

Cumulative Regret Analysis of the Piyavskii–Shubert Algorithm and Its Variants for Global Optimization

Kaan Gokcesu^{1,4}, Hakan Gokcesu^{2,3}

¹Regrify AI

²Department of Electrical and Electronics Engineering, Bilkent University, Ankara, Turkey

³Network Technologies Next Generation R&D, Turkcell Technology, Istanbul, Turkey

⁴MIT EECS

gokcesu@regrify.com, hgokcesu@ee.bilkent.edu.tr

Abstract

We study the problem of global optimization, where we analyze the performance of the Piyavskii–Shubert algorithm and its variants. For any given time duration T , instead of the extensively studied simple regret (which is the difference of the losses between the best estimate up to T and the global minimum), we study the cumulative regret up to time T . For L -Lipschitz continuous functions, we show that the cumulative regret is $O(L \log T)$. For H -Lipschitz smooth functions, we show that the cumulative regret is $O(H)$. We analytically extend our results for functions with Hölder continuous derivatives, which cover both the Lipschitz continuous and the Lipschitz smooth functions, individually. We further show that a simpler variant of the Piyavskii–Shubert algorithm performs just as well as the traditional variants for the Lipschitz continuous or the Lipschitz smooth functions. We further extend our results to broader classes of functions, and show that, our algorithm efficiently determines its queries; and achieves nearly minimax optimal (up to log factors) cumulative regret, for general convex or even concave regularity conditions on the extrema of the objective (which encompasses many preceding regularities). We consider further extensions by investigating the performance of the Piyavskii–Shubert variants in the scenarios with unknown regularity, noisy evaluation and multivariate domain.

Introduction

In many applications such as hyper-parameter tuning for learning algorithms and complex system design, the goal is to optimize an unknown function with as few evaluations as possible and use that optimal point in the design (Poor 1994; Cesa-Bianchi and Lugosi 2006). In these types of problems, evaluating the performance of a set of parameters often requires numerical simulations (or cross-validations) with significant computational cost; and the constraints of the application often require a sequential exploration of the solution space with small number of evaluations. Moreover, the function to be optimized may not necessarily have nice properties such as linearity or convexity. This generic problem of sequentially optimizing the output of an unknown and potentially non-convex function is often referred to as the following terminologies: global optimization (Pintér 1991), black-box optimization (Jones, Schonlau, and Welch 1998),

derivative-free optimization (Rios and Sahinidis 2013). In the problem of global optimization, we have a function $f(\cdot)$ that we want to optimize. Let this function $f(\cdot)$ be univariate such that $f(\cdot): \Theta \rightarrow \Omega$, where $\Theta, \Omega \subset \mathbb{R}$ and they are compact intervals. The goal of the global optimization is to find the optimizer of this function $f(\cdot)$ with as little evaluations of the function as possible. Every time a new point x is queried, only its value $f(x)$, i.e., the result of the query, is revealed. Thus, for any query $x \in \Theta$, we observe $f(x) \in \Omega$. Without loss of generality, let $\Theta = [0, 1]$, since optimization on any compact interval Θ can be reduced to optimization on $[0, 1]$ after translating and scaling of the input x .

Related Works

Global optimization has received considerable attention over the past decades. Many algorithms have been proposed in a myriad of fields such as convex optimization (Nesterov 2003; Boyd, Boyd, and Vandenberghe 2004; Bubeck 2015), non-convex optimization (Hansen, Jaumard, and Lu 1992, 1991; Jones, Perttunen, and Stuckman 1993; Jain and Kar 2017; Basso 1982; Shang, Kaufmann, and Valko 2019), stochastic optimization (or approximation) (Spall 2005; Shalev-Shwartz et al. 2012), Bayesian optimization (Brochu, Cora, and De Freitas 2010), bandit optimization over metric spaces (Munos 2014). Additionally, due to numerous applications, it has been heavily utilized in decision theory (Moody and Saffell 2001), control theory (Berenji and Khedkar 1992), game theory (Song, Lewis, and Wei 2016), distribution estimation (Gokcesu and Kozat 2018a; Willems 1996; Gokcesu and Kozat 2017; Shamir and Merhav 1999), anomaly detection (Gokcesu et al. 2019; Delibalta et al. 2016), signal processing (Ozkan et al. 2015), prediction (Singer and Feder 1999; Vanli et al. 2016), multi-armed bandits (Bubeck and Cesa-Bianchi 2012; Gokcesu and Kozat 2018b; Neyshabouri et al. 2018).

There exists various heuristics in literature that deal with this problem such as model-based methods, genetic algorithms and Bayesian optimization. However, the most popular approach is the regularity-based methods since, in many applications, the system has some inherent regularity in its objective, i.e., $f(\cdot)$ satisfies some regularity condition. While there are works with different regularity conditions (e.g., (Bartlett, Gabillon, and Valko 2019), (Grill, Valko, and Munos 2015)) use a notion of smoothness related to hierar-

chical partitioning), the most common is the Lipschitz continuity, where $f(\cdot)$ is Lipschitz continuous with some L .

Lipschitz regularity was first studied in Piyavskii’s work (Piyavskii 1972), which proposes a sequential deterministic method to solve the global optimization problem. The algorithm works by the iterative construction of a function $F(\cdot)$ that lower bounds the function $f(\cdot)$ and the evaluation of $f(\cdot)$ at a point where $F(\cdot)$ reaches its minimum. In the same year, Shubert has independently published the same algorithm (Shubert 1972). Hence, this algorithm has been dubbed the Piyavskii–Shubert algorithm. Basso (Basso 1982), Schoen (Schoen 1982), Shen and Zhu (Shen and Zhu 1987), Horst and Tuy (Horst and Tuy 1987) propose other formulations of the Piyavskii’s algorithm. Sergeyev (Sergeyev 1998) use an alternative smooth auxiliary function as a lower bounding function. Hansen and Jaumard (Hansen and Jaumard 1995) summarize and discuss the algorithms proposed in the literature and present them in a high-level programming language. Brent (Brent 2013) studies another aspect of the Piyavskii’s algorithm, where the function is defined on a compact interval with a bounded second derivative (i.e., Lipschitz smoothness). Jacobsen and Torabi (Jacobsen and Torabi 1978) assume that the function is differentiable and a sum of convex and concave functions. Mayne and Polak (Mayne and Polak 1984) propose an outer approximation when the objective is non-differentiable. Mladineo (Mladineo 1986) investigates the application on multimodal objectives. Breiman and Cutler (Breiman and Cutler 1993) propose an approach that uses the Taylor expansion to build a bounding function $F(\cdot)$. Baritomp and Cutler (Baritomp and Cutler 1994) propose an acceleration of the Breiman and Cutler’s method. Ellaia et al. (Ellaia, Es-Sadek, and Kasbioui 2012) suggest a modified Piyavskii’s algorithm. Horst and Tuy (Horst and Tuy 2013) provides a discussion on deterministic algorithms.

