The No Free Lunch and Realistic Search Algorithms

Yossi Borenstein
University of Essex
Colchester, U.K
yboren@essex.ac.uk

Riccardo Poli
University of Essex
Colchester, U.K
rpoli@essex.ac.uk

ABSTRACT

The No-Free-Lunch theorems (NFLTs) are criticized for being too general to be of any relevance to the real world scenario. This paper investigates, both formally and empirically, the implications of the NFLTs for realistic search algorithms. In the first part of the paper, by restricting ourselves to a specific performance measure, we derive a new NFL result for a class of problems which is not closed under permutations. In the second part, we discuss properties of this set which are likely to be true for realistic search algorithms. We provide empirical support for this in [1].

Categories and Subject Descriptors

F.2 [**Theory of Computation**]: ANALYSIS OF ALGORITHMS AND PROBLEM COMPLEXITY

General Terms

Algorithms,Theory

Keywords

No Free Lunch, Genetic algorithms, Heuristics, Theory

1. INTRODUCTION

The publication of the No-free-lunch theorems (NFLTs) [2] led to an ongoing debate about its applicability to real-world problems. In particular, the main argument against the NFLT is its generality: it applies to the class of all possible problems. Hence, the problems on which some algorithms may fail will be, in most cases, artificial. In this paper we attempt to connect the general framework of the No Free Lunch Theorems (NFLTs) to real-world problems.

Let $\{f_t\}$ denote the set of fitness values that the algorithm sampled during a run. The NFLTs consider *any* performance measure which depends on this set. While this is general enough to account to many possible performance measures, more often than not, the only *relevant* performance measure is the number of fitness evaluations it takes to find an optimum point. In section 2, we restrict our attention to this type of performance measures and derive a new NFL result for a class of functions which is not closed under permutation [3].

The second part (section 3), which is the main result of the extended version of the paper [1], analyzes the properties of this set considering realistic search algorithms. It is mainly based on simple observations regarding the nature of search for realistic algorithms which are supported by empirical results (given in [1]). Perhaps the basic component of the NFL is the notion that an optimizer has to "pay" for its superiority on one subset of functions with inferiority on the complementary subset. Our main observation is that approximately a NFL result might hold on a very small set of functions. In particular, if algorithm a outperforms random search on f_1 there exists an f_2 on which random search outperforms algorithm a to the same degree. In other words the performance of the algorithm over f_1 and f_2 is symmetric w.r.t. random search.

In this paper, we will formulate, for realistic search algorithms, a possible connection between these two complementary functions (f_1 and f_2). We will claim that this connection is general, i.e., that if this symmetry exists for (a, f_1, f_2) it will exist for any other realistic search algorithm as well. This suggests how the NFLT manifests itself in practice for realistic search algorithms.

2. NO FREE LUNCH FOR GLOBAL OPTIMIZERS

Let X be an ordered finite set and $f : X \to \mathbb{R}$ a fitness function. Let f_{max} denote the highest fitness value of f. This paper investigates the class of functions $F^X \equiv \{f^{x_i}\}$ where $x_i, x_j \in X$ and f^{x_i} is defined as follows:

$$f^{x_i}(x_j) = \begin{cases} f(x_j) & \text{if } x_j \neq x_i, \\ f_{max} + 1 & \text{otherwise.} \end{cases} \quad (1)$$

In other words, the function f^{x_i} is identical to f with the exception that x_i is set to be, explicitly, the global optimum. The performance of an algorithm a on a function f^{x_i}, $P(a, f^{x_i})$, is defined as the expected number of fitness evaluations it takes a to sample x_i (i.e., the global optimum) for the first time.

The performance of a search algorithm on the class F^X can be derived from its performance on the function f. In particular, let $h_f^a(x_i)$ denote the expected number of fitness evaluations it takes algorithm a to sample x_i for the first time when executed on f – in other words, the first hitting time for x_i.

Copyright is held by the author/owner(s).
GECCO'06, July 8–12, 2006, Seattle, Washington, USA.
ACM 1-59593-186-4/06/0007.

The following is a simple consequence of our definition:

LEMMA 1. *([1]) Let a be a search algorithm and $f : X \rightarrow \mathbb{R}$ a fitness function.*

$$\forall x_i \in X \; P(a, f^{x_i}) = h_f^a(x_i) \qquad (2)$$

The vector $\{h_f^a(x_i)\}$ corresponds, therefore, to the expected performance of a on the class F^X. The one-to-one correspondence between $P(a, f^{x_i})$ and $h_f^a(x_i)$ has interesting consequences. Let us consider the vector $\{h_f^a(x_i)\}$. If we restrict our attention to a non-resampling search algorithm (or alternatively to a resampling one which is charged only for distinct fitness evaluations) the set $\{h_f^a(x_i)\}$ represents the *order* in which the algorithm samples the search space (given f). Irrespectively of the fitness function, the average of $\{h_f^a(x_i)\}$ equals the value $(|X| + 1)/2$ – which is the *expected* performance of the algorithm on any function f. This implies a NFL result for the set F^X.

