Intelligent Design and the NFL Theorems

Olle Häggström*

June 2, 2006

Abstract

Another look is taken at the model assumptions involved in William Dembski's (2002a) use of the NFL theorems from optimization theory to disprove the Darwinian theory of evolution by natural selection, and his argument is shown to be irrelevant to evolutionary biology.

1 Introduction

Recent years have witnessed, mainly in the United States, a change of focus in the anti-Darwinian discourse. Biblical literalists and young-earth creationists have to a large extent given way to proponents of *Intelligent Design* (ID), who accept much of modern biology and natural history, insisting only that complex creatures such has ourselves cannot come about "bottom-up" in a universe governed just by natural laws, but bear unmistakable signs of being the work of an intelligent agent. The identity of this agent – be it God, extraterrestrial aliens, or something else – is typically left out of the discussion. For critical surveys of the ID movement, see for instance Crews (2001) and Orr (2005); see also the collections edited by Pennock (2001) and Brockman (2006).

Although there are many advocates of ID with a high profile in public debate, there are in fact very few that combine this with scientific aspirations. Besides Michael Behe, famous for his best-seller *Darwin's Black Box* (1996), the most well-known such advocate is William Dembski. The purpose of the present paper is to make a critical evaluation of some central parts of Dembski's arguments against the Darwinian paradigm in biology.

Going back to Paley (1802) and others, the so-called "argument from design" is classical: it is unfathomable that advanced life forms, with all

*Professor of mathematical statistics, Chalmers University of Technology, Sweden, `http://www.math.chalmers.se/~olleh/`

their complexity and apparent purposefulness, could come about if not as the work of an intelligent designer. Phrased in this way, the argument suffers, obviously, from a lack of precision. In his book *The Design Inference* (1998), Dembski sets out to improve it by making precise the meaning of "complexity" through his notion of *specified complexity* (although see Wein (2002a) for an account of how inconsistently Dembski uses his own concept). His follow-up book *No Free Lunch* (Dembski, 2002a) is an ambitious project. In his own words: "*The Design Inference* laid the groundwork. This book demonstrates the inadequacy of the Darwinian mechanism to generate specified complexity" (Dembski 2002a, p. xiii). The title of the book refers to the key role in Dembski's argument played by the so-called NFL (No Free Lunch) theorems from optimization theory.

After giving the necessary background on the mathematics of optimization theory and the NFL theorems in Sections 2 and 3, I will outline Dembski's use of the latter in Section 4. Then, in Sections 5 and 6, I will demonstrate the main error in his argument and the irrelevance of NFL to evolutionary biology.

Dembski's *No Free Lunch* (2002a) has been sharply criticized elsewhere, as in Orr (2002), Shallit (2002) and especially Wein (2002a)[1]. However, much of this criticism is less devastating than it might have been with a proper understanding of what the NFL theorems actually say (in my concluding Section 7, I will briefly comment on some of these shortcomings). The role of the present paper is to try to remedy the situation by offering a mathematician's account of what the NFL theorems really mean, and why they cannot be applied to evolutionary biology. Along the way, we will see that they are in fact much simpler than earlier marketing has suggested, and readers looking for a quick and easy way to grasp what NFL is all about are recommendad to glance ahead at statement (6) at the end of Section 5.

2 A few mathematical preliminaries

In order to discuss mathematical optimization theory, we first need to recall the basic mathematical notions of sets and functions.

A *set* is a collection of objects, called the *elements* of the set. A set may be finite or infinite. Examples of finite sets are, e.g., the set S_1 of all positive integers up to 3, and the set S_2 of all Nordic countries: $S_1 = \{1, 2, 3\}$ and $S_2 = \{$Denmark, Finland, Iceland, Norway, Sweden$\}$. As an example of an infinite set, me may for instance take S_3 to be the set of *all* positive integers:

[1] See also the subsequent exchange in Dembski (2002b, 2002c) and Wein (2002b).

$S_3 = \{1, 2, 3, \ldots\}$. It is important to note that the definition of a set is independent of the order in which the elements are written down, so that for instance $\{1, 2, 3\} = \{2, 1, 3\}$. As shorthand mathematical notation for the statement "the set S includes the element x", we write $x \in S$. For instance, $1 \in S_1$, Sweden $\in S_2$ and $792 \in S_3$ are true statements, while $792 \in S_1$ is not. Furthermore, write $|S|$ for the number of elements of S, so that for instance $|S_1| = 3$, $|S_2| = 5$, and $|S_3| = \infty$; $|S|$ is called the *cardinality* of S.

