Optimistic Optimization of a Brownian

Jean-Bastien Grill  
SequeL team, INRIA Lille - Nord Europe, France  
jean-bastien.grill@inria.fr

Michal Valko  
SequeL team, INRIA Lille - Nord Europe, France  
michal.valko@inria.fr

Rémi Munos  
DeepMind Paris*  
munos@google.com

Abstract

In this paper, we address the problem of optimizing a Brownian motion. More precisely, we consider a (random) realization $W$ of a Brownian motion on $[0, 1]$. Now, given this function, our goal is to return an $\varepsilon$-approximation of its maximum using the smallest possible number of function evaluations. This number is called sample complexity of the algorithm. We provide an algorithm with sample complexity of order $\log^2(1/\varepsilon)$. This improves over previous results of Al-Mharmah and Calvin [1996] and Calvin et al. [2017] which provided polynomial rates only. Our algorithm is adaptive — each query depends on previous values — and can be seen as an instance of the optimism-in-the-face-of-uncertainty principle.

1 Introduction

We are interested in optimizing a sample of a standard Brownian motion on $[0, 1]$, denoted by $W$. More precisely, we want to sequentially select query points $t_n \in [0, 1]$, observe $W(t_n)$, and decide when to stop and return a point $\hat{t}$ and its value $\hat{M} = W(\hat{t})$ in order to well approximate its maximum $M = \sup_{t \in [0, 1]} W(t)$. The evaluations $t_n$ can be chosen adaptively as a function of previously observed values $W(t_1), ..., W(t_{n-1})$. Our goal is, given an $\varepsilon > 0$, to stop evaluating the function as early as possible while still being able to return $\hat{t}$ such that with probability at least $1 - \varepsilon$, $M - W(\hat{t}) \leq \varepsilon$. The expected number of function evaluations required by the algorithm to achieve this $\varepsilon$-approximation of the maximum defines the sample-complexity.

Motivation

There are two types of situations where this problem is useful. The first type is when the random sample function $W$ (drawn from the random process) already exists prior to the optimization. Either it has been generated before the optimization starts and queries correspond to reading values of the function already stored somewhere. For example, financial stocks are stored at some high temporal resolution and we want to retrieve the maximum of the stock using the smallest number of queries to the memory. Alternatively, the process physically exists and queries correspond to measuring it.

Another situation is when the function does not exist prior to the optimization but is built simultaneously as it is optimized. In other words, observing the function actually creates it. An application of this is when we want to return a sample of the maximum (and point of maximum) of a Brownian motion conditioned on a set of already observed values. For example, in Bayesian optimization for Gaussian processes, a technique called Thomson sampling [Thompson, 1933, Chapelle and Li, 2011, Agrawal and Goyal, 2012, Kaufmann et al., 2012, Russo et al., 2017, Basu and Ghosh, 2017] requires returning the maximum of a sampled function drawn from the posterior distribution. The problem considered in the present paper can be seen as a way to approximately perform this step in a computationally efficient way when the Gaussian process is a Brownian motion.

*on the leave from SequeL team, INRIA Lille - Nord Europe, France

32nd Conference on Neural Information Processing Systems (NIPS 2018), Montréal, Canada.
Moreover, even though our algorithm comes from the ideas of learning theory, it has some application beyond it. In order to computationally sample a solution to a stochastic differential equation, Hefter and Herzwurm [2017] express its solution as a function of the Brownian \( W \) and its running minimum. They then need, as a subroutine, an algorithm for the optimization of Brownian motion to compute its running minimum.

**Prior work** Al-Mharmah and Calvin [1996] provide a non-adaptive method to optimize a Brownian motion. They prove that their method is optimal among all non-adaptive methods and their sample complexity is polynomial of order \( 1/\sqrt{\varepsilon} \). More recently, Calvin et al. [2017] provide an adaptive algorithm with a sample complexity lower than any polynomial rate showing that adaptability to previous samples yields a significant algorithmic improvement. Yet their result does not guarantee better rate than a polynomially one.

**Our contribution** We introduce the algorithm \( \text{OOB} \) (optimistic optimization of the Brownian motion). It uses the optimism-in-face-of-uncertainty principle. Given \( n - 1 \) points already evaluated, we define a set of functions \( \mathcal{U}_n \) in which \( W \) lies with high probability. We then select the next query point \( t_n \) where the maximum of the most optimistic function of \( \mathcal{U}_n \) is reached:

\[
t_n = \arg \max_{t \in [0,1]} \max_{f \in \mathcal{U}_n} f(t)\,.
\]

Our result is a simple algorithm that requires an expected number of queries of the order of \( \log^2(1/\varepsilon) \) to return an \( \varepsilon \)-approximation of the maximum, with probability at least \( 1 - \varepsilon \) w.r.t. the random sample of the Brownian motion. This makes our sample complexity better than any polynomial rate.

Munos [2011] provides sample complexity results for optimizing a function \( f \) characterized by two coefficients \((d, C)\) where \( d \) is the near-optimality dimension and \( C \) the corresponding constant (see their Definition 3.1). It is defined as the smallest \( d \geq 0 \) such that there exists a semi-metric \( \ell \) and a constant \( C > 0 \), such that, for all \( \varepsilon > 0 \), the maximal number of disjoint \( \ell \)-balls of radius \( O(\varepsilon) \) with center in \( \{x, f(x) \geq \sup_x f(x) - \varepsilon\} \) is less than \( C \varepsilon^{-d} \). Under the assumption that \( f \) is locally (around the global maximum) one-sided Lipschitz with respect to \( \ell \) (see their Assumption 2), they proved that for a function \( f \) characterized by \((d = 0, C)\), their \( \text{DOO} \) algorithm has a sample complexity of \( O(C \log(1/\varepsilon)) \), whereas for a function characterized by \((d > 0, C)\), the sample complexity of \( \text{DOO} \) is \( O(C/\varepsilon^d) \).