The performance analysis of optimization algorithms are generally done by the convergence to the optimizer. However, the convergence in the input domain becomes challenging when the objective function does not have nice properties like convexity. Hence, in global optimization, the convergence is studied with regards to the functional evaluation of the optimizer (i.e., the optimal value $\min_{x \in [0,1]} f(x)$). At each time t , the learner selects a point x_t to be queried (evaluated). Generally, after each evaluation, the learner selects a point x_t^* , which may differ from the last queried point x_t . The accuracy of the approximation provided by the point x_t^* (returned after the t^{th} evaluation of the function) is measured with its closeness to the optimal function evaluation, i.e., the simple regret at evaluation T is given by the difference of the evaluation of the estimation x_T^* and the optimal point (global minimizer) x^* . Although the Piyavskii–Shubert algorithm was extensively studied in the literature, its regret analysis was limited. The study of the number of iterations by the Piyavskii–Shubert algorithm was initiated by Danilin (Danilin 1971). For its simple regret analysis (which is the difference between the best evaluation so far and the optimal evaluation), a crude regret bound of the form $\tilde{r}_T = O(T^{-1})$ is derived by Mladineo (Mladineo 1986) when the function $f(\cdot)$ is Lipschitz continuous. The authors

further show that the Piyavskii–Shubert algorithm is mini-max optimal and superior to uniform grid search. For univariate functions, a bound on the evaluation complexity was derived by Hansen et al. (Hansen, Jaumard, and Lu 1991) for a variant of the Piyavskii–Shubert algorithm that stops automatically after returning an ϵ -optimal of $f(\cdot)$. They proved that the number of iterations required to reach precision ϵ is at most proportional to $\int_0^1 (f(x) - f(x^*) + \epsilon)^{-1} dx$. For these results, the authors explicitly study the lower bounding functions and improve upon the results of (Danilin 1971). The work of Ellaia et al. (Ellaia, Es-Sadek, and Kasbioui 2012) further improves upon the results of (Danilin 1971; Hansen, Jaumard, and Lu 1991). A work by Malherbe and Vayatis (Malherbe and Vayatis 2017) studied a variant of the Piyavskii–Shubert algorithm called LIPO. Rather than optimizing a single proxy function, LIPO queries the next point randomly in a set of potential optimizers. Their regret bounds depend on stronger assumptions. The work by Bouttier et al. (Bouttier, Cesari, and Gerchinovitz 2020) studies the simple regret of Piyavskii–Shubert under noisy evaluations. Furthermore, (Bouttier, Cesari, and Gerchinovitz 2020) and Bachoch et al. (Bachoc, Cesari, and Gerchinovitz 2021) show that the Piyavskii–Shubert algorithm has instance-optimal simple regret guarantees.

Main Results and Contributions

As in the learning literature, we consider $f(\cdot)$ as a loss function that we want to minimize by choosing estimations x_t from Θ at each time t . The goal is to design an iterative algorithm that creates new estimations using the previously acquired information, i.e., $x_t = \Gamma_t(f(x_1), \dots, f(x_{t-1}), x_1, \dots, x_{t-1})$, where $\Gamma(\cdot)$ is some mapping, and the initial selection x_1 is predetermined. It is not straightforward to optimize any arbitrary $f(\cdot)$. To this end, we define a regularity measure. Instead of the restrictive Lipschitz continuity; we define a weaker, more general regularity condition.

Definition 1. *Let the function $f(\cdot)$ that we want to optimize satisfy the following condition: $|f(x) - f(x_E)| \leq d(|x - x_E|)$, for any local extremum (minimum or maximum) x_E of $f(\cdot)$, where $d(\cdot): [0, 1] \rightarrow \mathbb{R}$ is a known monotone nondecreasing function that satisfies $d(0) = 0$.*

A preliminary contribution of our work is the regularity in Definition 1. There are two key points: • Instead of the traditional Lipschitz continuity with the absolute function $|\cdot|$, we analyze a general regularity $d(\cdot)$, which covers other regularities such as Lipschitz smoothness and Hölder continuity. • The regularity in Definition 1 is only on the extrema of $f(\cdot)$ instead of the whole function, which is especially meaningful when $d(\cdot)$ is convex. If a function $f(\cdot)$ satisfies a convex regularity at every point on its domain, it may only be a constant trivial function. On the other hand, with the extrema regularity of Definition 1, we can consider a multitude of functions, e.g., periodic functions.

Since the objective is treated as a loss function, the goal of an algorithm is to minimize the evaluation $f(x_t)$ observed at time t . Since the functional evaluations can be arbitrarily high, we use the notion of regret to define our algo-

rithmic performance. Let the global minimum of $f(\cdot)$ be $f_* \triangleq \min_{x \in \Theta} f(x)$. Hence, the goal is to make the difference $f(x_t) - f_*$ as small as possible. Instead of the weaker simple regret, for the first time in the literature, we study the stronger cumulative regret for global optimization. Thus, instead of the simple regret $r_T \triangleq \min_{t \in \{1, \dots, T\}} f(x_t) - f_*$, we analyze the cumulative regret up to time T , which is $R_T \triangleq \sum_{t=1}^T f(x_t) - \sum_{t=1}^T f_*$. There exist two key points: • Firstly, cumulative regret is more meaningful when there are operational costs driven by evaluation, i.e., each evaluation corresponds to an action selection that results in a loss or gain (e.g., reinforcement learning). • Secondly, cumulative regret may be more indicative of how well an algorithm performs, since the trivial grid search has a linear cumulative regret in spite of its minimax optimal simple regret (i.e., simple regret is not very distinctive for this problem setting).

For the first time in the literature, we derive cumulative regret bounds for the famous traditional Piyavskii–Shubert algorithm for various polynomial regularities. These bounds have alluded the global optimization field for many years. We were able to achieve them by showing recursive relations for functions that upper-bound the regrets.

Result 1 (Regret of Traditional Piyavskii–Shubert Algorithm). *When the regularity function is $d(\cdot) = K|\cdot|^p$ for $K > 0, p \geq 1$; the Piyavskii–Shubert variants have the cumulative regret $R_T \leq 2K \min(\log(4T), (1 - \gamma)^{-1})$, where $\gamma \triangleq (2^p - 1)^{-1} \leq 1$. Specifically; when $p = 1$, we have $R_T \leq O(K \log(T))$; when $p = 2$, we have $R_T \leq 2K$; for all $p > 2$, we have $R_T \leq 3K$; for any $p > 1$, we have $R_T \leq O(K)$, i.e., independent from T .*

We show that a simpler Piyavskii–Shubert algorithm performs just as well as the traditional variants. It is motivated from the fact that the traditional variants have a lack of control on the iterative optimization regions. Because of its simpler structure, we are able to provide cumulative regret guarantees for a larger variety of regularities, including general convex and concave regularities.