A *function* is a rule which to each element of a given set assigns a single element from another given set. We write $f : V \to S$ to emphasize that f is a function that to each element of the set V assigns an element of the set S. In this case we say that f is a function *from V to S*. For instance, $f : S_2 \to S_3$, with S_2 and S_3 as above, could be the function which to each Nordic country assigns its population number as of January 1, 2006. Nothing prevents two elements of the first set to be assigned the same value from the second set, as would be the case here if Denmark and Finland happened to have exactly the same number of inhabitants.

If V and S are sets, then the collection of all possible functions from V to S is itself a set, and is denoted by S^V. This notation is partly explained by the fact that if V and S are finite sets with cardinalities $|V|$ and $|S|$, then S^V has cardinality $|S|^{|V|}$. As an example, let us take $V = \{1, 2, \ldots, 100\}$ and $S = \{0, 1\}$. Then each $f \in S^V$ is a function that gives a binary value (0 or 1) to each integer between 1 and 100. The function is specified by its values $f(1), f(2), \ldots, f(100)$, and can thus be thought of as a binary sequence consisting of 100 bits (0's or 1's), where $f(1)$ specifies the first bit, $f(2)$ specifies the second bit, and so on. This means that the set

$$S^V = \{0, 1\}^{\{1,2,\ldots,100\}} \tag{1}$$

can be thought of as the set of all possible binary sequences of length 100, and by the cardinality formula $|S|^{|V|}$ there exist 2^{100} different such sequences.

For any set S and any positive integer n, it is customary to write S^n as shorthand for $S^{\{1,2,\ldots,n\}}$. For instance, the set of length-100 binary strings in (1) can be written as $\{0, 1\}^{100}$. As another example $\{A, C, G, T\}^{1000}$ denotes the set of all length-1000 DNA sequences, and there exist precisely 4^{1000} different such sequences.

3 Optimization and the NFL theorems

In combinatorial optimization, one is given a finite set V and a function $f : V \to \mathbf{R}$ which to each $x \in V$ assigns a real number (we write, following convention, $\mathbf{R}$ for the set of all real numbers). The task is to find an element $x \in V$ that maximizes $f(x)$. At first sight, this may seem like a trivial task: since V is finite, all we need to do is simply to go through all $x \in V$ systematically, calculate $f(x)$ for each of them, while keeping track of the maximum seen so far.

The reason why this "brute force" approach does not suffice is that V is usually so large that time constraints make the approach infeasible. Typically, the number of elements of V grows exponentially (or faster) in some parameter n that describes the size of the problem in some natural way. For instance, V could be the set of binary strings of length n, a set having cardinality 2^n. Or V could be the set of all permutations of n distinct objects (i.e., all the ways to line up the n objects in a queue); in this case V has cardinality $n!$. In both cases, the brute force method of calculating $f(x)$ for all $x \in V$ is out of the question even for moderately sized problems such as $n = 100$.

Other, less time-consuming, algorithms are therefore needed. A common approach involves so-called *local search* in V. This necessitates the introduction of some "geographic" structure in V, which can be accomplished by declaring the existence of *links* between some (but not all) pairs of elements $x, y \in V$. The set of all y that are linked to a given $x \in V$ is called the *neighborhood* of x. There is much freedom in setting up the links, but it needs to be done in such a way that, on one hand, each x has a neighborhood of manageable size, and, on the other hand, the network of links becomes "well connected" (in some sense). In specific examples, natural link structures often more or less suggest themselves: when V is the set of length-n binary strings, we may declare links precisely between those $x, y \in V$ that differ only in one bit, or when V is the set of permutations of n objects we may decide to declare a link between two permutations when one of them can arise from the other by interchange of exactly two of the objects.

Given the link structure, the basic local search algorithm proceeds as follows. Start at some arbitrary $x \in V$, compute f at x and at all of its neighbors, and move to the neighbor y whose f-value is the largest (unless they are all smaller than $f(x)$ in which case we stay at x). Then repeat the process, moving to the vertex z that has the largest f-value among y and y's neighbors. This goes on until we get stuck.

