

Simulation of Biological Evolution and the NFL Theorems

Ronald Meester*

July 21, 2008

Abstract

William Dembski (2002) claimed that the NFL theorems from optimization theory render darwinian biological evolution impossible. Häggström (2006) argued that the NFL theorems are not relevant for biological evolution at all, since the assumptions of the NFL theorems are not met. Although I agree with Häggström (2006), in this article I argue that the NFL theorems should be interpreted as dealing with an extreme case within a much broader context. This broader context *is* in fact relevant for scientific research of certain evolutionary processes; not in the sense that the theorems can be used to draw conclusions about any intelligent design inference, but in the sense that it helps us to interpret computer simulations of evolutionary processes. As a result of this discussion, I will argue that from simulations, we do not learn much about how complexity arises in the universe. This position is in contrast with certain claims in the literature that I will discuss.

Keywords and phrases: evolution, mathematical model, no free lunch theorems, simulation, arise of complexity.

1 Introduction

The *No Free Lunch Theorems* (NFL theorems) were introduced in the biological field by William Dembski in his book *No Free Lunch* (2002), with the goal to show that darwinian evolution is incapable of producing certain complex structures. In his book (and elsewhere), Dembski then argues that these complex structures must have been *designed* by some intelligence, thereby

*Department of Mathematics, VU-University Amsterdam, De Boelelaan 1081, 1081 HV Amsterdam, The Netherlands; email: rmeester@few.vu.nl

following Michael Behe's best-selling *Darwin's Black Box* (1996). Dembski and Behe are two major figures in the *Intelligent Design* (ID) movement; a movement which claims that nature bears the unmistakable signs of intelligent design.

In this article, I will argue that the NFL theorems do play a role in the discussion about evolution, albeit a somewhat different role than Dembski envisioned. This does not imply that I defend ID in any way; I would like to emphasise this from the outset. Indeed, I have criticized Dembski's intelligent design inference in many other publications before, see e.g. Meester (2003).¹ I certainly believe that we have been produced by some evolutionary process, and this is precisely the reason that I am very much interested in scientific research of biological evolutionary processes, especially when my own discipline - mathematics and probability theory in particular - comes into play. And mathematics comes into play very much as soon as we make *models* of evolutionary processes and even more so when we *simulate* such processes on a computer. Simulating evolution is a genuine branch of research, with research papers doing *only* simulations making it to *Nature*; see for instance Lenski *et al.* (2003) for a typical example to which I will return later in this article.

The (very reasonable) objections against any inference of intelligent design have been so strong and emotional (see for instance Orr (2002) for a typical reaction), that most people didn't really take the trouble to think deeper about certain claims. In this article then, I want to explain why I think that the NFL theorems give rise to interesting considerations concerning the interpretation of simulations of evolutionary processes, without - of course - defending or making any intelligent design claim.

The article is built up as follows. First, I want to explain what the NFL theorems say, and how this can be connected to evolutionary processes. I will illustrate the NFL theorems with some toy-examples, to make the reader somewhat familiar with their use, and also to anticipate the illustration of my point later on. I briefly sketch the reasoning of Dembski (2002) leading to his claim that these NFL theorems disprove darwinian evolution. After that, I will summarise some of the findings of Haggstrom (2006), who convincingly showed that the NFL theorems themselves are not relevant for biological evolution at all. However, I will put the NFL theorems into a somewhat broader framework which *is* relevant for evolution. I will then argue that this broader framework still has not much to say about the real evolution, but at the same time, that it does say something about simula-

¹This publication is in Dutch.

tions of evolutionary processes. Indeed, I will argue, based on the above mentioned broader framework, that simulations can not tell us very much about the question whether or not darwinian biological evolution is capable of creating complex features into the universe. This position is in contrast with certain claims in the literature that I will discuss.

2 The NFL theorems; theory and some examples

A certain modest amount of mathematical notation is unavoidable in this section, but I will make it as light as possible.