Result 2 (Regret of Simpler Piyavskii–Shubert Algorithm). *When the regularity is $d(\cdot) = K|\cdot|^p$ for $K > 0, p > 0$; the simpler variants have the cumulative regret $R_T \leq K + 2K \frac{2^{(1-p)\log(2T)} - 1}{2^{1-p} - 1}$. Specifically; when $p = 1$: we have $R_T \leq O(K \log T)$; for all $p \geq 2$, we have $R_T \leq 5K$; for any $p > 1$: we have $R_T \leq O(K)$; when $0 < p < 1$: we have $R_T \leq O(KT^{1-p})$.*

We show that all of these regret bounds are minimax optimal up to logarithmic terms. Moreover, we also investigate various extensions in our analyses. When the regularity is unknown, we show that either the regret bound is a piecewise linear function of the input regularity coefficient or the logarithmic regret bound is a piecewise linear function of the logarithm of the input regularity coefficient depending on whether the unknown regularity satisfies some properties such as convexity or concavity. We also show how to deal with noisy evaluations and their effect on the regret bounds. We also show that the univariate optimization can be readily applied to the multivariate case with minimax optimal regret guarantees when the regularity is on the whole function.

Algorithm 1: Global Optimization Algorithmic Framework

- 1: Take the function to be optimized $f(\cdot) : [0, 1] \rightarrow \Re$ and the regularity parameters as input.
 - 2: Initialize candidate set $\mathcal{X} = \emptyset$ and evaluated set $\mathcal{Y} = \emptyset$.
 - 3: Evaluate $f(x)$ at $x = 0$. Evaluate $f(x)$ at $x = 1$. Add to \mathcal{Y} the evaluation pairs $(0, f(0))$ and $(1, f(1))$.
 - 4: If possible, determine a candidate $x \in (0, 1)$ and its score s with the evaluations $\{f(0), f(1)\}$. If s is lower than the minimum evaluation in \mathcal{Y} , add (x, s) to \mathcal{X} . The score s determines which candidate to evaluate, and it represents the minimum possible evaluation in a given interval depending on the regularity parameters.
 - 5: **while** $|\mathcal{X}| \neq \emptyset$ **do**
 - 6: Evaluate x_m with the minimum score s_m in \mathcal{X} . Remove (x_m, s_m) from \mathcal{X} . Add $(x_m, f(x_m))$ to \mathcal{Y} . Let x_m be the only point between the previously evaluated points x_l and x_r (i.e., $x_m \in (x_l, x_r)$).
 - 7: If possible, determine a candidate $x_{lm} \in (x_l, x_m)$ and its score s_{lm} with the respective evaluations $\{f(x_l), f(x_m)\}$. If s_{lm} is lower than the minimum evaluation in \mathcal{Y} , add (x_{lm}, s_{lm}) to \mathcal{X} .
 - 8: If possible, determine a candidate $x_{mr} \in (x_m, x_r)$ and its score s_{mr} with the respective evaluations $\{f(x_m), f(x_r)\}$. If s_{mr} is lower than the minimum evaluation in \mathcal{Y} , add (x_{mr}, s_{mr}) to \mathcal{X} .
 - 9: **end while**
 - 10: Output the point x with minimum evaluation $f(x)$ in \mathcal{Y}
-

Global Optimization Algorithm Framework

In this section^{1, 2}, we provide the algorithmic framework for various Piyavskii–Shubert variants, which can efficiently optimize the objective function $f(\cdot)$ with low regret bounds. Piyavskii–Shubert variants generally work by creating a function $F(\cdot)$, which globally lower bounds the objective function $f(\cdot)$. At each time t , the lower bounding function $F(\cdot)$ is updated (or recreated), hence, we have a time varying function $F_t(\cdot)$. Each $F_t(\cdot)$ is created by using the information of the past evaluations $\{f(x_\tau)\}_{\tau=1}^{t-1}$ and the query points $\{x_\tau\}_{\tau=1}^{t-1}$. Each estimation x_t are selected from the global minima of the lower bounding function $F_t(\cdot)$. Based on this methodology, we provide a general global optimization algorithmic framework in Algorithm 1. There exist two key points in this algorithm: • Firstly, instead of creating the whole lower bounding function $F_t(\cdot)$, our framework only creates the relevant minima and their function evaluations in $F_t(\cdot)$, i.e., the next query x_t and its score s_t . • Secondly, the algorithm updates the lower bounding function by only using the newly evaluated points and their evaluations. The lower bounding functions are created in a piecewise manner using the respective boundary evaluations. We observe

¹We use $\log(\cdot)$ to denote the base-2 logarithm.

²The proofs can be found in the full version.

DOI: 10.48550/arXiv.2108.10859

that the creation of the proxy function itself is inconsequential and the important aspect is determining its extrema. In another perspective, we postulate that it comes down to determining some candidate points and lower bounding scores.

Remark 1. Algorithm 1 is equal to the traditional Piyavskii–Shubert algorithm when the query point is chosen as the intersection of lines with reverse slopes from its respective boundary evaluations.

Remark 2. During runtime, let the number of evaluations so far be T . After each evaluation, at most two potential query points are iteratively created; and the queries not evaluated remain unchanged. The time complexity per evaluation is at most $O(\log T)$. Between any adjacent queries, there will be at most one potential query. Hence, the number of potential queries grows at most linearly with the number of queries. Thus, memory complexity is at most $O(T)$.

Remark 3. The algorithm naturally terminates when there are no queries left in the query list. Additional stopping criteria can be considered, such as stopping after a fixed amount of trials or when a sufficient closeness to the optimizer value is reached, i.e., the minimum evaluation so far is sufficiently close to the minimum score in \mathcal{X} .

Remark 4. We can further increase computational efficiency by eliminating the potential queries with scores higher than the minimum function evaluation queried so far.

Regret Analysis of the Traditional Piyavskii–Shubert Variants

In this section, we investigate the cumulative regret of the traditional Piyavskii–Shubert algorithm and its variants. The candidate point x (in Step 4, 7, 8 of Algorithm 1) is selected as the point with the lowest possible functional value between the nearest boundaries x_l and x_r . Its score s is set as its lowest possible evaluation. Next, we analyze different classes of commonly utilized functions.

Optimizing Lipschitz Continuous Functions

A function $f(\cdot)$ is Lipschitz continuous with L if $|f(x) - f(y)| \leq L|x - y|, \forall x, y \in \Theta$. We can see that for any extremum x_E of $f(\cdot)$, we have $|f(x) - f(x_E)| \leq L|x - x_E|$. Here, the candidate point x is the intersection of the lines that pass through points $(x_l, f_l), (x_r, f_r)$ with slopes $-L, L$, respectively. Using the following (x, s) in Algorithm 1 is the traditional Piyavskii–Shubert algorithm.

Lemma 1. For an L -Lipschitz continuous function $f(\cdot)$, given the boundaries x_l, x_r and evaluations $f_l \triangleq f(x_l), f_r \triangleq f(x_r)$; the candidate $x \in (x_l, x_r)$ with the lowest possible value and its score are

$$\begin{aligned} x &= 0.5(x_l + x_r + (f_l - f_r)/L), \\ s &= 0.5(f_l + f_r - L(x_r - x_l)), \end{aligned}$$

i.e., s is a lower bound on the evaluation of the set (x_l, x_r) .