Our result answers an open question raised in their work: What is the near-optimality dimension of a Brownian-motion? The Brownian motion being a stochastic process, this quantity is a random variable so we consider the number of disjoint balls in expectation. We show that for any \( \varepsilon \), there exists some particular metric \( \ell_\varepsilon \) such that the Brownian motion \( W \) is \( \ell_\varepsilon \)-Lipschitz with probability \( 1 - \varepsilon \), and there exists a constant \( C(\varepsilon) = O(\log(1/\varepsilon)) \) such that \((d = 0, C(\varepsilon))\) characterizes the Brownian motion. However, there exists no constant \( C < \infty \) independent of \( \varepsilon \) such that \((d = 0, C)\) characterizes the Brownian motion. This is compatible with our result that our algorithm has a sample complexity of \( O(\log^2(1/\varepsilon)) \).

## 2 Algorithm for Brownian optimization

The algorithm \( \text{OOB} \) is a version of \( \text{DOO} \) [Munos, 2011] with a modified upper bound on the function, in order to be able to optimize stochastic processes. Consider the points \( t_1 < t_2 < \ldots < t_n \) evaluated so far and \( t_0 = 0 \). The algorithm defines an upper confidence bound \( B_{[t_i,t_{i+1}]} \) for each interval \([t_i, t_{i+1}]\) with \( i \in \{0, \ldots, n - 1\} \) and samples \( W \) in the middle of the interval with the highest confidence bound.

More formally, let \( \varepsilon \) be the required precision, the only given argument of the algorithm. For any \( 0 \leq a \leq b \leq 1 \), the interval \([a, b]\) is associated with an upper bound \( B_{[a,b]} \) defined by

\[
B_{[a,b]} \triangleq \max(W(a), W(b)) + \eta_{\varepsilon}(b - a),
\]

where

\[
\forall \delta > 0 \text{ s.t. } \varepsilon \delta \leq \frac{1}{2}, \quad \eta_{\varepsilon}(\delta) \triangleq \sqrt{\frac{5\delta}{2} \ln \left(\frac{2}{\varepsilon \delta}\right)}.
\]

\( \text{OOB} \) operates as follows. It keeps track of a set \( \mathcal{I} \) of intervals \([a, b]\) with \( W \) already being sampled at \( a \) and \( b \). The algorithm first samples \( W(1) \) (\( W(1) \sim \mathcal{N}(0,1) \)) in order to initialize the set \( \mathcal{I} \) to
the singleton \( \{0, 1\} \). Then, the algorithm splits the interval \( I \in \mathcal{I} \) associated with the highest upper bound \( B_I \). We give the detailed procedure below as Algorithm 1.

Algorithm 1 OOB algorithm

Input: \( \varepsilon \)
Init: \( \mathcal{I} \leftarrow \emptyset \), \( i \leftarrow 0 \)
loop
\[ [a, b] \in \arg \max_{t \in \mathcal{I}} B_t \{ \text{break ties arbitrarily} \} \]
if \( \eta_{\varepsilon}(b - a) \leq \varepsilon \) then
   break
end if
\[ t_i \leftarrow W \left( \frac{a + b}{2} \right) \]
\( \mathcal{I} \leftarrow \mathcal{I} \cup [a, \frac{a + b}{2}] \cup [\frac{a + b}{2}, b] \} \backslash \{a, b\} \)
end loop
Output: location \( \hat{t}_c \leftarrow \arg \max_i W(t_i) \) and its value \( W(\hat{t}_c) \leftarrow \max_i W(t_i) \)

3 Main result

Let \( M = \sup_{t \in [0, 1]} W(t) \) be the maximum of the Brownian motion, \( \hat{t}_c \) the output of OOB called with parameter \( \varepsilon > 0 \), and \( N_c \) the number of Brownian evaluations performed until OOB terminates. All are random variables that depend on the Brownian motion \( W \). We now state our main result and prove it in the subsequent section.

Theorem 1. There exists a constant \( c > 0 \) such that for all \( \varepsilon < 1/2 \),
\[ \mathbb{P} \left[ M - W(\hat{t}_c) > \varepsilon \right] \leq \varepsilon \quad \text{and} \quad \mathbb{E}[N_c] \leq c \log^2 \left( \frac{1}{\varepsilon} \right). \]

The first inequality quantifies the correctness of our estimator \( \hat{M}_c = W(\hat{t}_c) \). Given a realization of the Brownian motion, our OOB is deterministic. The only source of randomness comes from the realization of the Brownian. Therefore, correctness means that among all possible realizations of the Brownian motion, there is a subset of measure at least \( 1 - \varepsilon \) on which OOB will output an estimate \( \hat{M}_c \) which is at most \( \varepsilon \)-away from the true maximum. We call this a PAC, probably approximately correct, guarantee. The second inequality quantifies performance. The result says that the expectation (over \( W \)) of the number of samples that OOB requires to optimize this realization with precision \( \varepsilon \) is of order \( O \left( \log^2 \left( \frac{1}{\varepsilon} \right) \right) \).