Consider a finite set V and a function $f : V \rightarrow \mathbb{R}$ (the set of real numbers) which assigns a real number to each element in V . Suppose we want to find an element $x \in V$ for which $f(x)$ is maximal. This may seem like an easy problem, but typically, V is extremely large, so a one by one search to find an $x \in V$ with maximal value of $f(x)$ would take much too long.²

Therefore, in optimization theory, one tries to set up some *search algorithm* to find an $x \in V$ that perhaps does not really maximize $f(x)$, but for which $f(x)$ is high in a certain sense; for instance, one often tries to find an x which satisfies $f(x) \geq t$, for some given t . (See for instance Aarts and Lenstra (1997) for much more information on algorithmic search.)

In our context, such algorithms have the following general form. Let f be a function as above (very often, f is referred to as a *fitness function* and we will adopt this terminology), and define a *target set* $T \subset V$, say a set of the form $T = \{x \in V : f(x) \geq t\}$. First we choose an element $x_1 \in V$ according to some rule (not involving the function f but possibly involving additional randomness) and we compute $f(x_1)$. Given some rule that can only take into account x_1 and $f(x_1)$ (and additional randomness), another point x_2 is chosen, different from x_1 , and $f(x_2)$ is computed. After k steps, we have k points x_1, \dots, x_k and in addition the values $f(x_1), \dots, f(x_k)$. The algorithm then goes on to choose a *new* point x_{k+1} which has not been selected before, according to some rule that only involves $x_1, \dots, x_k, f(x_1), \dots, f(x_k)$ and possibly additional randomness. After that, $f(x_{k+1})$ is computed, et cetera.

If we choose the next point x_k uniformly at random among all points that have not been chosen before, that is, in such a way that each of the points not yet chosen have equal probability to be chosen, then we call the resulting algorithm *blind search*, for obvious reasons. Often though, the rule according to which one chooses the next x_k uses some structure of the set

²For instance, if V is the set of possible ways in which we can line up, say, 150 objects, then there is no way in the world that we can *ever* calculate $f(x)$ for every $x \in V$.

V . For instance, if V consists of pairs (k, l) for integers k and l , then one could, for example, choose the next x_k among neighbors of points already chosen. Some examples follow soon.

The stage is now set for the NFL theorem (see Wolpert and Macready 1997).³ They state that in a certain average sense, no search algorithm can be better than any other. In their (and our) setting, the functions f are only allowed to take values in a finite set S . Since also V is finite, there exist only finitely many functions $f : V \rightarrow S$, and the NFL theorem is concerned with a certain average over *all* these functions. To formally state the result, let E_k be the event that at least one of the points x_1, \dots, x_k falls into the target set T . When we take $T = \{x \in V : f(x) \geq t\}$, this boils down to the event that at least one of the recorded values $f(x_1), \dots, f(x_k)$ has a value at least t .

The NFL theorem. *Let $p_{A,f}$ be the probability of the event E_k when using the function f and search algorithm A . When we average these numbers $p_{A,f}$ over all f , the outcome is independent of the search algorithm A .*

The conclusion of the theorem seems quite surprising. It says that no algorithm is better than any other at quickly finding a target set T . In particular, no search algorithm is better than the blind search algorithm, in which the next x_k is simply chosen uniformly at random among all points that have not been chosen before. Put in yet other words: we cannot expect a search algorithm to be efficient unless we restrict ourselves to functions f that are distinguishable from the “average” f , and I believe that this last formulation is a concise description of the importance of the NFL theorem.

Surprising as the result may sound at first sight, the NFL theorem is in fact almost a mathematical triviality once you see things from the right angle. I do not intend to explain this here, and I refer to Häggström (2006) for an excellent explanation as to why this is the case.