This algorithm is sometimes called the *hill-climber*, as it can be pictured as a hiker in a hilly landscape, always going in the direction of the steepest climb, until the top of a hill is reached. Such hill-climbing sometimes works well, but a huge drawback is that the algorithm may get stuck on a relatively modest hill without noticing the huge mountain peak further away.

To deal with this drawback, a variety of modifications of the hill-climber algorithm have been proposed and are widely used; see, e.g., Aarts and Lenstra (1997). These modifications may for instance include randomizing the walk to allow occasional downhill steps (as in the famous simulated annealing algorithm) or permitting occasional "long jumps" in the landscape. Many of these modifications are quite sophisticated.

These algorithms are not only used for the pure optimization problem that we have focused on so far, but also – in fact more often – for the purpose of locating some large (but not necessarily the largest) value of f. Specifically, the goal may be to find an $x \in V$ such that $f(x)$ exceeds some given level t. The algorithm then proceeds until it encounters an element of the set T consisting of all $x \in V$ satisfying $f(x) \geq t$. The problem of finding some $x \in T$ should, strictly speaking, be called a *search problem* rather than an optimization problem. We call T the *target set*, and it can be written in compact mathematical notation as

$$T = \{x \in V : f(x) \geq t\} . \tag{2}$$

More generally, we may not always be in a situation where "the larger the value of f, the better", so it makes sense to allow for a target set T that is not necessarily of the form (2), but may be an arbitrary subset of V. In interesting search problems, T is typically very rare, in the sense that only a very small fraction of all elements $x \in V$ are also in T.

This sets the stage for the NFL theorems of Wolpert and Macready (1997), who showed that for these optimization and search problems, no algorithm is better than any other, in a certain average sense. This may sound very surprising, so let me describe in more detail what the basic NFL theorem actually says.[2]

Wolpert and Macready restrict to the setting where the function f is only allowed to take values in some prescribed finite subset S of $\mathbf{R}$. In terms of set notation, $f : V \to S$, where S is a finite set of real numbers. This restriction is natural because in a computer implementation everything is necessarily discrete.

Once the set V over which we optimize, and the set S of allowed values for the function f, are given, we know from Section 2 that there exist exactly $|S|^{|V|}$ different functions $f : V \to S$. Usually the number $|S|^{|V|}$ of such functions is a stupendously large numbers, since already $|V|$ is typically very large. The basic NFL theorem concerns an average over all these functions.

[2] Most of the discussion will focus on this particular NFL theorem, but see Section 7 for some indication of why the plural form "theorems" is used above.

The algorithms considered by Wolpert and Macready are of the following form. First, an element $x_{(1)} \in V$ is chosen according to some rule (which, like those that follow, may or may not involve the use of random numbers), and $f(x_{(1)})$ is computed. Then $x_{(2)} \in V$ is chosen according to some rule that may take into account $x_{(1)}$ and $f(x_{(1)})$, after which $f(x_{(2)})$ is computed. And so on: after k steps of the algorithm, it has recorded $x_{(1)}, \ldots, x_{(k)}$ and $f(x_{(1)}), \ldots, f(x_{(k)})$, and goes on to choose an $x_{(k+1)}$ using a rule that may take into account all these previous values. The only other condition that the basic NFL theorem requires is that no $x \in V$ is chosen more than once.

Imagine now that the first k f-values $f(x_{(1)}), \ldots, f(x_{(k)})$ have been recorded, and define some event E_k solely in terms of these. The prototype example is to take E_k to be the event that at least one of the recorded values $f(x_{(1)}), \ldots, f(x_{(k)})$ puts its corresponding $x_{(i)}$ in the target set T. The basic NFL theorem now states that

> averaged over all the $|S|^{|V|}$ different possible functions f,
> the probability of the event E_k is the same for any choice
> of algorithm.

Among other things, this tells us that no algorithm is better than any other at quickly finding an element in the target set T. In particular, no algorithm is better than the "blind search" algorithm that does the following: first pick $x_{(1)}$ uniformly at random from V (i.e., every element of V has the same probability $1/|V|$ of being chosen), then $x_{(2)}$ is chosen uniformly at random among the others (regardless of $f(x_{(1)})$), and so on. If, as usual, V is a very large set and the target set T is very rare, then the time taken to find some $x \in T$ will most likely be enormous.

Thus, the basic NFL theorem seems to provide us with a disheartening message: no matter how clever we are, we cannot expect to devise algorithms that are better than the hopelessly primitive and inefficient blind search algorithm.

