P}$ that must be evaluated by A to find an optimal point $x_{\text{opt}} \in \mathcal{P}$ with $f(x_{\text{opt}}) = y_{\text{opt}}$. By using a dictionary, algorithms can avoid evaluating a search point twice. For randomized algorithms $\text{perf}(A, f)$ is the expected number of different search points. The average performance $\text{perf}_F(A, f)$ of an algorithm A over the set F of all functions is the average over all $\text{perf}(A, f)$, $f \in F$. Wolpert and Macready proved the following no free lunch theorem [10]:

Theorem 1. *In the NFL scenario for algorithms A and A' the equation*

$$perf_F(A, f) = perf_F(A', f)$$

holds.

The original proof is quite long and technical. By using permutations and complete induction, a much shorter proof is possible [4]. Furthermore, the NFL theorem can be generalized to specific subsets of F. A subset $F' \subseteq F = \{f : \mathcal{P} \to W\}$ is called closed under permutations, if for $f \in F'$, $f_\pi \in F'$ is also valid for every permutation π on $\mathcal{P}$ with $f_\pi(x) := f(\pi(x))$. For every subset $F' \subseteq F$ that is closed under permutations, a NFL result holds, thus $\mathrm{perf}_{F'}(A, f) = \mathrm{perf}_{F'}(A', f)$ [4].

It is often claimed that evolutionary algorithms are able to gain information about the fitness function during a run. The evolutionary algorithm can use this information to create successful offspring with a higher probability, for example, by (self-) adapting a strategy parameter like the mutation rate (step-size). In this regard it is interesting to model an optimization algorithm as an information processing process.

It is possible to obtain NFL results by means of information theory. A first attempt was made by English [6, 7]. However, the suggested approach has a disadvantage. He used joint probability distributions that were not defined on the same probability space. We present an alternative approach.

2 Entropy and Mutual Information

It is impossible to express the broad concept of information in a single definition. Therefore it is necessary to restrict the discussion to a particular aspect of information. We use the concept of entropy to measure the uncertainty of a random variable. Entropy has many properties one would expect from a measure of information. For a comprehensive introduction to information theory, we refer to the book of Cover and Thomas [1].

Let (Ω, Prob) be a discrete probability space and Ω' an arbitrary set. The function $X : \Omega \to \Omega'_X$ is called a (discrete) random variable. The distribution of X gives the probability for the occurrence of single values of X. Let $\Omega'_X := \{X(\omega) : \omega \in \Omega\}$ be the codomain of X. Then $\mathrm{Prob}_X(x) = \mathrm{Prob}(\{\omega \in \Omega : X(\omega) = x\})$ with $x \in \Omega'_X$ is the probability function of X. In the following we use the notations $\mathrm{Prob}(X = x) := \mathrm{Prob}_X(x)$, and $p(x) := \mathrm{Prob}(X = x)$.

Definition 1. *The entropy $H(X)$ of the discrete random variable X is*

$$H(X) := -\sum_{x \in \Omega'} p(x) \log p(x).$$

Therefore, the entropy is the average number of bits required to describe the random variable. It is a measure of the average uncertainty in X.

To discuss probability models that involve several random variables, one can use a joint probability mass function. For the probability space (Ω, Prob) we have n random variables with the corresponding codomains $\Omega'_1, \ldots, \Omega'_n$. An n-dimensional random vector X with the codomain $\Omega' = \Omega'_1 \times \cdots \times \Omega'_n$ is defined by $X(\omega) = (X_1(\omega), \ldots, X_n(\omega))$. The joint distribution of the random variables

$X_1, \ldots, X_n$ is given by $\mathrm{Prob}(X_1 = x_1, \ldots, X_n = x_n)$ for all $(x_1, \ldots, x_n) \in \Omega'$. In the following we use the abbreviation $p(x_1, \ldots, x_n) = \mathrm{Prob}(X_1 = x_1, \ldots, X_n = x_x)$.

Definition 2. *The joint entropy of a pair of discrete random variables $X : \Omega \to \Omega'_1$ and $Y : \Omega \to \Omega'_2$ with a joint distribution $p(x, y)$ is defined as*

$$H(X,Y) := -\sum_{x \in \Omega'_1} \sum_{y \in \Omega'_2} p(x,y) \log p(x,y).$$

Definition 3. *If both random variables X and Y are dependent, the conditional entropy*

$$\begin{aligned} H(Y|X) &:= \sum_{x \in \Omega'_1} p(x) H(Y|X = x) \\ &= -\sum_{x \in \Omega'_1} \sum_{y \in \Omega'_2} p(x,y) \log p(y|x), \end{aligned}$$

is the entropy of Y given the knowledge that X has already been observed.