THEOREM 1. *([1]) Let $f : X \rightarrow Y$ be a fitness function with one global optimum and $P(a, f)$ denote the expected number of fitness evaluations required to sample the global optimum. For any non-resampling search algorithms, a_1, a_2 the following holds:*

$$\sum_i P(a_1, f^{x_i}) = \sum_i P(a_2, f^{x_i}) \qquad (3)$$

Constraining the way performance is measured we were able to derive a new NFL result for a set of functions which is not closed under permutation. We would like to focus, however, on the properties of the set $\{h_f^a(x_i)\}$ when it is induced by *realistic* search algorithms.

3. BEYOND THE NFL

The formal analysis of the NFL, or similarly, the new result in the previous section can only take us up to a certain depth. For example, if a is a deterministic non-resampling search algorithm, the vector $\{h_f^a(x_i)\}$ gives the *order* in which the algorithm samples the points of the search space. Take as an example the case where $|X| = 4096$, we can make several general observations. For example, irrespective of the algorithm, $\forall a E_x(h_f^a(x)) = 2048.5$ and hence all algorithms, on average, have the same performance. In particular, the expected performance of random search is exactly 2048.5. Similarly, for any function on which a outperforms random search (e.g. $h_f^a(x') = 2048.5 - 1048.5$) there exists a function on which random search outperform a to the same degree (e.g. $h_f^a(x'') = 2048.5 + 1048.5$).

The NFLT results cannot go beyond this point. For example, knowing the identity of x' in the example above, does not give us any knowledge about the identity of x''. In fact, this is the fundamental observation of the NFLT: because we have no a priori knowledge on the problem, x'' can be any solution from the set $X \setminus \{x'\}$. By restricting our attention to realistic search algorithms, even if we are unable to define formally what "realistic" means, the situation changes.

Let us define x_{min} to be the solution that a samples with minimum number of fitness evaluations – i.e. $x_{min} = \arg\min_x h_f^a(x)$. We suggest that the distance of a solution from x_{min} is a good *indicator* for the expected number of fitness evaluation it takes to sample it. Remember that this is equivalent, using our notation, to say that the performance of a on the *function* f^x is correlated to $d(x, x_{min})$ where $f^{x_{min}}$ is the function for which a has the best performance.

CONJECTURE 1. *Let a be a search algorithm and f a fitness function. Define $x_{min} = \arg\min_x h_f^a(x)$. $d(x, x_{min})$ is a good indicator to the first hitting time for x (i.e., $h_f^a(x)$). That is, the bigger the distance the longer the first hitting time.*

The reason for that is the tendency of reasonable search algorithms to sample points in the proximity of points with high fitness value. If the landscape is not random, it causes the algorithm to follow one (or more) trajectories. x_{min} indicates the general direction that the algorithm takes. It is our conjecture that the further a point is from this main path the less likely it is to be sampled.

As a consequence of this property, knowing $h_f^a(x_{min})$ we can now make predictions for the first hitting time of any solution in the search space. As our main concern is the relation with the NFLTs, we will focus on the symmetry that the NFLTs predict w.r.t. the performance of random search. We make the following two predictions:

1. Let n denote the diameter of the landscape (i.e. the biggest distance between two points). The expected performance of a on f^x for x such that $d(x_{min}, x) = n/2$ is that of random search.
2. Let $\overline{x} = \arg\max_{x'} d(x', x)$. The average of the expected first hitting times for x and $\overline{x}$ equals the performance of random search. That is $(h_f^a(x) + h_f^a(\overline{x}))/2 = (|X| + 1)/2$.

Following this line of reasoning we made in [1] further speculations which were supported empirically using mainly *genetic algorithms* but also *particle swarm optimization* and local search on several problems.

4. CONCLUSION

The NFLTs are criticized for being too general to be of any relevance to real-world problems. In this paper we showed (1) that by putting some constraints on the performance measure it is possible to derive formal NFL results which might be of more interest to the real-world scenario and (2) trying to compromise between the size of the problem set and the accuracy of the prediction, we were able to show that *approximately*, averaged over *any* arbitrary problem and its negation, the performance of realistic algorithms is similar.

5. ACKNOWLEDGMENT

This work was supported by the Anglo-Jewish-Association and the Harold Hyam Wingate Foundation.

6. REFERENCES

[1] Yossi Borenstein and Riccardo Poli. The no free lunch and realistic search algorithms. Technical report, University of Essex, 2006.

[2] David Wolpert and William G. Macready. No free lunch theorems for optimization. *IEEE Trans. Evolutionary Computation*, 1(1):67–82, 1997.

[3] C. Schumacher, M. D. Vose, and L. D. Whitley. The no free lunch and problem description length. In Lee Spector et al. editors, GECCO 2001. Morgan Kaufmann.