The Unrestricted Black-Box Complexity of Jump Functions

Maxim Buzdalov
ITMO University
Saint Petersburg, Russia

Benjamin Doerr
Laboratoire d'Informatique (LIX)
École Polytechnique
Palaiseau, France

Mikhail Kever
ITMO University
Saint Petersburg, Russia

ABSTRACT

We analyze the unrestricted black-box complexity of the n-dimensional JUMP function classes. We show very precise bounds for various values of the jump size ℓ, including a novel $n + \Theta(\sqrt{n})$ bound for the extreme case that only the middle one (for n even) or the middle two (for n odd) Hamming levels are not part of the plateau surrounding the optimum. To obtain these results, we significantly extend the classic information theoretic argument. It now allows to exploit structural properties of the underlying optimization problems, whereas before it relied only on the number of different fitness values.

This abstract for the GECCO'17 Hot-off-the-Press track summarizes work that appeared as M. Buzdalov, B. Doerr, and M. Kever. The unrestricted black-box complexity of jump functions. Evolutionary Computation, 24(4):719-744, 2016 [4].

1 INTRODUCTION

To understand how difficult a problem is for evolutionary algorithms and other black-box optimizers, one proves upper bounds for the problem difficulty by designing and analyzing reasonable algorithms for the problem and lower bounds by studying how fast a theoretically best possible algorithm at most can be. These two approaches, *design and analysis of algorithms* on the one hand and *complexity theory* on the other, complement each other. Their complementary nature has greatly spurred the development of algorithms, both by pointing out areas with room for improvement and by giving concrete hints on how to improve existing methods (see [5] for a recent example).

In this work, we study the second question, that is, how fast in principle a black-box optimization algorithm can solve certain optimization problems. Droste, Jansen, Tinnefeld, and Wegener [13] were the first to ask this question in the context of evolutionary algorithms. In their seminal paper, see also [14], they introduced the notion of *black-box complexity* as a measure of problem difficulty. In simple words, the black-box complexity of an optimization problem is the expected number of function evaluations that are performed by an optimal black-box algorithm until it evaluates an optimum for the first time. As many randomized search heuristics like evolutionary algorithms, ant colony optimization, or simulated annealing are black-box optimizers, the black-box complexity of a problem gives a lower bound on performance of all these search heuristics.

GECCO '17 Companion, Berlin, Germany

978-1-4503-4939-0/17/07…$15.00
DOI: http://dx.doi.org/10.1145/3067695.3084376

While dormant for several years, the area of black-box complexity became very active from 2009 on, possibly incited by the remarkable works [2] and [16]. Since then many deep and surprising results on black-box complexities were found, so that now we reasonably well understand the black-box complexities of classic test functions, e.g., $\Theta(n/\log n)$ for ONEMAX [2, 14] or $\Theta(n \log\log n)$ for LEADINGONES [1], and of several combinatorial optimization problems like sorting, maximum clique and the single-source shortest path problem [14], the minimum spanning tree problem [8], and the partition problem [6]. It was also observed that modified definitions of black-box complexity are able to study the influence of unbiasedness [17, 18], being ranking-based [11], memory size [10], parallelism [3], and elitism [12].

2 OUR RESULTS

In this work, we stay within the realm of classic black-box complexity of pseudo-Boolean functions, that is, we ask how many fitness evaluations an otherwise unrestricted algorithm must perform to find the optimum of a function $f : \{0,1\}^n \to \mathbb{R}$ (given in a black-box fashion) from a given problem class $\mathcal{F}$. With the ONEMAX and LEADINGONES classes as easy unimodal functions being studied, we now turn to JUMP functions. These are test functions of scalable difficulty, because the fitness landscape has a large plateau of low fitness around the optimum. For a jump function with jump size ℓ, this plateau consists of all search points x with Hamming distance $H(x,z)$ from the optimum z between 1 and ℓ. More precisely, for $\ell \in [1..\lfloor \frac{n}{2} \rfloor - 1]$ and $z \in \{0,1\}^n$, we define the jump function $\text{JUMP}_{n,\ell,z}$ by

$$\text{JUMP}_{n,\ell,z}(x) = \begin{cases} n & \text{if } x = z \\ n - H(x,z) & \text{if } \ell < H(x,z) < n - \ell \\ 0 & \text{otherwise} \end{cases}$$

for all $x \in \{0,1\}^n$. The class $\text{JUMP}_{n,\ell}$ then consists of all function $\text{JUMP}_{n,\ell,z}$ with $z \in \{0,1\}^n$.

Our motivation is both understanding the black-box complexity of this well-studied function class and using it as a trigger to develop new methods, in particular, for proving lower bounds for black-box complexities, where at the moment not much is known beyond the information theoretic argument of [14].