Here $p(y|x) = p(x,y)/p(x)$ is the conditional probability that $Y = y$ can be observed given the knowledge that $X = x$ is valid.

The amount of information that one random variable contains about another random variable can be measured by the mutual information.

Definition 4. *Let X and Y be two random variables with a joint probability function $p(x, y)$. The mutual information $I(X; Y)$ is*

$$I(X;Y) = \sum_{x \in \Omega'_1} \sum_{y \in \Omega'_2} p(x,y) \log \frac{p(x,y)}{p(x)p(y)}.$$

The marginal distributions of X and Y can be obtained by $p(x) = \sum_{y \in \Omega'_2} p(x,y)$ and $p(y) = \sum_{x \in \Omega'_1} p(x,y)$. The relationship $I(X;Y) = H(X) - H(X|Y)$ holds between entropy and mutual information.

3 Information Theory and Optimization

In the following we make use of information theory to prove NFL results. The subset $F' \subseteq F$ denotes the current optimization problem. All objective functions $f \in F'$ are instances of the optimization problem. We assume that all functions from F' have the same probability of being selected for optimization. The optimization algorithm A does not know which function from F' is to be optimized. The random variable $X : F' \to F$ denotes the objective function selected for optimization. Thus, we have

$$\mathrm{Prob}(X = f) = \begin{cases} \frac{1}{|F'|} & \text{if } f \in F', \\ 0 & \text{if } f \notin F'. \end{cases}$$

The random variable $Z_x : F' \to W$ with $Z_x(f) = f(x)$ describes the event that $f(x)$ is computed for the search point $x \in \mathcal{P}$. The probability that the function $f \in F'$ is subject to optimization is given by

$$\text{Prob}(X = f) = \text{Prob}(Z_{x_1} = f(x_1), \ldots, Z_{x_n} = f(x_n))$$

with $x_i \in \mathcal{P}$, $x_i \neq x_j$ for $i, j \in \{1, \ldots, |\mathcal{P}|\}$ and $i \neq j$. The random variable

$$Y_{\{x_1,\ldots,x_k\}} : F' \to G_k$$

describes the observed function values after visiting the points $x_1, \ldots, x_k$ with $k \in \{1, \ldots, |\mathcal{P}|\}$, $x_i \in \mathcal{P}$, $i = 1, \ldots, k$, and

$$G_k = \{(y_1, \ldots, y_k) \mid y_i \in W, i = 1, \ldots, k\} = \underbrace{W \times \cdots \times W}_{k} = W^k.$$

With these definitions we can write $Y_{\{x_1,\ldots,x_k\}}(f) = (f(x_1), \ldots, f(x_k))$ for every function $f \in F'$. The visited points are written as sets, because the order in which the points are visited is irrelevant to the obtained information (static objective function). To ensure that this presentation is unique, for two sets $\{x_1, \ldots, x_k\}$ and $\{x_1^*, \ldots, x_k^*\}$ with $\{x_1, \ldots, x_k\}/\{x_1^*, \ldots, x_k^*\} = \emptyset$, $x_i = x_i^*$ must always hold for $i = 1, \ldots, k$. It would also be possible to represent the visited points by a vector, but that would make the following presentation more complicated.

We thus have

$$\text{Prob}(Y_{\{x_1,\ldots,x_k\}} = (y_1, \ldots, y_k)) = \text{Prob}(Z_{x_1} = y_1, \ldots, Z_{x_k} = y_k).$$

How much information can algorithm A gain by visiting the points $\{x_1, \ldots, x_k\}$, to avoid groping in the dark when creating the next (new) point $x_{k+1} \in \mathcal{P}$? The reduction in uncertainty with regard to the outcome $f(x_{k+1})$ on the basis of the search points already visited can be expressed by the mutual information

$$I(Z_{x_{k+1}}; Y_{\{x_1,\ldots,x_k\}}) = H(Z_{x_{k+1}}) - H(Z_{x_{k+1}} | Y_{\{x_1,\ldots,x_k\}}).$$