Remark 1. We can get the classical \( (\delta, \varepsilon) \)-PAC guarantee easily as a corollary as follows. For any \( \delta > 0 \) and \( \varepsilon > 0 \), we can choose \( \varepsilon' = \min(\delta, \varepsilon) \) and apply Theorem 1 for \( \varepsilon' \) from which we get \( \mathbb{P} \left[ M - W(\hat{t}_c) > \varepsilon' \right] \leq \varepsilon' \) which is stronger than \( \mathbb{P} \left[ M - W(\hat{t}_c) > \varepsilon \right] \leq \delta \). Similarly, we also get \( \mathbb{E}[N_c] \leq c \log^2 \left( \frac{1}{\varepsilon'} \right) \leq 4c \left( \log \left( \frac{1}{\varepsilon} \right) + \log \left( \frac{1}{\delta} \right) \right)^2 \).

Remark 2. Our PAC guarantee is actually stronger than stated in Theorem 1. Indeed, the PAC guarantee analysis can be done conditioned on the collected function evaluations and get
\[ \mathbb{P} \left[ M - W(\hat{t}_c) > \varepsilon | W(t_1), ..., W(t_{N_c}) \right] \leq \varepsilon, \]
from which taking the expectation on both sides gives back the first part of Theorem 1. This means that the unfavorable cases (i.e., the Brownian realizations for which \( M - \hat{M}_c > \varepsilon \)) are not concentrated on some subsets of Brownian realizations matching some evaluations in \( t_1, ..., t_{N_c} \). In other words, the PAC guarantee also holds when restricted to the Brownian realizations matching the evaluations in \( t_1, ..., t_{N_c} \) only. This is made possible because \( N_c \) is not fixed but depends on the function evaluations done by OOB.

One difference with the result of Calvin et al. [2017] is that theirs is with respect to the \( L_p \) norm. For their algorithm, they prove that with \( n \) samples it returns \( \hat{t}_n \in [0, 1] \) such that
\[ \forall r > 1, p > 1, \quad \exists c_{r,p}, \quad \mathbb{E} \left[ \left| M - W(\hat{t}_n) \right|^p \right]^{1/p} \leq c_{r,p}/n^r. \]
To express their result in the same formalism as ours, we first choose to achieve accuracy $\varepsilon^2$ and compute the number of samples $n_{\varepsilon^2}$ needed to achieve it. Then, for $p = 1$, we apply Markov inequality and get that for all $r > 1$ there exists $c_{r,1}$ such that

$$
\mathbb{P} \left[ M - W(\hat{\ell}_{n_{\varepsilon^2}}) > \varepsilon \right] \leq \varepsilon \quad \text{and} \quad N_{\varepsilon} \leq c_{r,1}/\varepsilon^{1/r}.
$$

On the other hand, in our Theorem 1 we give a poly-logarithmic bound for the sample complexity which is better than any polynomial rate.

4 Analysis

We provide a proof of the main result. The full proofs the auxiliary lemmas are in the appendix. We let $\mathcal{I}_t$ be the set $\mathcal{I}$ kept by the algorithm after $t$ function evaluations. We define an event $C$ such that for any interval $I$ of the form $I = [k/2^h, (k+1)/2^h]$ with $k$ and $h$ being two integers where $0 \leq k < 2^h$, the process $W$ is lower than $B_I$ on the interval $I$.

**Definition 1.** The event $C$ is defined as

$$
C \triangleq \bigcap_{h=0}^{\infty} \bigcap_{k=0}^{2^h-1} \left\{ \sup_{t \in [k/2^h, (k+1)/2^h]} W(t) \leq B_{[k/2^h, (k+1)/2^h]} \right\}.
$$

The event $C$ can be seen as a proxy for the Lipschitz condition on $W$ for the pseudo-distance $d(x, y) = \sqrt{|y - x| \ln (1/|y - x|)}$ because $\eta_{c}(\delta)$ scales with $\delta$ as $\sqrt{\delta \ln (1/\delta)}$. It holds with high probability as stated in the following lemma. By definition, the Brownian bridge is the process $Br(t) \triangleq (W(t)|W(1) = 0)$. Then, Lemma 1 follows from the Markov property of the Brownian combined with a bound on the law of the maximum of the Brownian bridge to bound the probability $\mathbb{P} \left[ \sup_{t \in \mathcal{I}} W(t) \geq B_I \right]$ for any $I$ of the form $[k/2^h, (k+1)/2^h]$ and a union bound over all these intervals.

**Lemma 1.** The event $C$ from Definition 1 holds with high probability,

$$
\mathbb{P} \left[ C^c \right] \leq \varepsilon^5.
$$

Lemma 1 is useful for two reasons. As we bound the sample complexity on event $C$ and the complementary event in two different ways, we can use Lemma 1 to combine the two bounds to prove Proposition 2 in the end. We also use a weak version of it, bounding $\varepsilon^5$ by $\varepsilon$ to prove our PAC guarantee. For this purpose, we combine the definition of $C$ with the termination condition of 008 to get that, under event $C$, the best point evaluated so far by the algorithm, $\hat{M}_{\varepsilon}$, is close to the actual maximum $M$ of the Brownian up to $\varepsilon$. Now, since $C$ holds with high probability, we have the following PAC guarantee which is the first part of the main theorem.

**Proposition 1.** The estimator $\hat{M}_{\varepsilon} = W(\hat{\ell}_{\varepsilon})$ has a probably approximately correct guarantee,

$$
\mathbb{P} \left[ M - \hat{M}_{\varepsilon} > \varepsilon \right] \leq \varepsilon.
$$

In fact, Proposition 1 is the easy-to-obtain part of the main theorem. We are now left to prove that the number of samples needed before achieving such PAC guarantee is low. As the next step, we define the near-optimality property. A point $t$ is said to be $\eta$-near-optimal when its value $W(t)$ is close to the maximum $M$ of the Brownian up to $\eta$. We now state a precise definition.