We next study the cumulative regret of Algorithm 1 when the points and scores are selected with respect to Lemma 1. We start by investigating the regret of a single query.

Lemma 2. For an L -Lipschitz continuous function $f(\cdot)$, let x_m be the next query between the boundaries x_l, x_r and their evaluations $f_l \triangleq f(x_l), f_r \triangleq f(x_r)$. the regret incurred by x_m is

$$f(x_m) - \min_{x \in [0,1]} f(x) \leq 2L \min(x_m - x_l, x_r - x_m).$$

Lemma 2 bounds the single query regret with respect to the minimum distance to the previously evaluated boundaries. Let us observe that the lower bounding algorithm in Algorithm 1 is deterministic. Hence, given L , function $f(\cdot)$ and interval $[0, 1]$ (where 0 and 1 are initially evaluated), it will always evaluate the same points. Let these be x_1^T , i.e.,

$$x_1^T = PS_{LiCo}([0, 1], f(\cdot), L, T),$$

for some function $PS_{LiCo}(\cdot)$, which represents the application of Algorithm 1 to L -Lipschitz continuous functions with the selections in Lemma 1. Let

$$S_L([0, 1], x_1^T) = 2L \sum_{t=1}^T \min_{-1 \leq \tau \leq t-1} |x_t - x_\tau|,$$

be the cumulative bound resulting from Lemma 2 where $x_0 = 0, x_{-1} = 1$ are initially evaluated boundary points of $[0, 1]$. Let the maximum bound $R_L([0, 1], T)$ be equal to

$$\max_{0 \leq l \leq L f(\cdot): l-L.c.} \max S_L([0, 1], PS_{LiCo}([0, 1], f(\cdot), l, T)), \quad (1)$$

where $f(\cdot) : l-L.c.$ refers to the fact that $f(\cdot)$ is l -Lipschitz continuous. This is required for the subsequent analyses.

Lemma 3. We have the following recursive relation for the maximum bound defined in (1)

$$\begin{aligned} R_L([0, 1], T) &\leq 2L \min(x, 1 - x) + xR_L([0, 1], T_1) \\ &\quad + (1 - x)R_L([0, 1], T_2) \end{aligned}$$

for some $x \in [0, 1]$ and $T_1 + T_2 = T - 1$.

Using the recursion in Lemma 3, we reach the following.

Theorem 1. For an L -Lipschitz continuous function $f(\cdot)$, the traditional Piyavskii–Shubert variant with its application in Lemma 1 has the regret $R_T \leq 2L \log(4T)$.

Optimizing Lipschitz Smooth Functions

A differentiable function $f(\cdot)$ is Lipschitz smooth with H if $|f'(x) - f'(y)| \leq H|x - y|$. We can see that for any extremum x_E of $f(\cdot)$, we have $|f(x) - f(x_E)| \leq 0.5H|x - x_E|^2$, since the derivative is zero at any x_E . Here, the candidate point x (if exists) is the minimum point of a quadratic function that passes through points $(x_l, f_l), (x_r, f_r)$ with the second derivative H . (x, s) is given as in the following.

Lemma 4. For a H -Lipschitz smooth function $f(\cdot)$, given the boundaries x_l, x_r and evaluations $f_l \triangleq f(x_l), f_r \triangleq f(x_r)$, the point x with the lowest possible value (score s) is

$$\begin{aligned} x &= \frac{x_l + x_r}{2} + \frac{f_l - f_r}{H(x_r - x_l)}, \\ s &= f_l - \frac{1}{2}H(x - x_l)^2 = f_r - \frac{1}{2}H(x_r - x)^2. \end{aligned}$$

If $x \notin (x_l, x_r)$, there exists no candidate in $[x_l, x_r]$ since $f(x) \geq \min(f(x_l), f(x_r)), \forall x \in [x_l, x_r]$.

We next study the cumulative regret of Algorithm 1 when the points and scores are selected with respect to Lemma 4.

Lemma 5. For an H -Lipschitz smooth function $f(\cdot)$, let x_m be the next query, which is between the boundary points x_l , x_r and their corresponding values $f_l \triangleq f(x_l)$, $f_r \triangleq f(x_r)$. The regret incurred by the evaluation of x_m is

$$f(x_m) - \min_{x \in [0,1]} f(x) \leq H(x_r - x_m)(x_m - x_l).$$

Lemma 5 bounds the single query regret with respect to the multiplication of distances to the boundaries. Similar to Lemma 3, we have a recursive inequality for an upper bound of the regret.

Lemma 6. We have the following recursive relation for an upper bound on the regret

$$R_H([0, 1], T) \leq Hx(1-x) + x^2 R_H([0, 1], T_1) + (1-x)^2 R_H([0, 1], T_2)$$

for some $x \in [0, 1]$ and $T_1 + T_2 = T - 1$.

Using the recursion in Lemma 6, we reach the following constant cumulative regret bound.

Theorem 2. For an H -Lipschitz smooth function $f(\cdot)$, the traditional Piyavskii–Shubert variant with its application in Lemma 4 has the cumulative regret $R_T \leq H$.

Optimizing Hölder Smooth Functions

Here, we aim to optimize a K -Hölder smooth function $f(\cdot)$ with Hölder continuous derivative, i.e., $|f'(x) - f'(y)| \leq Kp|x - y|^{p-1}$, where $0 \leq p - 1 \leq 1$. When $p = 1$, this includes the Lipschitz continuous functions with $K = L$. When $p = 2$, this class is the Lipschitz smooth functions with $2K = H$. We have $|f(x) - f(x_E)| \leq K|x - x_E|^p$, where x_E is any local extremum of $f(\cdot)$, $K > 0$ and $1 \leq p \leq 2$. The candidate x (if exists) is the minimum point of a convex p^{th} -order polynomial that passes points (x_l, f_l) , (x_r, f_r) with the coefficient K .

Lemma 7. For a function $f(\cdot)$ satisfying $|f(x) - f(x_E)| \leq K|x - x_E|^p$, given the boundary points x_l , x_r and their values $f_l \triangleq f(x_l)$ and $f_r \triangleq f(x_r)$, the ‘candidate’ point x between the boundaries x_l and x_r (i.e., $x \in [x_l, x_r]$) with the lowest possible value is the solution of $K|x_r - x|^p - K|x_l - x|^p = f_r - f_l$, and its score is $s = f_l - K|x_l - x|^p = f_r - K|x_r - x|^p$.

If $x \notin (x_l, x_r)$ (where x may not necessarily have a simpler expression), there exists no candidate.

We next study the cumulative regret of Algorithm 1 with the application in Lemma 7.