Let A^* be an optimization algorithm that has visited the search points $\{x_1^*, \ldots, x_k^*\}$. Under which conditions is algorithm A^* not able to gain more information on the search point x_{k+1} than algorithm A? In the following we use the abbreviations $p(y) = \text{Prob}(Y_{\{x_1,\ldots,x_k\}} = y)$, $p(y^*) = \text{Prob}(Y_{\{x_1^*,\ldots,x_k^*\}} = y^*)$, and $p(z, y) = \text{Prob}(Z_{x_{k+1}} = z, Y_{\{x_1,\ldots,x_k\}} = y)$. By using the transformation

$$\begin{aligned} & I(Z_{x_{k+1}}; Y_{\{x_1,\ldots,x_k\}}) = I(Z_{x_{k+1}}; Y_{\{x_1^*,\ldots,x_k^*\}}) \\ \Leftrightarrow\ & H(Z_{x_{k+1}} | Y_{\{x_1,\ldots,x_k\}}) = H(Z_{x_{k+1}} | Y_{\{x_1^*,\ldots,x_k^*\}}) \\ \Leftrightarrow\ & \sum_{y \in G_k} \sum_{z \in W} p(z, y) \log \frac{p(z, y)}{p(y)} = \sum_{y^* \in G_k} \sum_{z \in W} p(z, y^*) \log \frac{p(z, y^*)}{p(y^*)}, \end{aligned} \tag{1}$$

we obtain an answer to the question. If for all $k \in \{1, \ldots, |\mathcal{P}| - 1\}$, $x_{k+1} \in \mathcal{P}$, and all subsets $\{x_1, \ldots, x_k\} \subset \mathcal{P}$, and $\{x_1^*, \ldots, x_k^*\} \subset \mathcal{P}$ the equation (1) is valid, no algorithm A^* has an advantage over algorithm A. Thus, we obtain an NFL result. Whatever k search points an algorithm visits, it never receives more information about the unvisited search points than any other algorithm. This does not mean that an algorithm obtains no information. All algorithms obtain the same amount of information.

It would be convenient to eliminate the sums in (1). If

$$\mathrm{Prob}(Y_{\{x_1,\dots,x_k\}} = (y_1,\dots,y_k)) = \mathrm{Prob}(Y_{\{x_1^*,\dots,x_k^*\}} = (y_1,\dots,y_k)), \qquad (2)$$

holds for all $k \in \{1,\dots,|\mathcal{P}|-1\}$, $\{x_1,\dots,x_k\} \subseteq \mathcal{P}$, $\{x_1^*,\dots,x_k^*\} \subseteq \mathcal{P}$, and $(y_1,\dots,y_k) \in G_k$, then (1) is also valid. This implication is true, because we can write

$$\begin{aligned}\mathrm{Prob}(Z_{x_{k+1}} = z, Y_{\{x_1,\dots,x_k\}} = (y_1,\dots,y_k)) &= \\ &\mathrm{Prob}(Z_{x_{k+1}} = z, Z_{x_1} = y_1,\dots,Z_{x_k} = y_k), \quad (3)\end{aligned}$$

and with (2), for all $y = y^*$ the corresponding addends on both sides in (1) have the same values. To clarify the connections, we split condition (2) in two features.

Definition 5. *The problem class $F' \subseteq F = \{f : \mathcal{P} \to W\}$ has an independent value frequency, if $\forall x, x^* \in \mathcal{P}$, and $\forall y \in W$ the equation*

$$|\{f \in F' \mid f(x) = y\}| = |\{f \in F' \mid f(x^*) = y\}|$$

holds.

Definition 6. *The problem class $F' \subseteq F = \{f : \mathcal{P} \to W\}$ is called pattern-creating, if for $B \subset \mathcal{P}$ with $|B| = k > 0$, and a vector $(y_1,\dots,y_k) \in G_k$ with Prob$(Y_B = (y_1,\dots,y_k)) > 0$, also Prob$(Y_C = (y_1,\dots,y_k)) > 0$ is valid for all $C \subset \mathcal{P}$ with $|C| = k$.*

Theorem 2. *If a problem class $F' \subseteq F = \{f : \mathcal{P} \to W\}$ has an independent value frequency and is pattern-creating, this is equivalent to the fulfillment of condition (2).*