**Definition 2.** When a $(h, k, \eta)$ verifies $W \left( \left( \frac{k}{2^h} \right) \right) \geq M - \eta$, we say that the point $t = (k/2^h)$ is $\eta$-near-optimal. We define $N_{h}(\eta)$ as the number of $\eta$-near-optimal points among $\{0/2^h, 1/2^h, \ldots, 2^h/2^h\}$,

$$
N_{h}(\eta) \triangleq \text{Card} \left\{ k \in 0, \ldots, 2^h, \text{ such that } W \left( \frac{k}{2^h} \right) \geq M - \eta \right\}.
$$

Notice that $N_{h}(\eta)$ is a random variable which depends on a particular realization of the Brownian. The reason why we are interested in the number of near-optimal points is that points the algorithm
will sample are \( \eta_\varepsilon (1/2^h) \)-near-optimal. Since we use the principle of optimism in face of uncertainty, we consider the upper bound of the Brownian and sample where this upper bound is the largest. If our bounds on \( W \) hold (i.e., under event \( C \)), then any interval \( I \) with optimistic bound \( B_I < M \) is never split by the algorithm. This is true when \( C \) holds because if the maximum of \( W \) is reached in \( I^* \), then \( B_{I^*} \geq M > B_I \) which shows that \( I^* \) is always chosen over \( I \). Therefore, a necessary condition for an interval \([a, b]\) to be split is that \( \max (W(a), W(b)) \geq M - \eta \) which means that either \( a \) or \( b \) are \( \eta \)-near-optimal which is the key point of Lemma 2.

**Lemma 2.** Under event \( C \), the number of evaluated points \( N_\varepsilon \) by the algorithm verifies

\[
N_\varepsilon \leq 2 \sum_{h=0}^{h_{\text{max}}} N_h (\eta_\varepsilon (1/2^h)),
\]

with \( h_{\text{max}} \) being the smallest \( h \) such that \( \eta_\varepsilon (1/2^h) \leq \varepsilon \).

Lemma 2 explicitly links the near-optimality dimension that we just defined with the number of samples \( N_\varepsilon \) performed by \( \text{OOB} \) before it terminates. Here, we used the optimism-in-face-of-uncertainty principle which can be applied to any function. In particular, we define a high-probability event \( C \) under which the number of samples is bounded by the number of near-optimal points \( N_h (\eta_h) \) for all \( h \leq h_{\text{max}} \).

We now prove a property specific to \( W \) by bounding the number of near-optimal points of the Brownian motion in expectation. We do it by rewriting it as two Brownian meanders \cite{Durrett}, both starting at the maximum of the Brownian, one going backward and the other one forward with the Brownian meander \( W^+ \) defined as

\[
\forall t \in [0, 1] \quad W^+(t) = \frac{W(\tau + t(1 - \tau))}{\sqrt{1 - \tau}}, \quad \text{where } \tau = \sup \{ t \in [0, 1] : W(t) = 0 \}.
\]

We use that the Brownian meander \( W^+ \) can be seen as a Brownian motion conditioned to be positive \cite{Denisov}. This is the main ingredient of Lemma 3.

**Lemma 3.** For any \( h \) and \( \eta \), the expected number of near-optimal points is bounded as

\[
\mathbb{E} [N_h (\eta)] \leq 6\eta^2 2^h.
\]

This lemma answers a question raised by Munos \cite{Munos}: *What is the near-optimality dimension of the Brownian motion?* We set \( \eta_h \equiv \eta_\varepsilon (1/2^h) \). In dimension one with the pseudo-distance \( \ell(x, y) \equiv \eta_h (|y - x|) \), the near-optimality dimension measures the rate of increase of \( N_h (\eta_h) \), the number of \( \eta_h \)-near-optimal points in \([0, 1]\) of the form \( k/2^h \). In Lemma 3, we prove that in expectation, this number increases as \( \mathcal{O}(\eta_h^2 2^h) = \mathcal{O}(\log (1/\varepsilon)) \), which is constant with respect to \( h \). This means that, for a given \( \varepsilon \), there is a metric under which the Brownian is Lipschitz with probability at least \( 1 - \varepsilon \) and has a near-optimality dimension \( d = 0 \) with \( C = \mathcal{O}(\log (1/\varepsilon)) \).

The final sample complexity bound is essentially constituted by one \( \mathcal{O}(\log (1/\varepsilon)) \) term coming from the standard \( \text{DOO} \) error for deterministic function optimization and another \( \mathcal{O}(\log (1/\varepsilon)) \) term because we need to adapt our pseudo-distance \( \ell \) to \( \varepsilon \) such that the Brownian is \( \ell \)-Lipschitz with probability \( 1 - \varepsilon \). The product of the two gives the final sample complexity bound of \( \mathcal{O}(\log^2 (1/\varepsilon)) \).

We now give a full proof of Lemma 3. As we pointed out, this result is the one that answers the open question of Munos \cite{Munos}.