Lemma 8. For a function $f(\cdot)$ satisfying $|f(x) - f(x_E)| \leq K|x - x_E|^p$, let x_m be the next evaluated candidate point, which is between the boundary points x_l , x_r and their corresponding values $f_l \triangleq f(x_l)$ and $f_r \triangleq f(x_r)$. The regret incurred by the evaluation of x_m is

$$f(x_m) - \min_{x \in [0,1]} f(x) \leq K\bar{x}^p + K(x_r - x_l - \bar{x})^p - K(x_r - x_l - 2\bar{x})^p,$$

where $\bar{x} = \min(x_m - x_l, x_r - x_m)$.

Similar to Lemma 3, we have a recursive inequality for an upper bound of the cumulative regret.

Lemma 9. We have the following recursive relation for an upper bound on the regret.

$$R_K([0, 1], \bar{T}) \leq K(\bar{x}^p + (1 - \bar{x})^p - (1 - 2\bar{x})^p) + x^p R_K([0, 1], T_1) + (1 - x)^p R_K([0, 1], T_2),$$

where $\bar{x} = \min(x, 1 - x)$ for some $x \in [0, 1]$ and $T_1 + T_2 = \bar{T} + 1 = T - 1$.

To derive the regret, we utilize the following useful result.

Proposition 1. For $x \in (0, 0.5]$, $p \geq 1$, we have

$$\frac{1 - (1 - 2x)^p}{1 - (1 - x)^p} \leq 2, \quad \frac{x^p}{1 - (1 - x)^p} \leq \frac{2^{-p}}{1 - 2^{-p}},$$

$$x^p + (1 - x)^p - (1 - 2x)^p \leq 1.$$

Using the recursion in Lemma 9, we reach the following constant cumulative regret bound.

Theorem 3. For a K -Hölder smooth function $f(\cdot)$, the traditional Piyavskii–Shubert variant with its application in Lemma 7 has the cumulative regret $R_T \leq 2K \min(\log(4T), (1 - \gamma)^{-1})$, where $p \geq 1$ and $\gamma \triangleq (2^p - 1)^{-1} \leq 1$.

- When $p = 1$, we have $\gamma = 1$. Choosing $K = L$ gives the regret bound $R_T \leq 2L \log(4T)$, which is the result in Theorem 1 for Lipschitz continuous $f(\cdot)$.
- When $p = 2$, we have $\gamma = 1/3$. $K = 0.5H$ gives, $R_T \leq 1.5H$, which is looser than Theorem 2 because of the similar proof style to Theorem 1.
- We can expand the results to $p \geq 2$. We have $R_T \leq \frac{2^{p+1}-2}{2^p-2}K$, for an arbitrary $p \geq 2$; and $R_T \leq 3K$ for all $p \geq 2$. For any $p > 1$, the cumulative regret is $O(K)$, i.e., independent from T .

As can be seen, higher order regularities result in smaller regret bounds, i.e., easier to learn.

Regret Analysis of the Simpler Piyavskii–Shubert Variants

In this section, as a design choice, the candidate x in the steps of Algorithm 1 is selected differently from the traditional lower-bounding algorithms. As opposed to the point with the lowest possible functional value between the boundaries x_l and x_r (which is the intersection of the lines that pass through points (x_l, f_l) , (x_r, f_r) with slopes $-L$, L respectively for Lipschitz continuous functions); we directly select it as the middle point of x_l and x_r . The exact expression is given in the following.

Definition 2. For an objective function $f(\cdot)$ satisfying Definition 1, given the boundaries x_l , x_r and their evaluations f_l , f_r ; the candidate x between x_l and x_r (i.e., $x \in (x_l, x_r)$) is given by $x = \frac{1}{2}(x_r + x_l)$, which is the middle point of $[x_l, x_r]$ if $|f_r - f_l| \leq d(|x_r - x_l|)$ (where $d(\cdot)$ is as in Definition 1). Otherwise, there does not exist a potential minimizer (from Definition 1) in (x_l, x_r) and we set no such candidate.

Because of the way the potential queries are determined, they can be represented as binary strings which increases memory and communication efficiency. The determination

of the candidate x is much easier than the traditional variants, which may require solving additional optimization problems. The score s in Algorithm 1 is much easier to determine as opposed to the traditional Piyavskii–Shubert variants. Its exact expression is given in the following.

Lemma 10. *For an objective $f(\cdot)$ satisfying Definition 1; given the boundaries x_l, x_r and their evaluations $f_l \triangleq f(x_l), f_r \triangleq f(x_r)$, we assign the potential query $x = \frac{1}{2}(x_r + x_l)$ the score $s = \min(f_l, f_r) - d\left(\frac{x_r - x_l}{2}\right)$, which lower bounds the evaluation of the region $[x_l, x_r]$.*

In our general regret analysis, we again start by bounding the regret of a single queried point.

Lemma 11. *For an objective $f(\cdot)$ satisfying Definition 1 with some known $d(\cdot)$, let x_m be the next query; x_l, x_r are the boundaries and $f_l \triangleq f(x_l), f_r \triangleq f(x_r)$ are their evaluations. The regret incurred by x_m is bounded as $f(x_m) - \min_{x \in [0,1]} f(x) \leq 2d(x_r - x_l)$.*

This result bounds the individual regret of a sampled point x_m with only its boundary values x_l and x_r (irrespective of the functional evaluations f_l and f_r); hence, is a worst case bound. Next, we study the cumulative regret of the algorithm with the candidate points and scores in Definition 2 and Lemma 10, respectively, by deriving the cumulative regret bound up to time T .

Theorem 4. *For a function satisfying Definition 1, using the candidate of Definition 2 and the score of Lemma 10 in Algorithm 1 results in $R_T \leq d(1) + 2 \sum_{i=0}^a 2^i d\left(\frac{1}{2^i}\right)$ and $r_T \leq 2d(2t^{-1})$, where a is an integer such that $2^a + B + 1 = T$ for some integer $1 \leq B \leq 2^a$.*

The cumulative regret is strongly related with the condition $d(\cdot)$. Next, we provide various examples. together with the algorithmic implementation and the regret results.

Corollary 1. *If $f(\cdot)$ is Lipschitz continuous with L , i.e., $|f(x) - f(y)| \leq L|x - y|$, and the algorithm is run using the score selection of Lemma 10 with the regularity $d(|\cdot|) = L|\cdot|$; it has the regrets $R_T \leq L(2 \log T + 3)$ and $r_T \leq 4LT^{-1}$.*

Corollary 2. *If $f(\cdot)$ is differentiable and Lipschitz smooth with H , i.e., $|f'(x) - f'(y)| \leq H|x - y|$, and the algorithm is run using the score selection of Lemma 10 with the regularity $d(|\cdot|) = 0.5H|\cdot|^2$; it has the regrets $R_T \leq 2.5H$, and $r_T \leq 4HT^{-2}$.*

Corollary 3. *If $f(\cdot)$ satisfies $|f(x) - f(x_E)| \leq K|x - x_E|^p$, for any local extremum x_E of $f(\cdot)$ with $K > 0, p > 0$ and the algorithm is run using the score selection of Lemma 10 with the regularity $d(|\cdot|) = K|\cdot|^p$; it has the regrets $R_T \leq K + 2K \frac{2^{(1-p) \log(2T)} - 1}{2^{1-p} - 1}$ and $r_T \leq 2^{1+p}KT^{-p}$, when $p \neq 1$.*

We have the following results for different p :

- When $p = 1$, this class of functions also includes the Lipschitz continuous functions. For $p \rightarrow 1$: we have the result of Corollary 1 with $K = L$. The result follows from substituting the relevant parameters in Corollary 3 and the application of L'Hospital's rule.
- When $p = 2$, this class of functions include the Lipschitz

smooth functions. For $p = 2$: we reach Corollary 2 with $K = H/2$. The result follows directly from substituting the relevant parameters in Corollary 3.