Proof. We first assume that condition (2) is valid for F'. Then F' has an independent value frequency and is pattern-creating. Suppose F' has no independent value frequency. Then there exist $x, x^* \in \mathcal{P}$, and $y \in W$ with (w.l.o.g.)

$$\begin{aligned}&|\{f \in F' \mid f(x) = y\}| < |\{f \in F' \mid f(x^*) = y\}| \\ \Leftrightarrow\ &\frac{|\{f \in F' \mid f(x) = y\}|}{|F'|} < \frac{|\{f \in F' \mid f(x^*) = y\}|}{|F'|} \\ \Rightarrow\ &\mathrm{Prob}(Y_{\{x\}} = y) \neq \mathrm{Prob}(Y_{\{x^*\}} = y).\end{aligned}$$

This contradicts (2). Thus, F' possesses an independent value frequency. Here we used the assumption that every function $f \in F'$ has the same probability of being subject to optimization. Now let us suppose that F' is not pattern-creating. For some $B \subset \mathcal{P}$ with $|B| = k > 0$ and a vector $(y_1, \ldots, y_k) \in G_k$ with $\mathrm{Prob}(Y_B = (y_1, \ldots, y_k)) > 0$, a subset $C \subset \mathcal{P}$ with $|C| = k$, and $\mathrm{Prob}(Y_C = (y_1, \ldots, y_k)) = 0$ exists. Then $\mathrm{Prob}(Y_B = (y_1, \ldots, y_k)) \neq \mathrm{Prob}(Y_C = (y_1, \ldots, y_k))$ is true. This is a contradiction to the assumption. Thus, F' is also pattern-creating.

We now show that the condition (2) is true if F' is pattern-creating and has an independent value frequency. The set

$$M_{\{x_1,\ldots,x_k\}} = \{(y_1, \ldots, y_k) \in G_k \mid (f(x_1), \ldots, f(x_k)) = (y_1, \ldots, y_k), f \in F'\}$$

contains all "patterns" $(y_1, \ldots, y_k)$ that are created by functions from F' after seeing the search points $\{x_1, \ldots, x_k\}$. Because F' is pattern-creating, for all $(y_1, \ldots, y_k) \in M_{\{x_1,\ldots,x_k\}}$, $(y_1, \ldots, y_k) \in M_{\{x_1^*,\ldots,x_k^*\}}$ is also true, hence

$$|M_{\{x_1,\ldots,x_k\}}| = |M_{\{x_1^*,\ldots,x_k^*\}}| \tag{4}$$

holds. The function $c_{\{x_1,\ldots,x_k\}} : G_k \to \mathbb{N}$ with

$$c_{\{x_1,\ldots,x_k\}}(y_1, \ldots, y_k) := |\{f \in F' \mid (f(x_1), \ldots, f(x_k)) = (y_1, \ldots, y_k)\}|$$

counts the number of pattern $(y_1, \ldots, y_k)$ that are created by functions from F' after evaluating the search points $\{x_1, \ldots, x_k\}$. From it

$$\sum_{(y_1,\ldots,y_k)\in G_k} c_{\{x_1,\ldots,x_k\}}(y_1, \ldots, y_k) = |F'| \tag{5}$$

is valid. Obviously we have

$$\mathrm{Prob}(Y_{\{x_1,\ldots,x_k\}} = (y_1, \ldots, y_k)) = \frac{c_{\{x_1,\ldots,x_k\}}(y_1, \ldots, y_k)}{|F'|}.$$

If the equation

$$c_{\{x_1,\ldots,x_k\}}(y_1, \ldots, y_k) = c_{\{x_1^*,\ldots,x_k^*\}}(y_1, \ldots, y_k)$$

holds for all $k \in \{1, \ldots, |\mathcal{P}|\}$ (2) is also fulfilled.