**Proof of Lemma 3.** We denote by \( W \), the Brownian motion whose maximum \( M \) is first hit at the point defined as \( t_M = \inf \{ t \in [0, 1] : W(t) = M \} \) and \( B^+ \) a Brownian meander \cite{Durrett}. We also define

\[
B^+_0 (t) \equiv \frac{M - W(t_M - t \cdot t_M)}{\sqrt{t_M}} \quad \text{and} \quad B^+_1 (t) \equiv \frac{M - W(t_M + t(1 - t_M))}{\sqrt{1 - t_M}}.
\]

If \( \overset{\mathcal{L}}{=} \) denotes the equality in distribution, then Theorem 1 of Denisov \cite{Denisov} asserts that

\[
B^+ \overset{\mathcal{L}}{=} B^+_0 \overset{\mathcal{L}}{=} B^+_1 \overset{\mathcal{L}}{=} B^+_0 \quad \text{and} \quad t_M \text{ is independent from both } B^+_0 \text{ and } B^+_1.
\]
We upper-bound the expected number of $\eta$-near-optimal points for any integer $h \geq 0$ and any $\eta > 0$,

$$
E[\mathcal{N}_h(\eta)] = E \left[ \sum_{k=0}^{2^h} \mathbb{1} \left\{ W \left( \frac{k}{2^h} \right) > M - \eta \right\} \right] = \sum_{k=0}^{2^h} E \left[ \mathbb{1} \left\{ W \left( \frac{k}{2^h} \right) > M - \eta \right\} \right]
$$

$$
= \sum_{k=0}^{2^h} E \left[ \mathbb{1} \left\{ \left\{ W \left( \frac{k}{2^h} \right) > M - \eta \cap \frac{k}{2^h} \leq t_M \right\} \cup \left\{ W \left( \frac{k}{2^h} \right) > M - \eta \cap \frac{k}{2^h} > t_M \right\} \right\} \right]
$$

$$
= \sum_{k=0}^{2^h} E \left[ \mathbb{1} \left\{ B_0^+ \left( 1 - \frac{k}{t_M2^h} \right) < \frac{\eta}{\sqrt{t_M}} \cap \frac{k}{2^h} \leq t_M \right\} \right]
$$

$$
+ \sum_{k=0}^{2^h} E \left[ \mathbb{1} \left\{ B_1^+ \left( \frac{k/2^h}{1-t_M} \right) < \frac{\eta}{\sqrt{1-t_M}} \cap \frac{k}{2^h} > t_M \right\} \right].
$$

If $X$ and $Y$ are independent then for any function $f$,

$$
E[f(X, Y)] = E \left[ E[f(X, Y)|Y] \right] \quad \text{(law of total expectation)}
$$

$$
\leq \sup_y E[f(X, y)|Y = y] \quad \text{(for any $Z$, $E[Z] \leq \sup_w Z(w)$)}
$$

$$
= \sup_y E[f(X, y)]. \quad \text{(because $X$ and $Y$ are independent)}
$$

Since $t_M$ is independent from $B_0^+$ and $B_1^+$, then using the above with $X = (B_0^+, B_1^+)$, $Y = t_M$, and

$$f : (x_0, x_1), y \rightarrow \sum_{k=0}^{2^h} \left( \mathbb{1} \left\{ x_0 \left( 1 - \frac{k}{t_M2^h} \right) < \frac{\eta}{\sqrt{y}} \cap \frac{k}{2^h} \leq y \right\} + \mathbb{1} \left\{ x_1 \left( \frac{k/2^h}{1-y} \right) < \frac{\eta}{\sqrt{1-y}} \cap \frac{k}{2^h} > y \right\} \right),
$$

$$E[\mathcal{N}_h(\eta)] = E[f(X, Y)] \leq \sup_{u \in [0,1]} E[f(X, u)]
$$

$$
\leq \sup_{u \in [0,1]} \left\{ \sum_{k=0}^{2^h} E \left[ B_0^+ \left( 1 - \frac{k}{u2^h} \right) < \frac{\eta}{\sqrt{u}} \cap \frac{k}{2^h} \leq u \right] \right\}
$$

$$
+ \sup_{u \in [0,1]} \left\{ \sum_{k=0}^{2^h} E \left[ B_1^+ \left( \frac{k/2^h}{1-u} \right) < \frac{\eta}{\sqrt{1-u}} \cap \frac{k}{2^h} > u \right] \right\}
$$

$$
= \sup_{u \in [0,1]} \left\{ \sum_{k=0}^{\lfloor u2^h \rfloor} P \left[ B_0^+ \left( 1 - \frac{k}{u2^h} \right) < \frac{\eta}{\sqrt{u}} \right] \right\}
$$

$$
+ \sup_{u \in [0,1]} \left\{ \sum_{k=\lfloor u2^h \rfloor}^{2^h} P \left[ B_1^+ \left( \frac{k/2^h}{1-u} \right) < \frac{\eta}{\sqrt{1-u}} \right] \right\}
$$

$$
= 2 \sup_{u \in [0,1]} \left\{ \alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 \right\},
$$

with

$$
\alpha_1 = \sum_{k=0}^{\lfloor 2^h \eta^2 \rfloor} P \left[ B_0^+ \left( 1 - \frac{k}{u2^h} \right) < \frac{\eta}{\sqrt{u}} \right], \quad \alpha_2 = \sum_{k=\lfloor 2^h \eta^2 \rfloor}^{\lfloor u2^h \rfloor} P \left[ B_0^+ \left( 1 - \frac{k}{u2^h} \right) < \frac{\eta}{\sqrt{u}} \right]
$$

$$
\alpha_3 = \sum_{k=\lfloor u2^h \rfloor}^{\lfloor u2^h \rfloor - \lfloor 2^h \eta^2 \rfloor} P \left[ B_0^+ \left( 1 - \frac{k}{u2^h} \right) < \frac{\eta}{\sqrt{u}} \right], \quad \text{and} \quad \alpha_4 = \sum_{k=\lfloor u2^h \rfloor - \lfloor 2^h \eta^2 \rfloor}^{\lfloor u2^h \rfloor} P \left[ B_0^+ \left( 1 - \frac{k}{u2^h} \right) < \frac{\eta}{\sqrt{u}} \right].
$$
Since a probability is always upper-bounded by 1, we bound $\alpha_1$ and $\alpha_4$, both by $\eta^2 2^h$, to get that $\alpha_1 + \alpha_4 \leq 2\eta^2 2^h$. We now bound the remaining probabilities appearing in the above expression by integrating over the distribution function of Brownian meander [Durrett et al., 1977, Equation 1.1],