In practice, however, there is no reason to despair. The key property of the basic NFL theorem that allow us in practice to circumvent its dark message is the averaging over all possible functions f that is involved. In almost all concrete optimization problems we have some prior information or at least some rough idea of how f varies across V, and such information can be exploited in the construction of clever and efficient optimization algorithms, unfettered by any NFL theorem. The reason why the pessimistic message of the basic NFL theorem no longer applies in such a situation is that it averages over *all* possible f, and not just over the kinds of f that we know to be more likely.

The moral of Wolpert and Macready (1997) is that we cannot expect to construct efficient optimization or search algorithms *unless we exploit some specific property of f*.[3] Further light on their result will be shed in Section 5, but before that, I will explain how NFL is claimed to disprove Darwinian evolution.

4 Dembski's application to evolution

During the last couple of decades, evolutionary biology has had a large influence on optimization theory: much of the development of optimization algorithms as described in the previous section has been based on mimicking the biological principles of reproduction, mutation, and evolution. Information has also travelled in the other direction, and viewing biological evolution from an algorithmic perspective has sometimes turned out useful; see, e.g., the popular account by Dennett (1995) for a very consistent employment of this perspective.

The algorithmic view on Darwinian evolution is also taken up by Dembski (2002a). In this section, I will describe his NFL-based argument in the case of a single species evolving in a fixed environment. I will thus ignore for the moment the complications of time-dependent environments or of several species coevolving. Dembski's argument, as well as my refutation of it, extend in a straightforward manner to these situations; see Section 7 for some brief remarks in this direction.

As a preparatory lemma to his main argument, Dembski notes that the kind of blind search that was described in the previous section cannot possibly account for the occurrence of what he calls specified complexity, such as ourselves or other large animals and plants. This is absolutely correct. The human genome is about 3,000,000,000 base pairs long. Let us now take V to consist of all DNA sequences up to that length, and the target set T to be the set of all such DNA sequences giving rise to a creature exhibiting specified complexity. The number of elements of V then becomes something of the order $10^{1,800,000,000}$ – a truly Vast number. (Following Dennett (1995), I write Vast for "Very much larger than ASTronomical.") The target set T is also Vast, but a more important observation is that T is so much smaller than V that if we pick an element at random (uniform distribution) from V, then the odds against getting an element of T are also Vast. The precise Vast-ness of this quantity is very difficult to estimate (partly because of the difficulty in pinpointing exactly what specified complexity is), but it

[3] It is this observation that prompted them to use the phrase No Free Lunch.

seems reasonably safe to state that $|V|/|T|$ is somewhere between 10^{1000} and $10^{1,000,000,000}$. Assuming this, the probability that a random choice from $|V|$ hits the target set $|T|$ is between 10^{-1000} and $10^{-1,000,000,000}$, and the number of attempts needed by the blind search algorithm before hitting T will most likely be somewhere between 10^{1000} and $10^{1,000,000,000}$. The age of the earth (or of the universe, for that matter) is nowhere near long enough to encompass such a search procedure – even if we take into account the massive parallelism that evolution may exploit through searching along a large number of lines of descent simultaneously. Thus, the infeasibility of the blind search algorithm is settled.

Equipped with this lemma, the basic NFL theorem does the rest, according to Dembski.[4] Of course, no one claims that Darwinian evolution proceeds via the above blind search algorithm. The basic NFL theorem, however, tells us that no other algorithm can expect to do better, and hence Darwinian evolution cannot produce specified complexity. That is, unless either the algorithm is set up using prior information of the function f (and here it is inconsequential whether this function represents some fitness quantity, or some more general phenotypic aspect) to help it reach the target set T, or conversely f is set up to fit the algorithm (Dembski 2002a, Sections 4.4 and 4.6). This means, still according to Dembski, that the specified complexity appearing as the result of biological evolution must have been present already in the algorithm or the fitness landscape – an observation that he calls *the displacement problem* (Dembski 2002a, Section 4.7). This demonstrates that natural laws are "intrinsically incapable of delivering [specified intelligence]. Indeed, all our evidence points to intelligence as [its] only source" (Dembski 2002a, p. 207).

Of course, this argument is elaborated in much more detail in *No Free Lunch*, and perhaps Dembski upon reading this will feel that the last three sentences of the previous paragraph do not give complete justice to his line of reasoning. The rough description I have given of Dembski's argument in this section is nevertheless sufficient to make it clear that the next two sections refute it irreparably.

