MVA Reinforcement Learning
Optimization of very difficult functions

Nathan de Lara, Florian Tilquin

January 2, 2016

1 Introduction

1.1 Problem statement

The goal of this paper is to test and compare recently developed algorithms for the optimization of *very difficult functions*. This work is based on the papers by Bubeck [2], Grill [4], Lazaric [1], Bull [3] and Valko [5]. Each one of this paper has a specific definition of *difficult* but the general idea is that the function to optimize has many local maxima and only one global maximum, it has very fast variations and is not necessarily differentiable such that a gradient-based approach to find the optimum should not be successful. All functions are assumed to be bounded and to have a compact support which, up to scaling can be fixed to be $[0,1]$. In the end, the general formulation of the problem is:

$$\text{maximize } f(x) \text{ for } x \in [0,1] \tag{1}$$

1.2 About the multi-armed bandit

1.3 Background

2 Algorithms

In this section, we list the algorithms to be compared and briefly present their respective behaviors.

Reference functions to optimize

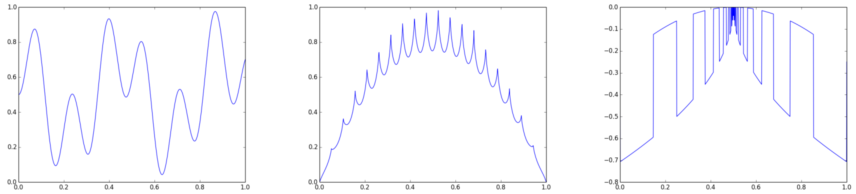

Figure 1: From the left to the right: Two-sine product, Garlang, Grill.

2.1 Hierarchical Optimistic Optimization

2.2 Parallel Optimistic Optimization

2.3 High-Confidence Tree

2.4 Stochastic Simultaneous Optimistic Optimization

2.5 Adaptive-Treed Bandits

3 Results

3.1 Experimental Setup

Objective functions We test the algorithms on different reference functions from [5] and [4]:

1. Two-sine product function: $f_1(x) = \frac{1}{2}(\sin(13x).\sin(27x)) + 0.5$.
2. Garland function: $f_2(x) = 4x(1-x).(\frac{3}{4} + \frac{1}{4}(1 - \sqrt{|\sin(60x)|}))$.
3. Grill function: $f_3(x) = s(\log_2(|x{-}0.5|).(\sqrt{|x-0.5|}{-}(x{-}0.5)^2){-}\sqrt{|x-0.5|}$ where $s(x) = \mathbf{1}(x - \lfloor x \rfloor \in [0, 0.5])$.

The associated plots are displayed in 3.1.

Algorithms setup In order to compare the performances of the different algorithms, we set a desired precision ϵ and a total number of function evaluations T and a number of runs N. Then we compute for each and each algorithm run the best value returned $\widehat{x}^*$ and the cumulative regret. The run is considered a success if $|\widehat{x}^* - x^*| \leq \epsilon$. The success rates are average cumulative regrets are displayed in 3.1.

3.2 Analysis

References

[1] Mohammad Gheshlaghi Azar, Alessandro Lazaric, and Emma Brunskill. Online stochastic optimization under correlated bandit feedback. *arXiv preprint arXiv:1402.0562*, 2014.

Success rates and average cumulative regrets of the algorithms

Algorithm	f_1	f_2	f_3	$\bar{R}_1$	$\bar{R}_2$	$\bar{R}_3$
HOO						
POO						
HCT						
StoSOO						
ATB						

Figure 2: These results are obtained for $\epsilon =$, $T =$ and $N =$.

[2] Sébastien Bubeck, Rémi Munos, Gilles Stoltz, and Csaba Szepesvari. X-armed bandits. *The Journal of Machine Learning Research*, 12:1655–1695, 2011.

[3] Adam D Bull. Adaptive-treed bandits. *arXiv preprint arXiv:1302.2489*, 2013.

[4] Jean-Bastien Grill, Michal Valko, and Rémi Munos. Black-box optimization of noisy functions with unknown smoothness. In *Neural Information Processing Systems*, 2015.

[5] Michal Valko, Alexandra Carpentier, and Rémi Munos. Stochastic simultaneous optimistic optimization. In *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, pages 19–27, 2013.