\[
P \left[ B^+_0 (t) < x \right] = 2 \sqrt{2\pi} \int_{0}^{x} \frac{y \exp \left( -y^2 / (2t) \right)}{t^{1/2} \pi} \int_{0}^{y} \frac{\exp \left( -u^2 / (2(1-t)) \right)}{\sqrt{2\pi (1-t)}} \, du \, dy
\]

\[
\leq \frac{2}{t^{1/2} \pi (1-t)} \int_{0}^{x} y^2 \exp \left( -y^2 / (2t) \right) \, dy
\]

\[
\leq \frac{2x^3}{3t^{1/2} \pi (1-t)} \leq \frac{2x^3}{3t (1-t) \sqrt{2\pi (1-t)}} = \frac{2}{3 \sqrt{2\pi}} \left( \frac{x}{\sqrt{t} (1-t)} \right)^3,
\]

where the first two inequalities are obtained by upperbounding the terms $\exp \left( \cdot \right)$ by one. Now, we use the above bound to bound $\alpha_2 + \alpha_3$,

\[
\alpha_2 + \alpha_3 = \sum_{k=\lceil 2^h \eta^2 \rceil}^{\lfloor u^2 \rfloor} \mathbb{P} \left[ B^+_0 \left( 1 - \frac{k}{u^2 2^h} \right) < \frac{\eta}{\sqrt{u}} \right] + \sum_{k=\lceil 2^h \eta^2 \rceil}^{\lfloor u^2 \rfloor} \mathbb{P} \left[ B^+_0 \left( 1 - \frac{k}{u^2 2^h} \right) < \frac{\eta}{\sqrt{u}} \right]
\]

\[
\leq \sum_{k=\lceil 2^h \eta^2 \rceil}^{\lfloor u^2 \rfloor} \frac{2}{3 \sqrt{2\pi}} \left( \frac{\eta}{\sqrt{u}} \right)^3 + \sum_{k=\lceil 2^h \eta^2 \rceil}^{\lfloor u^2 \rfloor} \frac{2}{3 \sqrt{2\pi}} \left( \frac{\eta}{\sqrt{u}} \right)^3
\]

\[
\leq \frac{1}{6 \sqrt{\pi}} \left( \frac{\eta}{\sqrt{u}} \right)^3 + \frac{1}{6 \sqrt{\pi}} \left( \frac{\eta}{\sqrt{u}} \right)^3
\]

\[
\leq \frac{1}{6 \sqrt{\pi}} \left( \frac{\eta}{\sqrt{u}} \right)^3 + \frac{1}{6 \sqrt{\pi}} \left( \frac{\eta}{\sqrt{u}} \right)^3
\]

\[
= \frac{\left( 2^h \eta^2 \right)^{3/2}}{6 \sqrt{\pi}} \left( \sum_{k=\lceil 2^h \eta^2 \rceil}^{\lfloor u^2 \rfloor} \frac{1}{k^{3/2}} + \sum_{k=\lceil 2^h \eta^2 \rceil}^{\lfloor u^2 \rfloor} \frac{1}{k^{3/2}} \right)
\]

\[
\leq \frac{\left( 2^h \eta^2 \right)^{3/2}}{3 \sqrt{\pi}} \leq \frac{3}{\sqrt{2^h \eta^2}} \leq \frac{1}{\sqrt{\pi}} \eta^2 2^h \leq \eta^2 2^h,
\]

where in the last line we used that for any $k_0 \geq 1$,

\[
\sum_{k=k_0}^{\infty} \frac{1}{k^{3/2}} \leq \frac{1}{k_0^{3/2}} + \sum_{k=k_0+1}^{\infty} \int_{k-1}^{k} \frac{1}{u^{3/2}} \, du = \frac{1}{k_0^{3/2}} + \int_{k_0}^{\infty} \frac{1}{u^{3/2}} \, du = \frac{1}{k_0^{3/2}} + \frac{2}{\sqrt{k_0}} \leq \frac{3}{\sqrt{k_0}}.
\]

We finally have

\[
\forall u, \quad \alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 \leq 3 \eta^2 2^h
\]

and therefore

\[
\mathbb{E} \left[ N_6 (\eta) \right] \leq 6 \eta^2 2^h.
\]

To conclude the analysis, we put Lemma 2 and Lemma 3 together in order to bound the sample complexity conditioned on event $C$.

**Lemma 4.** There exists $c > 0$ such that for all $\varepsilon \leq 1/2$, $\mathbb{E} \left[ N_6 | C \right] \leq c \log^2 (1/\varepsilon) .

We also bound the sample complexity on $C^c$, the event complementary to $C$, by the total number of possible intervals $[k/2^h, (k+1)/2^h]$ with $n_c (1/2^h) \leq \varepsilon$. Then, we combine it with Lemma 1, Lemma 2, and Lemma 3 to get Proposition 2 which is the second part of the main theorem.
**Proposition 2.** There exists $c > 0$ such that for all $\varepsilon < 1/2$, 

$$
\mathbb{E}[N_{\varepsilon}] \leq c \log^2 (1/\varepsilon)
$$

Proposition 2 together with Proposition 1 concludes the proof of Theorem 1.