[4] Dembski picked up the idea that the NFL theorems might pose a challenge to evolutionary biology from Stuart Kauffman, who writes: "The no-free-lunch theorem proves that, averaged over all landscapes, no search algorithm outperforms any other. [...] And here we organisms are, stuck using mutation, recombination and selection. [...] Where did the 'good' landscapes come from, those that Darwinian gradualism works so well in searching?" (Kauffman 2000, p. 197). A similar quote of Kauffman appears in Dembski (2002a, p. 224).

5 A probabilistic interpretation of NFL

In this section I will give a probabilistic interpretation of the underlying assumption of the NFL theorems that make them look a lot less mysterious. Earlier treatments of the NFL theorems have, as far as I know, not employed this perspective.[5] In particular, Dembski (2002a) shows no sign of probabilistic understanding of the NFL theorems, although later in Dembski (2005) he begins to speak of similar matters in probabilistic terms.

The basic NFL theorem involves an average over all possible functions f. Whenever an average or a weighted average appears in a mathematical argument, one may stop and consider whether the averaging has some probabilistic interpretation (as it usually does), and if so, how the implicit probabilistic model might be interpreted. This can often be quite illuminating.

In the setting of Section 3, the averaging amounts to picking one of the $|S|^{|V|}$ different possible functions $f : V \to S$ at random according to uniform distribution, meaning that each one is picked with probability $1/|S|^{|V|}$. For the search problem of finding some $x \in V$ belonging to the target set T, an equivalent probabilistic way of formulating the basic NFL theorem is thus as follows: the distribution of the time taken for a search algorithm A to find an element of T is the same regardless of the choice of A – provided that the function f is generated by a random mechanism that picks one of the $|S|^{|V|}$ possible realizations with equal probability.

It is worthwile to reflect over what it means that f is chosen according to uniform distribution on S^V. I claim that

> choosing a random function $f : V \to S$ according to uniform distribution on S^V is equivalent to choosing, for each $x \in V$ *independently*, $f(x)$ according to uniform distribution on S. (3)

Here and thoughout, independence is taken to mean *statistical independence.* This, in turn, means in this particular context that for any collection $x_1, \ldots, x_k$ of different elements of V, and any given values $s_1, \ldots s_k$ from S, we have that[6]

$$\mathbf{P}(f(x_1) = s_1, \ldots, f(x_k) = s_k) \quad = \quad \mathbf{P}(f(x_1) = s_1) \times \cdots \times \mathbf{P}(f(x_k) = s_k)\,.$$

[5] Figure 2 of Wein (2002a) suggests that Wein is aware of the key observation (3) below, but it is not spelled out in his text.

[6] $\mathbf{P}$ is short for "the probability of".

A nice intuitive interpretation is that no knowledge of the f-values for any collection of elements from V gives reason to deviate from the belief that the f-values of the *other* elements from V are uniformly distributed on S.

Statement (3) is a well-known fact in probability theory, and really nothing more than a straightforward extension of the standard first-year textbook example concerning the roll of two dice: the statement that all 36 outcomes $(1,1),(1,2),\ldots,(1,6),(2,1),\ldots,(6,6)$ have the same probability is equivalent to the the statement that the two dice are independent and that the distribution for each of them is uniform on $\{1,2,\ldots,6\}$.

For completeness and for the reader's convenience, let me nevertheless give the explicit argument for (3). Suppose that V has m elements $x_1,\ldots,x_m$, and that S has l elements $s_1,\ldots,s_l$. Suppose furthermore that for each $x \in V$ independently, we choose $f(x)$ according to uniform distribution on S. To prove the claim (3), we need to show that for any $s_1,\ldots,s_m \in S$, the formula

$$\mathbf{P}(f(x_1)=s_1,\ldots,f(x_m)=s_m)=1/l^m \tag{4}$$

holds. Now, the independence assumption tells us that the left-hand-side of (4) can be factorized into

$$\mathbf{P}(f(x_1)=s_1)\times\cdots\times\mathbf{P}(f(x_m)=s_m)\,. \tag{5}$$

Since each of the factors in (5) equals $1/l$, the identity (4) is verified, and the claim (3) established.