Next we show, that F' has no independent value frequency, if

$$c_{\{x_1,\ldots,x_k\}}(y_1, \ldots, y_k) \neq c_{\{x_1^*,\ldots,x_k^*\}}(y_1, \ldots, y_k)$$

is valid. Without loss of generality we suppose that

$$c_{\{x_1,\ldots,x_k\}}(y_1, \ldots, y_k) < c_{\{x_1^*,\ldots,x_k^*\}}(y_1, \ldots, y_k).$$

From equations (4) and (5), a vector $(y_1', \ldots, y_k') \in M_{\{x_1,\ldots,x_k\}}$ exists with

$$c_{\{x_1,\ldots,x_k\}}(y_1', \ldots, y_k') > c_{\{x_1^*,\ldots,x_k^*\}}(y_1', \ldots, y_k').$$

The vectors $(y_1, \ldots, y_k)$ and $(y_1', \ldots, y_k')$ differ in at least one position $i \in \{1, \ldots, k\}$, so that $y_i \neq y_i'$. To obtain an independent value frequency for x_i, x_i^*, and y_i, a vector $(y_1^*, \ldots, y_k^*) \in M_{\{x_1,\ldots,x_k\}}$ with

$$c_{\{x_1,\ldots,x_k\}}(y_1^*, \ldots, y_k^*) > c_{\{x_1^*,\ldots,x_k^*\}}(y_1^*, \ldots, y_k^*),$$

and $y_i^* = y_i$ must be chosen. If no such vector $(y_1^*, \ldots, y_k^*)$ exists, the ith element cannot be compensated. For $y_i \in W$ the condition of independent value frequency would be violated. Otherwise a position $j \neq i$ exists such that $y_j^* \neq y_j'$ holds. The jth element must be compensated in turn. For each of at most $|F'|$ rounds, we can argue in the same way.

4 Examples

We now use the following example to discuss the previous results. Let $F := \{f : \{1,2,3\} \to \{0,1\}\}$, and $F', F'' \subset F$ with $F' = \{f_1, f_2, f_3\}$, and $F'' = \{f_1, f_2, f_3, f_4\}$. To define the functions we use the matrix representation shown in Figure 1. Every function f_i is represented by a list of its function values. The single lists are written one below the other. Position (i,j) of the matrix holds the function value $f_i(j)$ with $i \in \{1,2,3,4\}$, and $j \in \{1,2,3\}$.

	1	2	3
f_1	1	0	0
f_2	0	1	0
f_3	0	0	1
f_4	0	0	0

Fig. 1. Matrix representation of the functions f_1, f_2, f_3 and f_4. At position (i,j) of the matrix the function value $f_i(j)$ with $i \in \{1,2,3,4\}$, and $j \in \{1,2,3\}$ can be found.

The problem class F' is closed under permutation. Thus, a NFL result holds for F'. The class F' has an independent value frequency. Every column of the matrix (Figure 1) contains a one and two zeroes. Furthermore F' is pattern-creating. To see this, we look at every pattern (0), (1), $(0,0)$, $(0,1)$, $(1,0)$, $(1,1)$, and all subsets of search points $\{1\}$, $\{2\}$, $\{3\}$, $\{1,2\}$, $\{1,3\}$, $\{2,3\}$. From the matrix we obtain the following probabilities for F':

$$\begin{array}{llll}
\mathrm{Prob}(Y_{\{1\}} = (0)) & = \mathrm{Prob}(Y_{\{2\}} = (0)) & = \mathrm{Prob}(Y_{\{3\}} = (0)) & = \frac{2}{3} \\
\mathrm{Prob}(Y_{\{1\}} = (1)) & = \mathrm{Prob}(Y_{\{2\}} = (1)) & = \mathrm{Prob}(Y_{\{3\}} = (1)) & = \frac{1}{3} \\
\mathrm{Prob}(Y_{\{1,2\}} = (0,0)) & = \mathrm{Prob}(Y_{\{1,3\}} = (0,0)) & = \mathrm{Prob}(Y_{\{2,3\}} = (0,0)) & = \frac{1}{3} \\
\mathrm{Prob}(Y_{\{1,2\}} = (0,1)) & = \mathrm{Prob}(Y_{\{1,3\}} = (0,1)) & = \mathrm{Prob}(Y_{\{2,3\}} = (0,1)) & = \frac{1}{3} \\
\mathrm{Prob}(Y_{\{1,2\}} = (1,0)) & = \mathrm{Prob}(Y_{\{1,3\}} = (1,0)) & = \mathrm{Prob}(Y_{\{2,3\}} = (1,0)) & = \frac{1}{3} \\
\mathrm{Prob}(Y_{\{1,2\}} = (1,1)) & = \mathrm{Prob}(Y_{\{1,3\}} = (1,1)) & = \mathrm{Prob}(Y_{\{2,3\}} = (1,1)) & = 0.
\end{array}$$