### 5 Numerical evaluation

For an illustration, we ran a simple experiment and for different values of $\varepsilon$, we computed the average sample complexity $N_{\varepsilon}$ on 250 independent runs (graph on the left). We also plot one point for each run of OOB instead of averaging the sample complexity. The experiment indicates a linear dependence between the sample complexity and $\log^2(1/\varepsilon)$.

![Graph showing expected number of samples against $\log^2(1/\varepsilon)$](image1)

![Graph showing number of samples against $\log^2(1/\varepsilon)$](image2)

### 6 Conclusion and discussion

We presented OOB, an optimistic algorithm inspired by DOO [Munos, 2011] that efficiently optimizes a Brownian motion. We prove that the sample complexity of OOB is of order $O\left(\log^2 \left(1/\varepsilon\right)\right)$ which improves over the previously best known bound [Calvin et al., 2017]. Yet, several questions remain open as we are not aware of a lower bound for the Brownian motion optimization. Therefore, it is still unknown whether $O\left(\log^2 \left(1/\varepsilon\right)\right)$ is the minimax rate of the sample complexity or if there exists an algorithm able to maximize a Brownian motion faster.

A natural question opens up, is what we would be need to do, if wants to use our approach to get similar results for Gaussian processes with a different kernel. The optimistic approach we took is quite general, only Lemma 3 would need additional work as this the ingredient most specific to Brownian motion. Notice, that Lemma 3 bounds (in expectation) the number of near-optimal nodes of a Brownian motion and it uses the result of Denisov [1984] which is based on 2 items:

- A Brownian motion can be rewritten as an affine transformation of a Brownian motion conditioned to be positive, translated by an (independent) time at which the Brownian motion attains its maximum.
- A Brownian motion conditioned to be positive is a Brownian meander. This requires some additional work to prove that a Brownian motion conditioned to be positive is actually properly defined.

A similar result for another Gaussian process or its generalization of the result to a larger class of Gaussian processes would need to adapt or generalize these two items in Lemma 3. However, the adaptation/generalization of the other parts of the proof would be straightforward. Moreover for the second item, the full law of the process conditioned to be positive is actually not needed, only the local time of the Gaussian process conditioned to be positive at points near zero.
References


A Full proofs

Lemma 1. The event \( C \) from Definition 1 holds with high probability,
\[
\mathbb{P}[C^c] \leq \varepsilon^5.
\]

Proof. For any interval \( I \),
\[
B_I = \max (W(a), W(b)) + \eta_e (b - a) \quad \text{(by definition of } B_I) \]
\[
= \frac{W(a) + W(b)}{2} + \frac{|W(a) - W(b)|}{2} + \eta_e (b - a) \quad \forall x, y, \max(x, y) = \frac{1}{2} (x + y + |x - y|) \]
\[
\geq \frac{W(a) + W(b)}{2} + \sqrt{\frac{(W(a) - W(b))^2}{4} + (\eta_e (b - a))^2} \quad \forall x > 0, y > 0, \ (x + y)^2 \geq x^2 + y^2. \]

We now define
\[
t_{h,k} \triangleq \frac{k}{2^h}, \quad \Delta_{h,k} \triangleq \frac{W(t_{h,k}) - W(t_{h,k+1})}{2}, \quad \eta_h \triangleq \eta_e (b - a),
\]
and
\[
A_{h,k} \triangleq \left\{ \sup_{t \in [k/2^h, (k+1)/2^h]} W(t) > B_{[k/2^h, (k+1)/2^h]} \right\}.
\]
First, for any \( a < b \), the law of the maximum of a Brownian bridge gives us
\[
\forall x \geq \max (W(a), W(b)): \quad \mathbb{P} \left[ \sup_{t \in [a,b]} W(t) > x \middle| W(a) = W_a, W(b) = W_b \right] = \exp \left( - \frac{2(x - W_a)(x - W_b)}{b - a} \right).
\]
Combining it with the definition of \( A_{h,k} \) and the first inequality of the proof we get
\[
\mathbb{P} \left[ A_{h,k} \middle| W(t_{h,k}), W(t_{h,k} + 1) \right] \leq \exp \left( - \frac{2 \left( \frac{W(t_{h,k} + 1) - W(t_{h,k})}{2} + \sqrt{\frac{\Delta_{h,k}^2 + \eta_h^2}{t_{h,k+1} - t_{h,k}}} \right)}{t_{h,k+1} - t_{h,k}} \right)
\]
\[
= \exp \left( - \frac{2 \left( \sqrt{\frac{\Delta_{h,k}^2 + \eta_h^2}{2^h}} - \Delta_{h,k} \right) \left( \sqrt{\frac{\Delta_{h,k}^2 + \eta_h^2}{2^h}} + \Delta_{h,k} \right)}{2^h} \right)
\]
\[
= \exp \left( -2^{h+1} \frac{\eta_h^2}{2^h} \right) = \exp \left( -2^{h+1} \frac{5}{2} \frac{\varepsilon}{2^h} \ln \left( \frac{2^h}{\varepsilon} \right) \right) = \left( \frac{\varepsilon}{2^h} \right)^5.
\]
By definition of \( C, C \triangleq \bigcap_{h=0}^{\infty} \bigcap_{k=0}^{2^h-1} A_{h,k} \triangleq \bigcup_{h=0}^{\infty} \bigcup_{k=0}^{2^h-1} A_{h,k} \). We use a union bound on all \( A_{h,k} \) to finally get
\[
\mathbb{P}[C^c] \leq \sum_{h \geq 1} \sum_{k=0}^{2^h-1} \mathbb{P}[A_{h,k}] \leq \sum_{h \geq 1} \sum_{k=0}^{2^h-1} \left( \frac{\varepsilon}{2^h} \right)^5 \leq \sum_{h \geq 1} \frac{\varepsilon^5}{2^{4h}} \leq \varepsilon^5.
\]