Now that we are equipped with the characterization (3), the basic NFL theorem becomes very easy to understand (and to prove). To this end, imagine an algorithm A, as in Section 3, that after k steps has visited $x_{(1)},\ldots,x_{(k)} \in V$, and observed $f(x_{(1)}),\ldots,f(x_{(k)})$.[7] Now, whichever $x_{(k+1)}$ the algorithm chooses to visit next, the f-value that it will find there is, due to the independence property in (3), uniformly distributed on S (regardless of which elements $x_{(1)},\ldots,x_{(k)}$ of V the algorithm has visited, and which values of $f(x_{(1)}),\ldots,f(x_{(k)})$ it has observed). Hence, the rule for how to select $x_{(k+1)}$ does not influence what we see there. Since k was arbitrary it follows that $f(x_{(1)}),f(x_{(2)}),\ldots$ form a sequence of independent random values whose common distribution is uniform on S. Since this conclusion is reached regardless of the details of A, it follows that the choice of A has no influence on the distribution of the sequence $f(x_{(1)}),f(x_{(2)}),\ldots$. And this is precisely what the basic NLF theorem says.

[7] The notation is worth stressing: $x_{(i)}$ denotes the i:th element visited by the algorithm, whereas x_i denotes the i:th element in some fixed but arbitrary enumeration of V.

In fact, not only does the observation (3) provide us with an almost trivial proof of the basic NFL theorem – it also suggests some immediate generalizations. Indeed, the argument we just indicated uses that the $f(x)$'s are independent with the same distribution, but not that their common distribution is uniform on S. Hence, the assertion of the basic NFL theorem holds under this weaker independence assumption. And by the same token, the assumption can be weakened even further to that of so-called *exchangeability*, which means that the joint distribution of $f(x_1), \ldots, f(x_m)$ equals the joint distribution of any permutation of them (see, e.g., Kallenberg, 2005).

With this latter generality in mind, the basic NFL theorem is not much more than a fancy (and more general) way of phrasing the following fact:

> If we spread a well-shuffled deck of cards face-down on a table and wish to find the ace of spades by turning over as few cards as possible, then no sequential procedure for doing so is better than any other. (6)

This obvious card-deck example summarizes pretty much all there is to the basic NFL theorem (or any of its variants). In spite of this, Dembski is not the only one who has tried to create a hype around the result. For instance, Wolpert and Macready themselves (1997) contribute their share to the hype. And with astonishing lack of perspective, Ho and Pepyne (2002) compare the basic NFL theorem to one of the deepest achievements in 20th century mathematics: Gödel's incompleteness theorem.

6 Dembski's error

Let us now examine Dembski's use of NFL in the light of the probabilistic interpretation given in Section 5. For concreteness take, as in Section 4, V to be the set of all DNA sequences of length up to 3,000,000,000. Also, take $f : V \to S$ to be some measure of fitness, so that for each $x \in V$, $f(x)$ describes the fitness of an organism with DNA sequence x. Of course, most such DNA sequences do not correspond to an organism at all, so for such x we take $f(x)$ to be the minimum of the set S of possible values – say, $f(x) = 0$.

Furthermore, let us equip V with a link structure as in Section 3. Specifically, let us declare a link between two DNA sequences $x, y \in V$ precisely when one of them can be obtained from the other either by changing a single nucleotide pair, by inserting one, or by deleting one. This choice of link structure is made in order that a move from an element $x \in V$ to a neighbor

$y \in V$ corresponds to a mutation of the simplest possible (single-nucleotide) kind.[8] Thus, the reproduction-mutation-selection mechanism of Darwinian evolution can be seen as one variant or another of the local search algorithms in Section 3, with the given link structure. Although we do not know the precise details of this algorithm, let us call it A.

Dembski's (2002a) application of NFL now says that

$$\begin{array}{c}\textit{if the fitness function } f \textit{ is generated at random according} \\ \textit{to uniform distribution among all the } |S|^{|V|} \textit{ possibilities,}\end{array} \tag{7}$$

then the Darwinian algorithm A cannot be expected to fare any better than blind search, and will therefore almost certainly fail to produce specified complexity (the odds against it suceeding to do so are Vast).