Before we continue, let me very briefly sketch why Dembski thinks that the NFL theorem renders darwinian evolution impossible. It is fairly common to view evolution as an algorithmic search procedure (see e.g. Dennett (1995)), or as an optimization problem in which we want to maximize a fitness function f which assigns, for instance, a fitness to each DNA sequence. The search consists of changing the DNA sequence at certain locations, leading to a new element in the space of DNA sequences (our space V). Since the “target set” of DNA sequences which correspond to living organisms is extremely small compared to the total number of DNA sequences, blind

³There are many similar results, but in what follows, we will be concerned with only one of them.

search will essentially never find it. Therefore, since according to the NFL theorem, no algorithm works better than blind search, darwinian evolution is impossible, unless one “leads” the algorithm to the target set by using a very special fitness function or algorithm. This however would, according to Dembski, be cheating since the search of such a special algorithm or fitness function would be at least as difficult as the search for the target set itself in the first place. (Dembski coins this the *displacement problem*; see Dembski (2002), Section 4.7.)

I do not disagree with Häggström concerning his criticism on this argument. Indeed, it is simply not the case that a biological fitness function can be viewed as an average over all possible fitness functions. Indeed, had this been the case, then a simple mathematical argument shows that the fitnesses of two DNA sequences which differ in only one position, would have been completely independent of each other, and this is clearly not the case.⁴ Therefore the NFL theorem simply does not apply.

This sounds as the end of the story, and if this were the case there was no need to write the present paper. However, the story does not end here and in order to make my point later, I will now first discuss two simple examples of the NFL theorem in action.

Example 1. Let V consist of only 3 points a , b and c . Suppose in addition that S consist of 2 points, say 0 and 1. Then there are only 8 different functions from V to S , which we can represent by the 8 triplets of 0’s and 1’s. We distinguish two different search procedures A and B : A will be blind search, that is, each next point will be chosen at random among the remaining ones, whereas B searches in a fixed order a , b and c . We are interested in the probability that the search algorithm finds a point $x \in V$ with $f(x) = 1$ in at most two steps. For the algorithms A and B we denote these probabilities by $p_{A,f}$ and $p_{B,f}$ respectively. In the next table, we write all 8 functions f , together with the values of $p_{A,f}$ and $p_{B,f}$. For instance on the second line, blind search will find the point c with probability $2/3$, whereas under algorithm B , the only point (namely c) where f takes the value 1, will not be found and hence the 0 in the last column. Note that although $p_{A,f}$ and $p_{B,f}$ are sometimes different for the same f , the *average value* over all 8 functions is equal to $3/4$ in both cases, in accordance with the NFL theorem.

⁴See Häggström (2006) for a full and detailed explanation of this claim.

number	$f(a)$	$f(b)$	$f(c)$	$p_{A,f}$	$p_{B,f}$
1	0	0	0	0	0
2	0	0	1	2/3	0
3	0	1	0	2/3	1
4	0	1	1	1	1
5	1	0	0	2/3	1
6	1	0	1	1	1
7	1	1	0	1	1
8	1	1	1	1	1
average				3/4	3/4

Table 1. The 8 functions, together with $p_{A,f}$ and $p_{B,f}$.

Example 2. Let V consist of all sequences of letters of the alphabet of length 3. Consider one such sequence, for instance the word YES, and defined a fitness function $f : V \rightarrow \mathbb{R}$ as follows: For any word $\alpha\beta\gamma \in V$, $f(\alpha\beta\gamma) = 1$ if only the first letter matches with the word YES; $f(\alpha\beta\gamma) = 2$ if only the second letter matches; $f(\alpha\beta\gamma) = 4$ if only the third letter matches. Furthermore, if two or more letters match, we add the values of the matching letters. So for instance, $f(FEW) = 2$, $f(YEW) = 3$, $f(RES) = 6$ and $f(YES) = 7$; the word YES maximizes this fitness function. This fitness function is chosen in such a way that we can read off from its value not only how many letters match with YES but also which ones. For instance, if we have a word with fitness 5, then we know that the first and last letter match, and the second letter does not.