- When $1 \leq p \leq 2$, this class of functions includes the Hölder smooth functions. For $p > 1$: we have $R_T \leq K(1 + 2(1 - 2^{1-p})^{-1})$. The result follows with $T \rightarrow \infty$ in Corollary 3, since $2^{1-p} < 1$. The regret bound is a function of p .

- For all $p > 2$, we have $R_T \leq 5K$. The result follows from bounding p by 2 with $T \rightarrow \infty$ in Corollary 3, since $2^{1-p} < 1$. The regret bound does not depend on p .

Moreover, we have the following new result:

- When $0 < p \leq 1$, this class of functions includes the Hölder continuous functions. For $0 < p < 1$: we have $R_T \leq K \frac{2^{2-p} - 1}{2^{1-p} - 1} T^{1-p}$. The result follows by the direct substitution of $2^{(1-p) \log(2T)}$ with $(2T)^{1-p}$ in Corollary 3. When $0 < p < 1$, the regret bound is polynomial in T .

Corollary 4. *If function $f(\cdot)$ satisfies $|f(x) - f(x_E)| \leq g(|x - x_E|)$, for any local extremum x_E of $f(\cdot)$, where $g(\cdot)$ is nonnegative, convex and $g(0) = 0$; and the algorithm is run using the score selection of Lemma 10 with the regularity $d(|\cdot|) = g(|\cdot|)$; it has the regrets $R_T \leq g(1)(2 \log(T) + 3)$, and $r_T \leq 2g(2T^{-1})$.*

If we have different convex regularities in unknown regions of the domain, we can model the regularity as the maximum of these regularities and achieve at most logarithmic regret since the maximum of convex functions is convex. Moreover, since Corollary 4 holds for any convex regularity, we have $R_T \leq K(1 + 2 \min(\log(2T), (1 - 2^{1-p})^{-1}))$, for any $T, p \geq 1$ after combining with Corollary 3.

Corollary 5. *If function $f(\cdot)$ satisfies $|f(x) - f(x_E)| \leq h(x - x_E)$, for any local extremum x_E of $f(\cdot)$, where $h(\cdot)$ is nonnegative, concave and $h(0) = 0$; and the algorithm is run using the score selection of Lemma 10 with the regularity $d(|\cdot|) = h(|\cdot|)$; it has the regrets $R_T \leq 4Th \left(\frac{2 \log T + 3}{2T - 3} \right)$, and $r_T \leq 2h(2T^{-1})$.*

Theorem 5. *For convex or concave regularities, our regret optimality gap is at most logarithmic.*

Extensions

Implementation on Multivariate Scenarios

We can straightforwardly extend the algorithm for use in multivariate global optimization when the regularity condition is on the function itself and not just its extrema. As an example, we will investigate the Hölder continuous functions, which satisfy the following condition: $|f(x) - f(y)| \leq \|x - y\|_\infty^\alpha$, for any $x, y \in [0, 1]^d$ (where d is the dimension) and $\alpha > 0$. To cast the multivariate optimization problem of $\min_{x \in [0,1]^d} f(x)$ to a univariate optimization problem, we can fill the optimization region $[0, 1]^d$ with hyper-cubes of dimension d and edge length ϵ . For suitable ϵ , the total number of these hyper-cubes are $N = O(\epsilon^{-d})$. Then, we connect the centers of these hyper-cubes with a non-overlapping path. The total length of this curve is $D \leq O(\epsilon^{1-d})$. If we treat this path as our univariate domain, the Hölder condition still holds because of the triangle inequality.

Theorem 6. We have $R_T \leq O(T^{1-\frac{\alpha}{d}})$, for time horizon T , when the hyper-cube length $\epsilon = T^{-\frac{1}{d}}$.

The exact structure of the path has no significance in the worst case analysis. The regret bound is minimax optimal when the space filling path is chosen such that it does not overlap and it has complete coverage over the hyper-cube centers. However, real world performances may differ depending on the path selection, e.g., lattice, spiral etc.

Dealing with Noisy Observations

Suppose each evaluation $f_t = f(x_t)$ has an additive noise v_t , i.e., we observe $\tilde{f}_t = f_t + v_t$. Because of the way the algorithm works, this noise will result in an additive component in the sample regret at every time t . Since the scores of our candidate points are compared against each other, which depend on the evaluations, the redundant regret at time t will be at most $\max_{0 \leq \tau < t} v_\tau - \min_{0 \leq \tau < t} v_\tau$, where $v_0 = 0$. In the cumulative regret bound, we will have an additive component $\sum_{t=1}^T (\max_{\tau < t} v_\tau - \min_{\tau < t} v_\tau)$. Let us assume v_t are independent identically distributed Gaussian noise with an unknown mean μ and unknown variance σ^2 . The expectation of the range of normal random variables over t samples is bounded by $O(\sigma\sqrt{\log t})$. Hence, the summation will be $O(\sigma T\sqrt{\log T})$. Obviously, this is not sublinear. We can circumvent this problem by evaluating each candidate K times and utilize the average of samples as its evaluation.

Theorem 7. Given the noise-free regret $\tilde{O}(T^{1-\alpha})$ for $0 \leq \alpha \leq 1$ and i.i.d. Gaussian sample noise with unknown mean μ and unknown variance σ^2 ; the expected cumulative regret is $\mathbb{E}[R_T] \leq \tilde{O}\left(T^{\frac{1+\alpha}{1+2\alpha}}\right)$, when each query is sampled $T^{\frac{2\alpha}{1+2\alpha}}$ times and their average is utilized as the evaluation.

Similar derivations can be made for different noises.

Regret for Unknown Regularity

For unknown regularity, Algorithm 1 is run with Corollary 1.

Definition 3. Given the boundaries x_l, x_r and their evaluations $f_l \triangleq f(x_l), f_r \triangleq f(x_r)$; the candidate $x = \frac{1}{2}(x_r + x_l)$ has the score $s = \min(f_l, f_r) - L_0 |(x_r - x_l)/2|$, for and input $L_0 > 0$.

We point out that because of the working structure of the algorithm, if all the scores were offset by some ϵ , the sampled points would not change since the candidate scores are compared against each other. Hence, the algorithm would sample the same candidate points with the same evaluations if the score were set as $s = \min(f_l, f_r) - L_0 |(x_r - x_l)/2| - \epsilon$, for any ϵ . In our general regret analysis, we start by bounding the regret of a single queried point.