Jump functions tend to be difficult for many randomized search heuristics, e.g., the simple (1 + 1) EA needs $\Omega(n^{\ell+1})$ expected time to find the optimum. We refer to [9] for the most recent review of the state of the art. On the complexity theoretic side, surprisingly, the unrestricted black-box complexity of these jump functions has not been regarded so far. In contrast, a detailed investigation exists for the unbiased black-box complexity model [7]. Here, new offspring can only be created from applying unbiased variation operators to up to k previously found search points, where unbiased means invariant under the automorphisms of the hypercube. For unrestricted arity $k = \infty$, the unbiased black-box complexity is $\Theta(n/\log n)$ when $\ell \le (0.5 - \varepsilon)n$, $\varepsilon > 0$ an arbitrary constant. In the case of extreme jump functions for even n, that is $\ell = 0.5n - 1$, the unbiased black-box complexity is $\Theta(n)$. It is clear that these results are valid for the unrestricted black-box complexity as well. Our results, though, will be stronger in that they provide more precise bounds and regard wider ranges of ℓ.

A second black-box complexity result regards the unrestricted black-box complexity, but of a different type of jump functions where (roughly speaking) the global optimum can be hidden on an arbitrary location of the plateau. With this little structure, it seems natural that there is no better algorithm than searching the plateau in a random order, and this is exactly the result proven in [15]. While this last result can be interpreted in the way that the traditional definition of jump functions is less natural, we feel that the long series of previous works still justifies analyzing their black-box complexity.

Our main technical result is a very precise determination of the unrestricted black-box complexity of the jump function class $\textsc{Jump}_{n,\ell}$ for almost all values of ℓ: When the jump parameter ℓ satisfies $\ell < \frac{n}{2} - \sqrt{n}\log_2 n$, the black-box complexity satisfies the upper bound of $(1 + o(1))\frac{2n}{\log_2 n}$, which is also the best known bound for the easy $\textsc{OneMax}$ test function class. Note that $\ell = \frac{n}{2} - \sqrt{n}\log_2 n$ is actually quite large, meaning that all search points with distance between 1 and $\frac{n}{2} - \sqrt{n}\log_2 n - 1$ from the optimum lie on the plateau of low fitness, making this a plateau of size $2^{n(1-o(1))}$ and diameter $\Theta(n)$. For even larger jump sizes $\frac{n}{2} - \sqrt{n}\log_2 n \le \ell < \lfloor n/2 \rfloor - \omega(1)$, we show an upper bound of $(1 + o(1))\frac{n}{\log_2(n-2\ell)}$, where the asymptotic notation refers to $n - 2\ell$ tending to infinity. This upper bound does not make precise the leading constant when $n - 2\ell$ is constant, in particular, not for the extremal case when the jump function has all fitness levels on the plateau except for the "middle level" (for even n) or except for the two middle levels $\lfloor n/2 \rfloor$ and $\lceil n/2 \rceil$ (for odd n). For such *extreme* jump functions (and thus also for all others), we show an upper bound of $n + O(\sqrt{n})$.

These upper bounds are asymptotically of the right order of magnitude. This follows mostly from the information theoretic argument (Theorem 2) in [14]: an optimization problem over a search space S such that each element of S is the unique solution to an instance of the problem has a black-box complexity of at least $\lceil \log_k |S| \rceil - 1$, where k is the maximum number of different fitness levels. For jump functions with jump size ℓ, this gives a lower bound of $\lceil \log_{n+1-2\ell}(2^n) \rceil - 1 = (1 + o(1))n/\log_2(n - 2\ell)$ with the asymptotics being with respect to $n - 2\ell$ tending to infinity. Consequently, for "small" ℓ, say $\ell \le 0.49n$, our upper and lower bounds show the same factor-2 gap that is known from the $\textsc{OneMax}$ problem. This gap shrinks with growing ℓ and for $n/2 - n^{0.5-o(1)} \le \ell \le n/2 - \omega(1)$, our bounds are tight including the leading constant. For constant values of $n-2\ell$, we again see a substantial gap between our upper bounds and the information theoretic lower bound. In the case of an extreme jump function for even n, we have the three different fitness values $0, n/2$, and n, and hence the lower bound is $\log_3(2) \cdot n$. For odd n, we have the four fitness values $0, \lfloor n/2 \rfloor, \lceil n/2 \rceil$, and n, giving a lower bound of $\log_4(2) \cdot n = 0.5n$ only. In both cases, our upper bound is $n + \Theta(\sqrt{n})$ and thus quite far away.