Given this fitness function, we can design a search algorithm that actually leads us to the word YES (which has maximal fitness 7) rather quickly. Indeed, if at some moment in time, we have a word with fitness 5 say, then we choose our next word by fixing the first and third letter (since these are already correct), and choosing a new second letter at random among all remaining possibilities that we have not been seen before. In not too many steps we will end up with the word YES, having maximal fitness. Note that this algorithm uses the value of f at the current point, unlike the algorithms of Example 1.

It is clear that there is nothing special about the word YES, and we can, by choosing the appropriate fitness function, direct a similar algorithm to any desirable final outcome. If you use the “wrong” fitness function, that is, not corresponding to the target word, then you may never end up at the target. The NFL theorem asserts that if you consider the collection of all fitness functions with values in $\{0, 1, 2, 3, 4, 5, 6, 7\}$ and average over all

these functions, then the (average) probability to reach a target with fitness 7, say, does not depend on the precise algorithm we use.

3 How to interpret an (in)efficient algorithm

Let us consider Example 1 again, and suppose that someone tries to locate a point in V with fitness 1 in at most two steps of a certain (unknown to us) search algorithm. We know from the example that for any such search algorithm, on average (averaged over all fitness functions) the probability to be successful is $3/4$. Now suppose someone repeats the search many times, and as it turns out, he is successful every time. What conclusion would we draw? Well, first of all, we would conclude that this person does not choose his fitness function uniformly at random at each search, since had he⁵ done that, the success probability would have been $3/4$. So we know that he chooses his fitness functions differently, although we do not really know how. He can choose the same function each time, or choose a different one each time but not according to a uniform distribution in which each function has equal probability of being chosen. Let us for simplicity assume that he uses the same fitness function all the time, say the fifth function given by $f(a) = 1$ and $f(b) = f(c) = 0$.

However, given that he uses this fitness function, there is still freedom in the search algorithm that he uses. In this particular example, it would be unwise to use B , since then the success probability is only $2/3$. No, *if* the fifth fitness function is used, *then* the choice of the algorithm must be *tailored around this choice* in order to get an efficient algorithm. In the present example, algorithm B must have been used in order to be successful every time. The point I want to make, then, is that it will typically be the case that for a given (class of) fitness function(s), different algorithms will behave very differently, and one has to make the “correct” choice in order to get an efficient algorithm.

A similar reasoning applies to Example 2. If we find that a certain researcher finds the word YES over and over again, then we conclude that his algorithm is too efficient to be the result of averaging over all fitness functions; it is not likely that he chooses his fitness function uniformly at random over all possibilities at the start of each new search. No, it is reasonable to conclude that he uses the fitness function corresponding to the word YES, *and* that he uses the search algorithm associated to that word. Again, note that the conclusion is two-fold: we know that he uses special

⁵I use “he” where I strictly speaking should use “he or she”.

fitness functions *and* we know that his search algorithm is tailored around this choice in order to get an efficient algorithm.

These remarks sound like very obvious ones, and in a way they are. Once a search algorithm is more (or less, for that matter) efficient than what you expect from the NFL theorem, it *must* be the case that you use a special choice, or a special class, of fitness functions. But at the same time, you must use a careful choice of the search algorithm which must have been tailored around your choice of (the class of) fitness functions. The point is that if you average over all fitness functions, then it doesn't matter which algorithm you take - this is precisely what the NFL theorem says. But if you choose your fitness function differently, then it *does* matter a lot which algorithm you choose, as is illustrated by the examples above.

There is another way to look at this, as follows. The NFL theorem deals with a situation in which we average over all fitness functions. If we average, then it does not matter at all which algorithm we use, that is what the theorem says. As such, the theorem deals with an extreme situation: averaging over *all* fitness functions. The other extreme is that we fix *one specific* fitness function. In that case, as we saw above, it matters a lot which algorithm one uses: only very special algorithms will be efficient in that case. Between these two extremes, numerous possibilities exist: we may for instance look at a certain class of fitness functions. It is not to be expected that one can write down interesting general mathematical theorems for these "intermediate" cases. Nevertheless, it is very reasonable to say that there is a certain tendency which makes the choice of the algorithm *more* significant when we *reduce* the possibilities for our fitness function.