Phrased in this way, the result is pretty much correct.[9] Its relevance to evolution depends, however, on the extent to which (7) reflects properties of the true fitness landscape. We could, if we wanted to, dismiss Dembski's application as irrelevant on the grounds that no physical or biological mechanism motivating (7) has been proposed.[10] But that would, in my mind, be to make things a bit too easy, because even if no candidate for such a mechanism is available, Dembski's NFL argument would still pose an interesting challenge to evolutionary biology provided that empirical evidence shows that the true fitness landscape is similar to what one would expect to see under assumption (7).[11]

A minimum requirement, however, for the NFL argument to merit taking seriously, is that the actual fitness landscape exhibits at least some rough

[8] This ignores inversions, gene duplications, and other kinds of macromutations. It also ignores the recombination mechanisms of sexual reproduction. Still, it provides a good enough model of evolution to make my point clear.

[9] This statement is still somewhat charitable to Dembski, as it ignores his confusion (remarked upon in Section 1) concerning what specified complexity actually means; see Wein (2002a).

[10] There certainly isn't any a priori reason to expect that the "blind forces of nature" should produce a fitness landscape distributed according to (7). Anyone reasonably experienced in probabilistic modelling in science knows that such uniform distributions have no privileged status over other models as realistic descriptions of what the laws of nature produce, and that in fact only rarely do they turn out to provide good models for physical or biological systems.

[11] In the hypothetical scenario that we had strong empirical evidence for the claim that the true fitness landscape looks like a typical specimen from the model (7), then this evidence would in particular (as argued in the next few paragraphs) indicate that an extremely small fraction of genomes at one or a few mutations' distance from a genome with high fitness would themselves exhibit high fitness. It is hard to envision how the Darwinian algorithm A could possibly work in such a fitness landscape.

resemblence with what one would expect to arise from a model based on (7). Alas, it does not. I will now show that any reasonably realistic model for the actual fitness landscape will produce something that is very, very different from what (7) produces.

From the characterization (3) that we established in Section 5, we see that under assumption (7), the fitnesses of any two DNA sequences (or any collection of them, for that matter) are independent – a complete disarray. It follows that, with overwhelming probability, a fitness landscape produced by (7) will exhibit no significant tendency for neighboring DNA sequences to give any more similar values than do two totally unrelated DNA sequences.[12]

On the other hand, any realistic model for a fitness landscape will have to exhibit a considerable amount of what I would like to call *clustering*, meaning that similar DNA sequences will tend to produce similar fitness values much more often than could be expected under model (7). In particular, if we take the genome of a very fit creature – say, you or me – and change a single nucleotide somewhere along the DNA, then we expect with high probability that this will still produce an organism with high fitness. In contrast, under assumption (7), changing a single nucleotide is just as bad as putting together a new genome from scratch and completely at random, something that we have already noted (see Section 4) will with overwhelming probability produce not just a slightly less fit creature, but no creature at all. (If the true fitness function had this property, then, given the human mutation rate, none of us would be around.)

Thus – and since fitness landscapes produced by (7) are extremely unlikely to exhibit any significant amount of clustering – we can safely rule out (7) in favor of fitness landscapes exhibiting clustering, thereby demonstrating the irrelevance of Dembski's NFL-based approach. Note in this context that it is to a large extent clustering properties that make local search algorithms (such as A) work better than, say, blind search: the gain of moving from an $x \in V$ to a neighbor y with a higher value of f is not so much this high value itself, as the prospect that this will lead the way to regions in V with even *higher* values of f.

One minor issue that we touched upon at the end of Section 4 remains to be considered, namely the idea that *if* the true fitness landscape f deviates in important respects from typical behavior under the averaging (7), *then* there is something fishy about f that cannot be explained without invoking

[12] For further discussion and an illustration of this lack of structure of fitness landscapes produced by (7), see Section 7 of the earlier version Häggström (2005) of the present paper.

an intelligent designer. Such an idea is hinted in Dembski (2002a), Section 4.4 and elsewhere. But the clustering property of f that demonstrates the irrelevance of results derived under model assumption (7) hardly requires such mysterious explanations. A much more sensible mode of explanation is to view it as a manifestation of the phenomenon "like causes tend to have like consequences" that is prevalent throughout science and elsewhere.[13]

7 Remarks on extensions

Other than Wein, one of the most ardent public critics of Dembski's *No Free Lunch* is the well-known evolutionary biologist H. Allen Orr (2002, 2005). His criticism in these mostly pertinent contributions fails, however, to identify the preponderant shortcoming of the NFL application outlined in Section 6, and some of his more mathematical concerns are unconvincing. In particular, in Orr (2002), it is claimed that the the NFL arguement does not apply when the function f changes over time (corresponding to an evolving fitness landscape). But in fact, Wolpert and Macready (1997) have a variant of the basic NFL theorem for precisely such cases, and this variant can be plugged into Dembski's argument to give an evolving-fitness-landscape analog of his constant-fitness-landscape result. Such a modified Dembski argument is vaguely hinted at in *No Free Lunch*, but the fact of the matter is that it fails to be relevant to biological evolution, for very much the same reasons as those outlined in Section 6.