\( \square \)

Proposition 1. The estimator \( \hat{M}_\varepsilon = W \left( \hat{t}_\varepsilon \right) \) has a probably approximately correct guarantee,
\[
\mathbb{P} \left[ M - \hat{M}_\varepsilon > \varepsilon \right] \leq \varepsilon.
\]
Proof. Let \( I^\ast = [a, b] \) be the interval that the algorithm would split next, if it was not terminated. Since the algorithm only splits the interval with the highest upper bound then \( B_{l^\ast} = \sup_{t \in I^\ast} B_t \). Also let \( I_m \in \mathcal{I}_t \) be one of the intervals where a maximum is reached, \( t_m \in \arg \max_{t \in [0, 1]} W(t) \) and \( t_m \in I_m \). Then, on event \( C \),

\[
M \leq B_{l_m} \leq B_{l^\ast} = \max (W(a), W(b)) + \eta_\epsilon (b - a).
\]

Since the algorithm terminated, we have that \( \eta_\epsilon (b - a) \leq \varepsilon \) and therefore,

\[
\max (W(a), W(b)) \geq M - \varepsilon,
\]

which combined with Lemma 1 finishes the proof as \( \varepsilon^5 \leq \varepsilon \).

\[\Box\]

**Lemma 2.** Under event \( C \), the number of evaluated points \( N_\varepsilon \) by the algorithm verifies

\[
N_\varepsilon \leq 2 \sum_{h=0}^{h_{\max}} N_h (\eta_\epsilon \left(1/2^h\right)),
\]

with \( h_{\max} \) being the smallest \( h \) such that \( \eta_\epsilon \left(1/2^h\right) \leq \varepsilon \).

Proof. Let \([a, b]\) be an interval of \( \mathcal{I}_t \) such that \( \max (W(a), W(b)) + \eta_\epsilon (b - a) < M \). Let \( I^\ast \in \mathcal{I}_t \) be the interval that the algorithm splits after \( t \) function evaluations. Since the algorithm only splits the interval with the highest upper bound, then \( B_{l^\ast} = \sup_{t \in I^\ast} B_t \). Moreover, if we let \( I_m \in \mathcal{I}_t \) be one of the intervals where a maximum is reached, \( t_m \in \arg \max_{t \in [0, 1]} W(t) \) and \( t_m \in I_m \), then on event \( C \),

\[
\max (W(a), W(b)) + \eta_\epsilon (b - a) = B_t < M \leq B_{l_m} \leq B_{l^\ast}.
\]

Therefore, under \( C \), a necessary condition for an interval \( I = [a, b] \) to be split during the execution of \( \text{OPT} \) is that \( \max (W(a), W(b)) \geq M - \eta_\epsilon (b - a) \), which means that either \( a \) or \( b \) or both are \( \eta_\epsilon (b - a) \)-near-optimal.

From the termination condition of the algorithm, we know that any interval that is satisfying \( I = [k/2^h, (k + 1)/2^h] \) with \( h \geq h_{\max} \) will not be split during the execution.\(^2\) Therefore, another necessary condition for an interval \( I = [a, b] \) to be split during the execution is that \( b - a > 1/2^{h_{\max}} \).

Writing \( \eta_h \triangleq \eta_\epsilon \left(1/2^h\right) \), we deduce from these two necessary conditions that

\[
N \leq \sum_{h=0}^{h_{\max}} \sum_{k=0}^{2^h-1} \left\{ \frac{k}{2^h} \text{ or } \frac{k + 1}{2^h} \text{ is } \eta_h \text{-near-optimal} \right\} \\
\leq \sum_{h=0}^{h_{\max}} \sum_{k=0}^{2^h-1} \left\{ \frac{k}{2^h} \text{ is } \eta_h \text{-near-optimal} \right\} + \sum_{h=0}^{h_{\max}} \sum_{k=0}^{2^h-1} \left\{ \frac{k + 1}{2^h} \text{ is } \eta_h \text{-near-optimal} \right\} \\
\leq 2 \sum_{h=0}^{h_{\max}} \sum_{k=0}^{2^h-1} \left\{ \frac{k}{2^h} \text{ is } \eta_h \text{-near-optimal} \right\} = 2 \sum_{h=0}^{h_{\max}} N_h (\eta_h).
\]

\[\Box\]

**Lemma 4.** There exists \( c > 0 \) such that for all \( \varepsilon \leq 1/2 \), \( \mathbb{E}[N_\varepsilon | C] \leq c \log^2 \left(1/\varepsilon\right) \).

Proof. By definition of \( h_{\max} \),

\[
\varepsilon^2 \leq \frac{5}{2 \times 2^{h_{\max}}} \ln \left(\frac{2^{h_{\max}}}{\varepsilon}\right) \leq \frac{5}{2} \frac{\sqrt{2^{h_{\max}}/\varepsilon}}{2^{h_{\max}}},
\]

from which we deduce that

\[
\sqrt{\varepsilon} \varepsilon^2 \leq \frac{5}{2 \times 2^{h_{\max}/2}} \quad \text{and therefore} \quad 2^{h_{\max}} \leq \frac{25}{4\varepsilon^5}.
\]

\(^2\)This holds despite \( \eta_\epsilon (\cdot) \) is not always decreasing.