So this is the conclusion that is connected to the NFL theorem (I emphasise that this conclusion is not *part* of the mathematical theorem itself): when a certain algorithm is efficient in combination with a (class of) fitness function(s), then the algorithm must have been chosen very carefully. Note the similarity with what Dembski called the "displacement problem", but also note the subtle but important difference between his and my conclusion. I draw my conclusion from a broader perspective in which the NFL theorems deals with one end of the spectrum only.

4 Application to evolution

What do we learn about evolution from this discussion? Let me first make a general remark about the role of mathematics in this context. Apart from the problems that were noted by Häggström (2006), the ideas in *No Free*

Lunch suffer from a too optimistic belief in the usefulness of mathematical models in biology (see also Olofsson (2007) for a similar conclusion), and this remains true in the context of the present article. The mathematical concepts that are used can of course be given a biological interpretation; we already mentioned this before. Indeed, the space V can be the space of DNA sequences (up to a certain length, say) and the fitness function f can - perhaps - be interpreted as a real fitness function. But what about the search algorithms? I do not think it is reasonable to summarise the extremely complex biology (and chemistry, physics . . .) that is associated to the process, into a single search algorithm. There are no realistic models of evolution at the molecular level that render this approach reasonable. Computing probabilities in a model is one thing, but for these computations to have any implication, the models had better be very good and accurate, and it is obvious that the various models do not live up to this requirement. In particular, it is quite meaningless to compute the probability that certain amino acids combine to produce a particular molecule, if there is no *realistic* mathematical model (on the molecular level) around.

In addition, in my discussion of the NFL theorem and beyond, I distinguished between algorithms and fitness functions; these concepts exist more or less independently of each other. In real life however, the process itself and the space on which this process acts, do not exist independently of each other. The evolutionary process itself generates DNA sequences which we took as our space V before. This means that the evolutionary picture is not nearly as simple and clear as suggested by Dembski, and I find it hard to imagine how the discussion in this article could apply directly to biology in any meaningful way.

This having said, I now want to argue that the algorithmic “NFL-way” of thinking about evolution is very meaningful when it concerns *computer simulations* of certain evolutionary processes. Unlike real evolution, computer simulations sometimes fit very well into the formal mathematical setting as discussed in this article, and the discussion applies to such simulations directly.

Before discussing an important example in the literature, let me first go back to Example 2 above. Suppose we are confronted with a scientist who claims to obtain the word YES over and over again. We can then be sure that the algorithm he uses is the one corresponding to the word YES; it is the algorithm that was described in detail in Example 2. It is hardly reasonable to say that when we simulate Example 2 on a computer, the process reached the final outcome with highest fitness *by itself*. No, given the fitness function (that is, given the word YES) the computer programmer chose an efficient

algorithm that would lead to YES. In other words, the programmer chose his program with insight in the future goal he wanted to reach, and hence the simulation was intrinsically teleological.

Let us now look at a much more sophisticated example of the same type. In Lenski *et al.* (2003), computer simulations are used to explain the evolutionary origin of complex features. In their work, *digital organisms* - computer programs that self-replicate, mutate and compete - evolve, and Lenski *et al.* wondered whether the programs would eventually evolve into programs that can perform certain logical operations. The fitness of a program very much depends on the operations it can handle. If at some point in time, a program can execute a given logical operation, it is rewarded with an amount of “energy”, which in this model boils down to a certain amount of processing time; this can be seen as the fitness of the organism. Lenski *et al.* noted that this complex system of interacting components produced (digital) organisms that could execute complicated logical operations that were built from simpler ones, and that the organisms with the highest fitness evolved via organisms with low or intermediate fitness. In the course of the process, they observed variation, selection and heritability in their digital organisms, very much analogous to the way evolution is supposed to work in real life. Some solutions found by the process were even unanticipated by the researchers. This made them draw the following conclusions:

These findings show how complex functions can originate by random mutation and natural selection.⁶

and

In closing, digital organisms provide opportunities to address important issues in evolutionary biology. They are particularly well suited to problems that are difficult to study with organic forms owing to incomplete information, insufficient time and the impracticality of experiments.⁷

How reasonable are their conclusions? First of all, let me emphasise that their article is a pleasure to read, and that I am very much impressed by the way they set up the simulation. It is, from this point of view, an impressive piece of work. But does their conclusion follow? To shed some light on this, let us discuss their approach in the language of the present article.

First of all, their simulation is a search algorithm in our sense, with V being a space of certain computer programs (their digital organisms).

⁶Lenski *et al.* (2003) page 139.

⁷Lenski *et al.* (2003) page 143.

Furthermore, their fitness function is chosen in such a way that only a very small part of V has a high fitness; the target set is very special indeed. Finally, their simulation is set up with the goal to show that the search algorithm can reach this very special set with high fitness.

All this taken together means that this simulation is in the domain where the discussion of the previous sections applies. It follows, therefore, that the target set can only be reached if the search algorithm is very carefully tailored around their fitness function. And, of course, this tailoring is precisely what happened: many program parameters must be fixed, and it should be clear that most “version” of any such algorithm would not produce anything interesting at all. In other words, it is certainly not the case that one can say that the algorithm found a digital organism with a high fitness “by itself”. The programming must have been done with insight into the future goal that they wanted to reach.

Of course, not everything that was observed was actually designed by the researchers. For instance, the researchers did not specify in advance what route would be followed towards programs with a high fitness. Also, as already mentioned before, the researchers noted that unanticipated solutions were sometimes found by the process. However, both these effect (the route not being controllable and the emergence of unanticipated solutions) can already be observed in a toy example like Example 2. Indeed, the actual path leading to the word YES is pretty much random and uncontrollable; we know that we will eventually obtain the word YES but we do not know - for instance - which of these three letters will be fixed first. And if we would search for a word of fitness 6, BES, say, then it is not a very big surprise that in the process we might bump into the unique word YES of fitness 7 along the way - this will happen in a significant fraction of the experiments.

Hence the careful choice of the algorithm does not - and should not, for that matter - rule out unanticipated solutions and random routes towards these solutions. Yet despite this, according to our earlier discussion, their algorithm can only work if it is tailored carefully around their fitness function, so the target must have been set in advance (and - of course - this was the case). Indeed, if your goal is sufficiently “special”, that is, if your fitness function assigns high fitnesses to a very small part of the full space of possibilities, then you cannot reach the target by chance, and your algorithm has to be intrinsically teleological. The discussion around the NFL theorem in the previous sections forces this conclusion: if you reach something “very special” in your simulations, then you must have programmed the computer with insight in the future.

Let us now return to the two quoted conclusion in Lenski *et al.* Is

it true that “these findings show how complex functions can originate by random mutation and natural selection”? Let me first remark that this conclusion should be interpreted as dealing with the real biological evolution, otherwise there does not seem to be a point. Indeed, if your goal would be to explain why mutation and selection can create complex features that you *specify in advance*, then the conclusion is clearly correct and uncontroversial. But do these findings explain the “spontaneous”, or *de novo* emergence of complex features in nature? I do not see how such an explanation could possibly follow. The algorithm takes darwinian evolution as a starting point, and it is (correctly) argued that variation, selection and heritability are enough to reach a certain well-defined and previously set target. However, the discussion around the NFL theorem demonstrates that in a computer simulation which constitutes a search algorithm, a “special” target can only be reached when you carefully choose your algorithm around the target, that is, if the algorithm is intrinsically teleological. And of course, darwinian evolution is not teleological, by definition it cannot be! This conclusion does *not* imply that I do not believe in darwinian evolution, but it does imply that computer simulations can not be used to show how complex features arise in the universe.