The reason is that the information theoretic argument pretends that at all times, all k answers may occur, and moreover, occur with similar frequency. This is often an overly optimistic view. To overcome this weakness, we significantly extend the information theoretic bound to allow taking care of such reasons for a smaller information gain, that is, to exploit these to prove stronger lower bounds. Our *matrix lower bound theorem* gives improved lower bounds for all black-box complexities of jump functions. In particular, for extreme jump functions, we raise the lower bounds from $\log_3(2) \cdot n$ to $n - 1$ when n is even and from $0.5n$ to $n - 2$ when n is odd. Note that the larger number of fitness values for n odd now has a much smaller influence on the result. With a small extra argument, we improve these bounds to $n + \Omega(\sqrt{n})$, giving a bound sharp up to the $\Theta(\sqrt{n})$ term.

The new lower bound methods triggered by this work, being essentially the first progress on lower bounds for unrestricted black-box complexities after the information-theoretic argument dating back to 2003, are the main methodological contribution of this work. We are optimistic that they can serve to give precise lower bounds also for other black-box complexity problems.

REFERENCES

[1] P. Afshani, M. Agrawal, B. Doerr, C. Doerr, K. G. Larsen, and K. Mehlhorn. The query complexity of finding a hidden permutation. In *Space-Efficient Data Structures, Streams, and Algorithms*, number 8066 in Lecture Notes in Computer Science, pages 1–11. Springer, 2013.

[2] G. Anil and R. P. Wiegand. Black-box search by elimination of fitness functions. In *Proceedings of Foundations of Genetic Algorithms*, pages 67–78, 2009.

[3] G. Badkobeh, P. K. Lehre, and D. Sudholt. Unbiased black-box complexity of parallel search. In *Parallel Problem Solving from Nature XIII*, number 8672 in Lecture Notes in Computer Science, pages 892–901. Springer, 2014.

[4] M. Buzdalov, B. Doerr, and M. Kever. The unrestricted black-box complexity of jump functions. *Evolutionary Computation*, 24(4):719–744, 2016.

[5] B. Doerr, C. Doerr, and F. Ebel. From black-box complexity to designing new genetic algorithms. *Theoretical Computer Science*, 567:87–104, 2015.

[6] B. Doerr, C. Doerr, and T. Kötzing. The unbiased black-box complexity of partition is polynomial. *Artificial Intelligence*, 216:275–286, 2014.

[7] B. Doerr, C. Doerr, and T. Kötzing. Unbiased black-box complexities of jump functions. *Evolutionary Computation*, 2017. To appear. Preliminary version in GECCO 2014.

[8] B. Doerr, T. Kötzing, J. Lengler, and C. Winzen. Black-box complexities of combinatorial problems. *Theoretical Computer Science*, 471:84–106, 2013.

[9] B. Doerr, H. P. Le, R. Makhmara, and T. D. Nguyen. Fast genetic algorithms. In *Proceedings of Genetic and Evolutionary Computation Conference*. ACM, 2017.

[10] B. Doerr and C. Winzen. Playing Mastermind with constant-size memory. *Theory of Computing Systems*, 55(4):658–684, 2014.

[11] B. Doerr and C. Winzen. Ranking-based black-box complexity. *Algorithmica*, 68(3):571–609, 2014.

[12] C. Doerr and J. Lengler. Elitist black-box models: Analyzing the impact of elitist selection on the performance of evolutionary algorithms. In *Proceedings of Genetic and Evolutionary Computation Conference*, pages 839–846. ACM, 2015.

[13] S. Droste, T. Jansen, K. Tinnefeld, and I. Wegener. A new framework for the valuation of algorithms for black-box optimization. In *Proceedings of Foundations of Genetic Algorithms*, pages 253–270. Morgan Kaufmann, 2003.

[14] S. Droste, T. Jansen, and I. Wegener. Upper and lower bounds for randomized search heuristics in black-box optimization. *Theory of Computing Systems*, 39(4):525–544, 2006.

[15] T. Jansen. On the black-box complexity of example functions: The real Jump function. In *Proceedings of Foundations of Genetic Algorithms*, pages 16–24. ACM, 2015.

[16] P. K. Lehre and C. Witt. Black-box search by unbiased variation. In *Proceedings of Genetic and Evolutionary Computation Conference*, pages 1441–1448. ACM, 2010.

[17] P. K. Lehre and C. Witt. Black-box search by unbiased variation. *Algorithmica*, 64:623–642, 2012.

[18] J. Rowe and M. Vose. Unbiased black box search algorithms. In *Proceedings of Genetic and Evolutionary Computation Conference*, pages 2035–2042. ACM, 2011.

2