Hence, F' is pattern-creating (see definition 6). Because all relevant probabilities are already given, we can directly see that condition (2) also holds. As an example, Table 1 shows the marginal distribution $p(z,y) = \mathrm{Prob}(Z_{x_3} = z, Y_{\{x_1,x_2\}} = y)$ with $z \in \{0,1\}$, $y \in \{0,1\}^2$, and $x_1, x_2, x_3 \in \{1,2,3\}$.

		y				
		(0,0)	(0,1)	(1,0)	(1,1)	
z	0	0	$\frac{1}{3}$	$\frac{1}{3}$	0	$\frac{2}{3}$
	1	$\frac{1}{3}$	0	0	0	$\frac{1}{3}$
		$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{3}$	0	1

Table 1. Marginal distribution $p(z, y) = \mathrm{Prob}(Z_{x_3} = z, Y_{\{x_1,x_2\}} = y)$ with $z \in \{0, 1\}$, $y \in \{0, 1\}^2$, and $x_1, x_2, x_3 \in \{1, 2, 3\}$ for the class F'.

For the class F'' a NFL result also holds. F'' is pattern-creating and has an independent value frequency. The constant function f_4 adds the same value to every column of the matrix in Figure 1. Thus, the independent value frequency of F' is maintained. A constant function that creates the pattern $(y_1, \ldots, y_k)$ with $y_1 = \cdots = y_k$ for the search points $\{x_1, \ldots, x_k\}$, creates the same pattern for every other set $\{x_1^*, \ldots, x_k^*\}$.

Corollary 1. *If a problem class that has an independent value frequency and is pattern-creating is extended by a constant function, the new problem class also possesses these features.*

5 Conclusion

We presented an alternative approach to obtain NFL results by means of information theory. Subsets $F' \subseteq F$ for which a NFL result holds can be characterized by the features of independent value frequency and pattern creation. One must be aware, however, that the practical implications of NFL results are limited. The NFL scenario does not model real life optimization [3]. In practice, the computation of $f(x)$ has to be fast (efficient) and the corresponding program for fitness evaluation is rather short (has a small Kolmogoroff complexity). Thus, a realistic optimization scenario leads to classes of functions with in some sense restricted complexities. For more realistic optimization scenarios it is possible that optimization techniques differ in their efficiency [3].

Acknowledgements

This research was supported by the Deutsche Forschungsgemeinschaft as part of the collaborative research center "Computational Intelligence" (531).

References

1. T. M. Cover and J. A. Thomas. *Elements of Information Theory.* Wiley, New York, NY, 1991.

2. L. Davis. *Handbook of Genetic Algorithms.* Van Nostrand Reinhold, New York, NY, 1991.
3. S. Droste, T. Jansen, and I. Wegener. Perhaps not a free lunch but at least a free appetizer. In W. Banzhaf, J. Daida, A. Eiben, M. Garzon, V. Honavar, M. Jakiela, and R. Smith, editors, *Proc. of the Genetic and Evolutionary Computation Conference (GECCO '99)*, pages 833–839, San Francisco, CA, 1999. Morgan Kaufmann.
4. S. Droste, T. Jansen, and I. Wegener. Optimization with randomized search heuristics – The (A)NFL theorem, realistic scenarios, and difficult functions. *Theoretical Computer Science*, 2002. (in press).
5. S. Droste and D. Wiesmann. On the design of problem-specific evolutionary algorithms. In A. Ghosh and S. Tsutsui, editors, *Advances in Evolutionary Computing*, Berlin, 2002. Springer. (in press).
6. T. M. English. Evaluation of evolutionary and genetic optimizers: No free lunch. In L. Fogel, P. J. Angeline, and T. Bäck, editors, *Evolutionary Programming V : Proc. Fifth Ann. Conf. on Evolutionary Programming*, pages 163–169, Cambridge, Mass., 1996. MIT Press.
7. T. M. English. Some information theoretic results on evolutionary optimization. In P. J. Angeline, editor, *Proc. of the 1999 Congress on Evolutionary Computation (CEC99), Washington D.C., July 6–9, 1999*, pages 788–795, Piscataway, NJ, 1999. IEEE Press.
8. D. E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning.* Addison-Wesley, Reading, MA, 1989.
9. Z. Michalewicz. *Genetic Algorithms + Data Structures = Evolution Programs.* Springer, Berlin, 1996.
10. D. H. Wolpert and W. G. Macready. No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, 1(1):67–82, 1997.