In Orr (2005), it is instead claimed that NFL does not apply to the situation of two or more coevolving species.[14] But again, although I have not been able to find in the literature an NFL theorem adapted to this situation, it is easy to devise one[15] and plug it into Dembski's argument. But yet again, the story is the same as in the evolving-fitness-landscape setting: the arguments in Section 6 show that also this extension lacks relevance to evolution.

Acknowledgement. I am grateful to Mats Rudemo for showing me Wein (2002a), to Timo Seppäläinen for scrutinizing the manuscript, and to an anonymous referee for valuable advice on how to make the paper more suitable for its target audience.

[13] See Häggström (2005) or Wein (2002a) for further elaboration of this point.

[14] The claim seems to originate from Wolpert (2002).

[15] See Häggström (2005), Footnote 19.

Bibliography

Aarts, E. and Lenstra, J.K., eds. (1997) *Local Search in Combinatorial Optimization*, Wiley, Chichester.

Behe, M. (1996) *Darwin's Black Box*, Free Press, New York.

Brockman, J. (2006) *Intelligent Thought: Science versus the Intelligent Design Movement*, Vintage, New York.

Crews, F. (2001) Saving us from Darwin, *New York Review of Books*, Oct 4 and Oct 18.

Dembski, W.A. (1998) *The Design Inference: Eliminating Chance Through Small Probabilities*, Cambridge University Press.

Dembski, W.A. (2002a) *No Free Lunch: Why Specified Complexity Cannot Be Purchased without Intelligence*, Roman & Littlefield, Lanham, MA.

Dembski, W.A. (2002b) Obsessively criticized but scarcely refuted,
http://www.designinference.com/documents/05.02.resp_to_wein.htm

Dembski, W.A. (2002c) The fantasy life of Richard Wein: a response to a response,
http://www.designinference.com/documents/2002.06.WeinsFantasy.htm

Dembski, W.A. (2005) Searching large spaces: displacement and the no free lunch regress,
http://www.designinference.com/documents/2005.03.Searching_Large_Spaces.pdf

Dennett, D.C. (1995) *Darwin's Dangerous Idea*, Simon & Schuster, New York.

Häggström, O. (2005) Intelligent design and the NFL theorems: debunking Dembski, http://www.math.chalmers.se/~olleh/Dembski.pdf .

Ho, Y.C. and Pepyne, D.L. (2002) Simple explanation of the no-free-lunch theorem, *Journal of Optimization Theory and Applications* **115**, 549–570.

Kallenberg, O. (2005) *Probabilistic Symmetries and Invariance Principles*, Springer, New York.

Kauffman, S. (2000) *Investigations*, Oxford University Press.

Orr, H.A. (2002) Book review: No Free Lunch, *Boston Review*, summer issue.

Orr, H.A. (2005) Devolution, *The New Yorker*, May 30.

Paley, W. (1802) *Natural Theology: Evidences of the Existence and Attributes of the Deity Collected from the Appearances of Nature*, reprinted by Lincoln-Rembrandt, Charlottesville, VA, 1986.

Pennock, R.T. (2001) *Intelligent Design Creationism and its Critics: Philosophical, Theological, and Scientific Perspectives*, MIT Press, Cambridge, MA.

Shallit, J. (2002) Book review: No Free Lunch, *BioSystems* **66**, 93–99.

Wein, R. (2002a) Not a free lunch but a box of chocolates,
http://www.talkorigins.org/design/faqs/nfl/

Wein, R. (2002b) Response? What response?
http://www.talkorigins.org/design/faqs/nfl/replynfl.html

Wolpert, D.H. (2002) William Dembski's treatment of the no free lunch theorems is written in jello, `http://www.talkreason.org/articles/jello.cfm`

Wolpert, D.H. and Macready, W.G. (1997) No free lunch theorems for optimization, *IEEE Transactions of Evolutionary Computation* **1**, 67–82.