Proposition 2. For an objective $f(\cdot)$ satisfying Definition 1 with some unknown $d(\cdot)$, and input L_0 ; let x_m be the next sampled query, which is the middle of the boundary points x_l, x_r together with their corresponding evaluations $f_l \triangleq f(x_l)$ and $f_r \triangleq f(x_r)$. The regret incurred by x_m is

$$f(x_m) - \min_{x \in [0,1]} f(x) \leq d(x_r - x_l) + L_0 \left(\frac{x_r - x_l}{2}\right) + \epsilon_0,$$

where $\epsilon_0 = \max_{0 \leq \Delta \leq 1} (d(\Delta) - L_0\Delta)$.

Next, we study the cumulative regret of the algorithm with the points and scores in Lemma 1 and Lemma 10, respectively, by deriving the cumulative regret up to time T .

Theorem 8. For a function $f(\cdot)$ satisfying Definition 1 running Algorithm 1 with the selections in Definition 3 results in $R_T \leq 2d(1) + \sum_{i=0}^a 2^i d\left(\frac{1}{2^i}\right) + L_0 a + \epsilon_0 T, r_T \leq 2d(2t^{-1})$, where a is an integer such that $2^a + B + 1 = T$ for some integer $1 \leq B \leq 2^a$.

This result differs from Theorem 4 by its dependence of L_0 and ϵ_0 .

Theorem 9. If $d(\cdot)$ is convex, we have regret $R_T \leq (3 + \log T)d(1) + L_0 \log T + (d(1) - L_0)^+ T$, which is piecewise linear with L_0 .

Theorem 10. If $d(\cdot)$ is concave, we have $R_T \leq (2T + 1)d\left(\frac{1}{T+1}\right) + L_0 \log T + \epsilon_0 T$, where $\epsilon_0 = \begin{cases} d(\Delta_0) - L_0\Delta_0, & \text{if } \exists \Delta_0 \in [0, 1] \text{ where } d'(\Delta_0) = L_0 \\ (d(1) - L_0)^+, & \text{otherwise} \end{cases}$.

If there exist no $0 \leq \Delta_0 \leq 1$ such that $d'(\Delta_0) = L_0$; the regret dependency on L_0 will be similar as in the convex regularity, where the regret R_T will be the base concave regret plus a piecewise linear term with the same behavior. More interesting case is the first one, where there exists $0 \leq \Delta_0 \leq 1$ such that $d'(\Delta_0) = L_0$. For presentation, let us assume $d(\cdot) = C|\cdot|^p$ for some $C > 0$ and $1 > p > 0$.

Corollary 6. If $d(\cdot) = C|\cdot|^p$ and $p < 1$, we have $R_T \leq 2C(T+1)^{1-p} + L_0 \log T + L_0^{-\frac{1-p}{1-p}} C^{\frac{1}{1-p}} T$. The logarithmic regret bound $\log(R_T)$ is piecewise linear with logarithm of the input, i.e., $\log(L_0)$.

Limitations

The limitations in our approach are: (1) the regret results are minimax optimal when the regularity condition is known, (2) for noisy evaluations, the noise is assumed to be additive and independent, (3) for minimax optimal regret in multivariate scenarios, the regularity is on the whole function in addition to its extrema, (4) for unknown regularities, sub-linear regret is achieved when the regularity prediction is sufficiently close to the true regularity.

Conclusion

We studied the problem of univariate global optimization and investigated the cumulative regret of the Piyavskii–Shubert algorithm and its variants. Instead of the traditional Lipschitz regularity, we consider the weaker extrema-specific regularity, which allows for a much larger class of functions in addition to the Lipschitz continuous, Lipschitz smooth and Hölder continuous functions. We showed that the Piyavskii–Shubert variants have minimax optimal cumulative regrets. We also showed that simpler Piyavskii–Shubert variants with predetermined queries perform just as well as the traditional variants. We extended our analyses to general convex and concave regularities. We considered further extensions by investigating the performance of the Piyavskii–Shubert variants in the scenarios with unknown regularity, noisy evaluation and multivariate domain.