5 Discussion, scope and conclusion

The above discussion only applies to simulations that constitute a search algorithm, and which involve trying to reach a sufficiently special target set. Indeed, only in these circumstances does the application of the NFL theorem lead to the conclusion that the programming (that is, the choice of the algorithm) must have been done with insight into the future. Needless to say, there are many simulations - also in an evolutionary context - which do not fall into this category, and whether or not a computer simulation is meaningful very much depends on the question one tries to answer. For instance, it is uncontroversial to use simulations to shed light on the development of antibiotic resistance, or on how population genetics are affected by different geographical structures and mating systems. Why is simulation not controversial in these cases? Well, in the two examples just mentioned, the point is not to reach a special target, but instead to compare the “typical” behavior of related systems. This means that the problems related to the NFL theorem simply do not enter the scene. Simulations in these circumstances are very useful, even though the results of such a simulation are often quite predictable. Nevertheless, it sometimes happens that simu-

lations show unanticipated behavior, which forces one to think about what is going on. Many evolutionary questions can be investigated with simulations when - for instance - the mathematical models are too complicated for a rigorous mathematical analysis, and they confirm that the concept of darwinian evolution, with all its vital ingredients, is a sensible one, and that it is a good thing to think about reality in those terms.

There is one more point I want to make, just to make sure that I am not misunderstood. I do not claim - of course - that a simulation can only be meaningful if there is no design in the simulation. Indeed, it is impossible to simulate without designing a program. Often this is no problem, but if the whole point of your simulation is to show that complexity can arise in the universe in a darwinian (and therefore non teleological) way, then it does become a problem, since the above discussion applies and shows that any successful computer program must be intrinsically teleological.

I can elaborate on this in connection to the final conclusion in Lenski *et al.* that (I repeat) “digital organisms provide opportunities to address important issues in evolutionary biology. They are particularly well suited to problems that are difficult to study with organic forms owing to incomplete information, insufficient time and the impracticality of experiments”. It is hard to disagree with the statement itself, but again it is the question as to what extent one is allowed to draw any conclusions about the real emergence of complex features from these simulations. To put the conclusion in a somewhat different style: if one wants to argue that there need not be any intelligent design in nature, then it is hardly convincing that one argues by showing how a *well-designed* and teleological algorithm behaves as real life is supposed to do.

Hence, the final conclusion can be formulated as follows: when we recognize the NFL theorem as dealing with an extreme case within a much larger class of situations, we see that in a search algorithm as defined in the present article, a sufficiently special target set can only be reached when the search algorithm is very carefully tailored around the fitness function. This conclusion is a direct consequence of our discussion of the NFL theorems and beyond. This implies that this special target can only be reached with insight into the future. Since darwinian evolution cannot look into the future, this leaves no other room than to conclude that simulations can not be used for the purpose of explaining how complex features arise into the universe. There certainly are other evolutionary problems which can be addressed with simulations in a non-controversial way, as long as they are not concerned with reaching special targets. I think it is very important to be aware of this problem, so that we are able to better interpret computer

simulations for evolutionary problems in the future.

Bibliography

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

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

Dembski, W.A. (2002) *No Free Lunch: why specified complexity cannot be purchased without intelligence*, Roman & Littlefield, Lanham, MA.

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

Hägström, O. (2007) Intelligent design and the NFL theorems, *Biology and Philosophy* **22**, 217-230.

Lenski, R.E., Ofria, C., Pennock, R.T. and Adami, C. (2003), The evolutionary origin of complex features, *Nature* **423**, 139-144.

Meester, R. (2003), *Het pseudoniem van God*, Ten Have Baarn.

Olofsson, P. (2007) Intelligent design and mathematical statistics: a troubled alliance, *Biology and Philosophy*, to appear.

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

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