References

- Bachoc, F.; Cesari, T.; and Gerchinovitz, S. 2021. Instance-Dependent Bounds for Zeroth-order Lipschitz Optimization with Error Certificates. *Advances in Neural Information Processing Systems*, 34: 24180–24192.
- Baritomba, W.; and Cutler, A. 1994. Accelerations for global optimization covering methods using second derivatives. *Journal of Global Optimization*, 4(3): 329–341.
- Bartlett, P. L.; Gabillon, V.; and Valko, M. 2019. A simple parameter-free and adaptive approach to optimization under a minimal local smoothness assumption. In *Algorithmic Learning Theory*, 184–206. PMLR.
- Basso, P. 1982. Iterative methods for the localization of the global maximum. *SIAM Journal on Numerical Analysis*, 19(4): 781–792.
- Berenji, H. R.; and Khedkar, P. 1992. Learning and tuning fuzzy logic controllers through reinforcements. *IEEE Transactions on Neural Networks*, 3(5): 724–740.
- Bouttier, C.; Cesari, T.; and Gerchinovitz, S. 2020. Regret analysis of the Piyavskii-Shubert algorithm for global Lipschitz optimization. *arXiv preprint arXiv:2002.02390*.
- Boyd, S.; Boyd, S. P.; and Vandenberghe, L. 2004. *Convex optimization*. Cambridge university press.
- Breiman, L.; and Cutler, A. 1993. A deterministic algorithm for global optimization. *Mathematical Programming*, 58(1): 179–199.
- Brent, R. P. 2013. *Algorithms for minimization without derivatives*. Courier Corporation.
- Brochu, E.; Cora, V. M.; and De Freitas, N. 2010. A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. *arXiv preprint arXiv:1012.2599*.
- Bubeck, S. 2015. Convex Optimization: Algorithms and Complexity. *Foundations and Trends® in Machine Learning*, 8(3-4): 231–357.
- Bubeck, S.; and Cesa-Bianchi, N. 2012. Regret Analysis of Stochastic and Nonstochastic Multi-armed Bandit Problems. *Foundations and Trends in Machine Learning*, 5(1): 1–122.
- Cesa-Bianchi, N.; and Lugosi, G. 2006. *Prediction, learning, and games*. Cambridge university press.
- Danilin, Y. M. 1971. Estimation of the efficiency of an absolute-minimum-finding algorithm. *USSR Computational Mathematics and Mathematical Physics*, 11(4): 261–267.
- Delibalta, I.; Gokcesu, K.; Simsek, M.; Baruh, L.; and Kozat, S. S. 2016. Online Anomaly Detection With Nested Trees. *IEEE Signal Process. Lett.*, 23(12): 1867–1871.
- Ellaia, R.; Es-Sadek, M. Z.; and Kasbioui, H. 2012. Modified Piyavskii’s Global One-Dimensional Optimization of a Differentiable Function. *Applied Mathematics*, 3: 1306–1320.
- Gokcesu, K.; and Kozat, S. S. 2017. Online anomaly detection with minimax optimal density estimation in nonstationary environments. *IEEE Transactions on Signal Processing*, 66(5): 1213–1227.
- Gokcesu, K.; and Kozat, S. S. 2018a. Online Density Estimation of Nonstationary Sources Using Exponential Family of Distributions. *IEEE Trans. Neural Networks Learn. Syst.*, 29(9): 4473–4478.
- Gokcesu, K.; and Kozat, S. S. 2018b. An Online Minimax Optimal Algorithm for Adversarial Multiarmed Bandit Problem. *IEEE Transactions on Neural Networks and Learning Systems*, 29(11): 5565–5580.
- Gokcesu, K.; Neyshabouri, M. M.; Gokcesu, H.; and Kozat, S. S. 2019. Sequential Outlier Detection Based on Incremental Decision Trees. *IEEE Trans. Signal Process.*, 67(4): 993–1005.
- Grill, J.-B.; Valko, M.; and Munos, R. 2015. Black-box optimization of noisy functions with unknown smoothness. *Advances in Neural Information Processing Systems*, 28: 667–675.
- Hansen, P.; and Jaumard, B. 1995. Lipschitz optimization. In *Handbook of global optimization*, 407–493. Springer.
- Hansen, P.; Jaumard, B.; and Lu, S.-H. 1991. On the number of iterations of Piyavskii’s global optimization algorithm. *Mathematics of Operations Research*, 16(2): 334–350.
- Hansen, P.; Jaumard, B.; and Lu, S.-H. 1992. Global optimization of univariate Lipschitz functions: I. Survey and properties. *Mathematical programming*, 55(1): 251–272.
- Horst, R.; and Tuy, H. 1987. On the convergence of global methods in multiextremal optimization. *Journal of Optimization Theory and Applications*, 54(2): 253–271.
- Horst, R.; and Tuy, H. 2013. *Global optimization: Deterministic approaches*. Springer Science & Business Media.
- Jacobsen, S. E.; and Torabi, M. 1978. A global minimization algorithm for a class of one-dimensional functions. *Journal of Mathematical Analysis and Applications*, 62(2): 310–324.
- Jain, P.; and Kar, P. 2017. Non-convex Optimization for Machine Learning. *Foundations and Trends® in Machine Learning*, 10(3-4): 142–336.
- Jones, D. R.; Perttunen, C. D.; and Stuckman, B. E. 1993. Lipschitzian optimization without the Lipschitz constant. *Journal of optimization Theory and Applications*, 79(1): 157–181.
- Jones, D. R.; Schonlau, M.; and Welch, W. J. 1998. Efficient global optimization of expensive black-box functions. *Journal of Global optimization*, 13(4): 455–492.
- Malherbe, C.; and Vayatis, N. 2017. Global optimization of lipschitz functions. In *International Conference on Machine Learning*, 2314–2323. PMLR.
- Mayne, D. Q.; and Polak, E. 1984. Outer approximation algorithm for nondifferentiable optimization problems. *Journal of Optimization Theory and Applications*, 42(1): 19–30.
- Mladineo, R. H. 1986. An algorithm for finding the global maximum of a multimodal, multivariate function. *Mathematical Programming*, 34(2): 188–200.
- Moody, J.; and Saffell, M. 2001. Learning to trade via direct reinforcement. *IEEE Transactions on Neural Networks*, 12(4): 875–889.

- Munos, R. 2014. From Bandits to Monte-Carlo Tree Search: The Optimistic Principle Applied to Optimization and Planning. *Foundations and Trends® in Machine Learning*, 7(1): 1–129.
- Nesterov, Y. 2003. *Introductory lectures on convex optimization: A basic course*, volume 87. Springer Science & Business Media.
- Neyshabouri, M. M.; Gokcesu, K.; Gokcesu, H.; Ozkan, H.; and Kozat, S. S. 2018. Asymptotically optimal contextual bandit algorithm using hierarchical structures. *IEEE transactions on neural networks and learning systems*, 30(3): 923–937.
- Ozkan, H.; Donmez, M. A.; Tunc, S.; and Kozat, S. S. 2015. A Deterministic Analysis of an Online Convex Mixture of Experts Algorithm. *IEEE Transactions on Neural Networks and Learning Systems*, 26(7): 1575–1580.
- Pintér, J. D. 1991. Global optimization in action. *Scientific American*, 264: 54–63.
- Piyavskii, S. 1972. An algorithm for finding the absolute extremum of a function. *USSR Computational Mathematics and Mathematical Physics*, 12(4): 57–67.
- Poor, H. V. 1994. *An Introduction to Signal Detection and Estimation*. NJ: Springer.
- Rios, L. M.; and Sahinidis, N. V. 2013. Derivative-free optimization: a review of algorithms and comparison of software implementations. *Journal of Global Optimization*, 56(3): 1247–1293.
- Schoen, F. 1982. On a sequential search strategy in global optimization problems. *Calcolo*, 19(3): 321–334.
- Sergeyev, Y. D. 1998. Global one-dimensional optimization using smooth auxiliary functions. *Mathematical Programming*, 81(1): 127–146.
- Shalev-Shwartz, S.; et al. 2012. Online Learning and Online Convex Optimization. *Foundations and Trends® in Machine Learning*, 4(2): 107–194.
- Shamir, G. I.; and Merhav, N. 1999. Low-complexity sequential lossless coding for piecewise-stationary memoryless sources. *IEEE Transactions on Information Theory*, 45(5): 1498–1519.
- Shang, X.; Kaufmann, E.; and Valko, M. 2019. General parallel optimization a without metric. In *Algorithmic Learning Theory*, 762–788. PMLR.
- Shen, Z.; and Zhu, Y. 1987. An interval version of Shubert’s iterative method for the localization of the global maximum. *Computing*, 38(3): 275–280.
- Shubert, B. O. 1972. A sequential method seeking the global maximum of a function. *SIAM Journal on Numerical Analysis*, 9(3): 379–388.
- Singer, A. C.; and Feder, M. 1999. Universal linear prediction by model order weighting. *IEEE Transactions on Signal Processing*, 47(10): 2685–2699.
- Song, R.; Lewis, F. L.; and Wei, Q. 2016. Off-Policy Integral Reinforcement Learning Method to Solve Nonlinear Continuous-Time Multiplayer Nonzero-Sum Games. *IEEE Transactions on Neural Networks and Learning Systems*, PP(99): 1–10.
- Spall, J. C. 2005. *Introduction to stochastic search and optimization: estimation, simulation, and control*, volume 65. John Wiley & Sons.
- Vanli, N. D.; Gokcesu, K.; Sayin, M. O.; Yildiz, H.; and Kozat, S. S. 2016. Sequential Prediction Over Hierarchical Structures. *IEEE Transactions on Signal Processing*, 64(23): 6284–6298.
- Willems, F. M. J. 1996. Coding for a binary independent piecewise-identically-distributed source. *IEEE Transactions on Information Theory*, 42(6): 2210–2217.